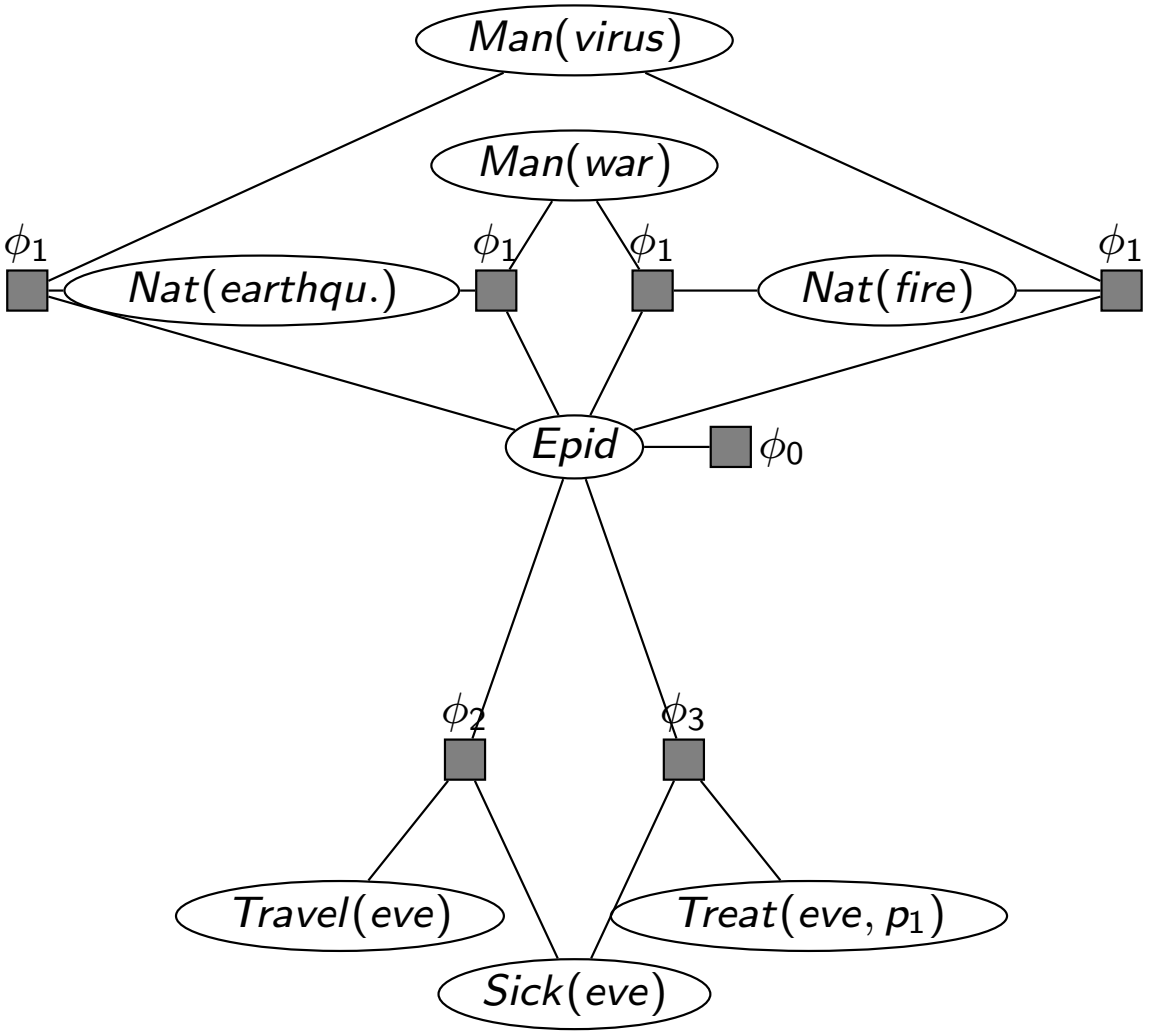
Parameterised Queries and Lifted Query Answering

Tanya Braun, Ralf Möller

Institute of Information Systems
University of Lübeck

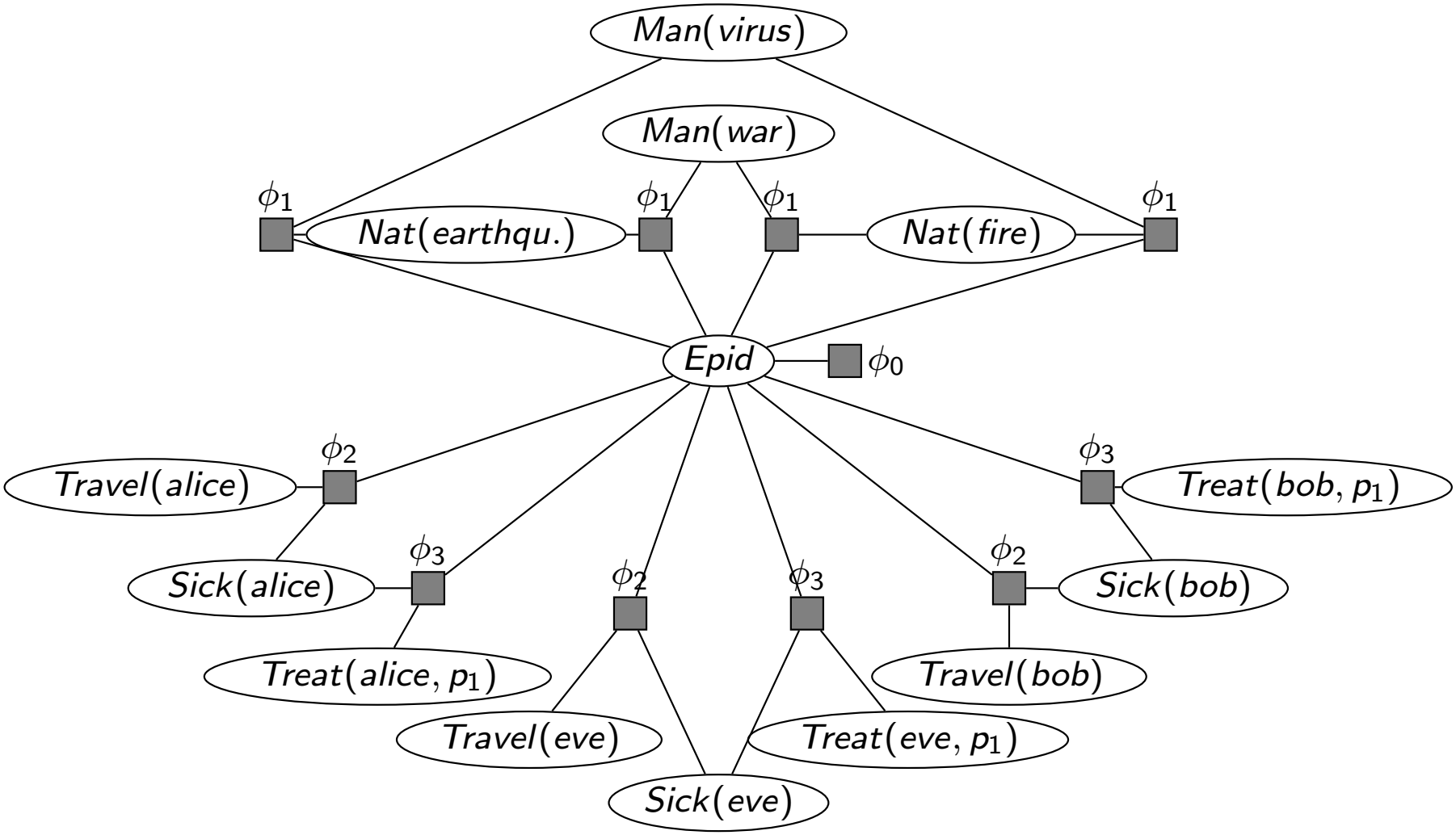
July 16, 2018

Probabilistic Graphical Models



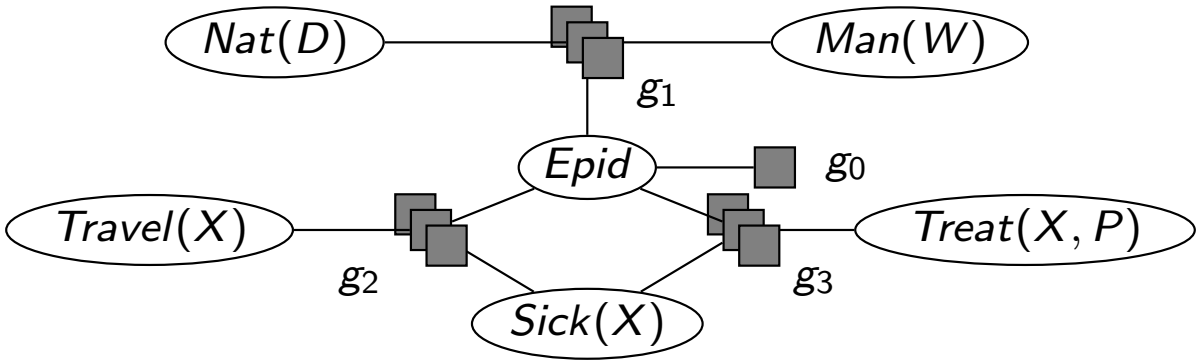
Query answering (QA): Eliminate all non-query variables

Probabilistic Graphical Models



Query answering (QA): Eliminate all non-query variables

Parameters and Lifted Variable Elimination



- Parameterisation
  - → Compact representation for isomorphic instances
- Lifted variable elimination (LVE)[1] for query answering
  - → Elimination: $\sum$ over range values of random variables
  - → Lifting: eliminate once and account for isomorphic instances

[1] Poole (2003), de Salvo Braz et al. (2006), Milch et al. (2008), Apsel & Brafman (2011), Taghipour et al. (2013)

Conference Contribution

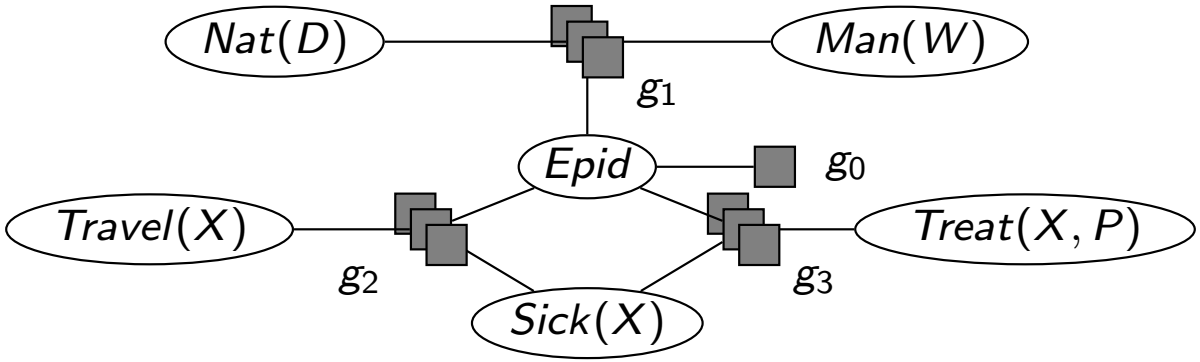
Parameterised Queries

- Compact query representation
- Lifted query answering with LVE
- Compact result representation if possible

$$\text{P(Sick(alice), Sick(eve), Sick(bob))}$$
$$\Downarrow$$
$$P(Sick(X))$$

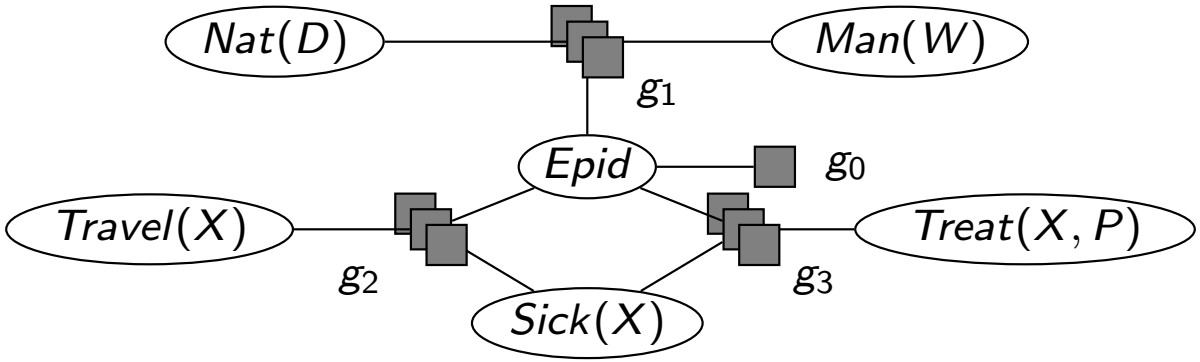
Marginal Distribution Ground Queries

$P(Sick(eve))$

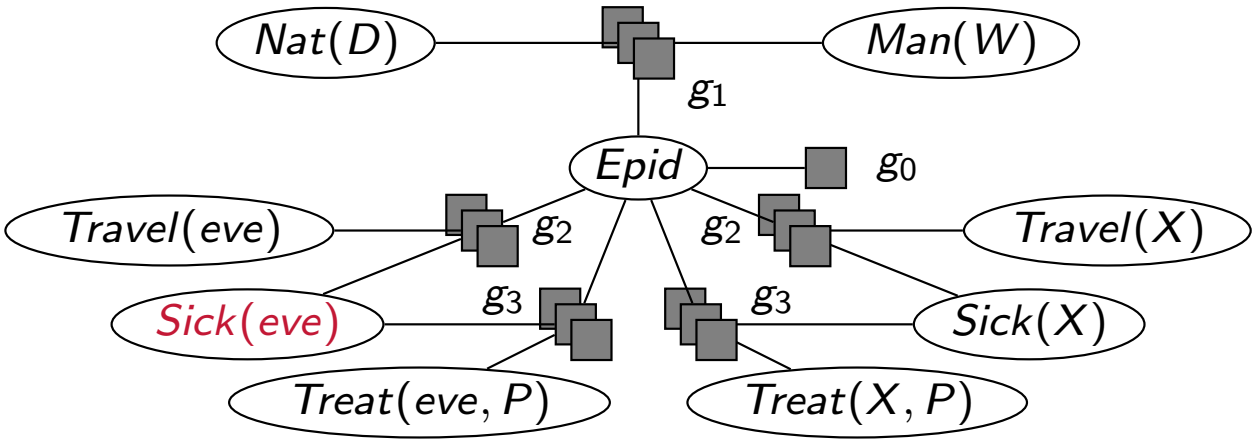


Marginal Distribution Ground Queries

$P(Sick(eve))$

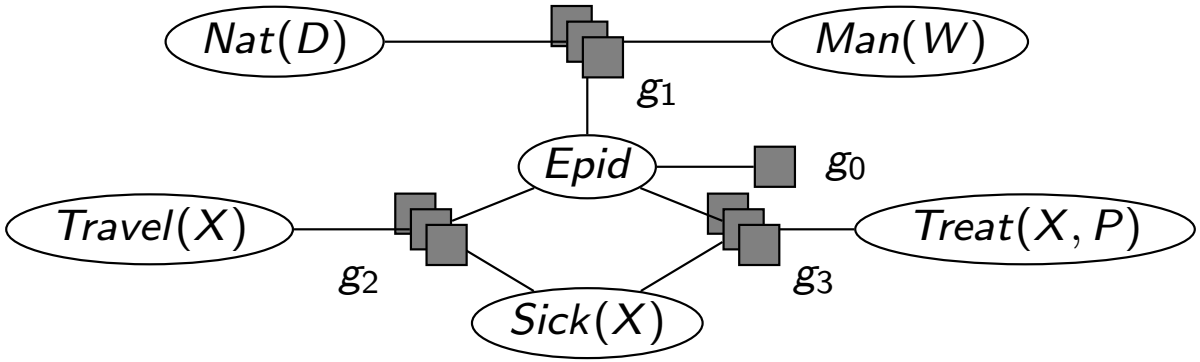


Preemptive shattering

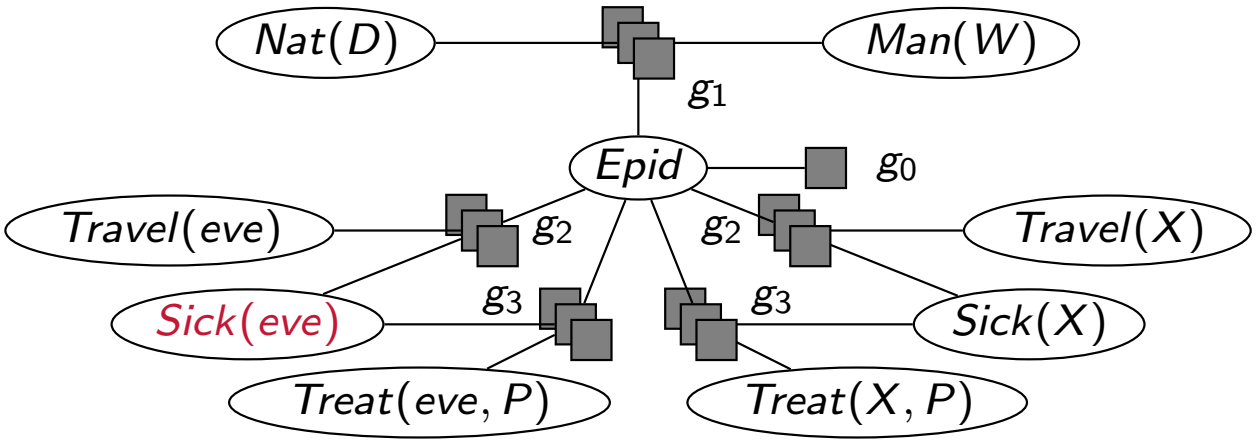


Marginal Distribution Ground Queries

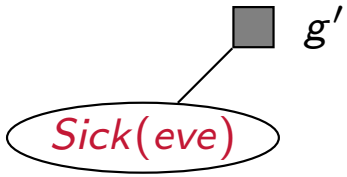
$P(Sick(eve))$



Preemptive shattering

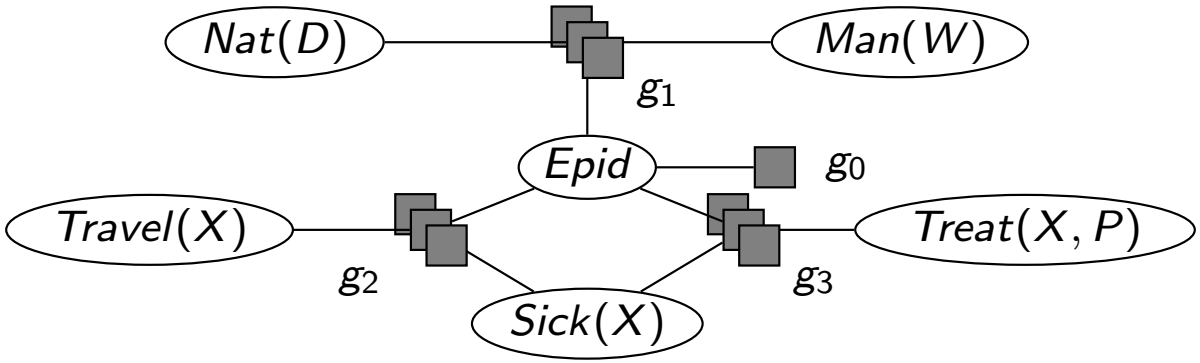


Elimination



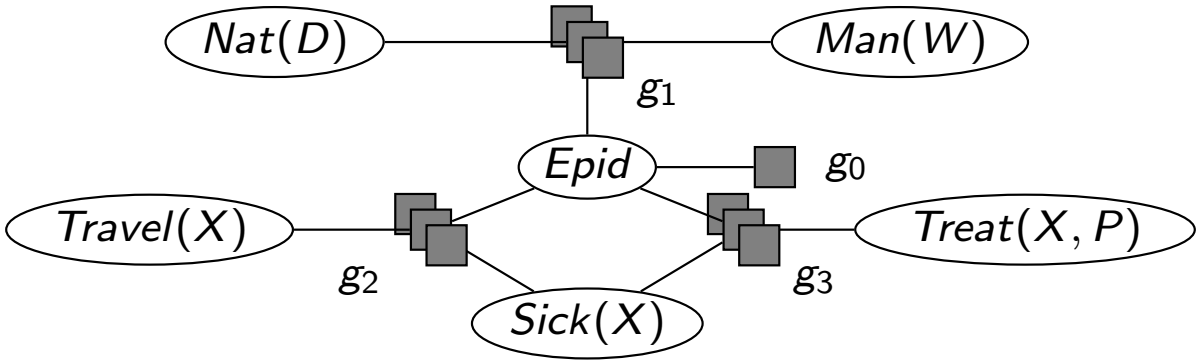
Problem: Isomorphic Instances in Queries

$P(Sick(eve), Sick(alice), Sick(bob))$

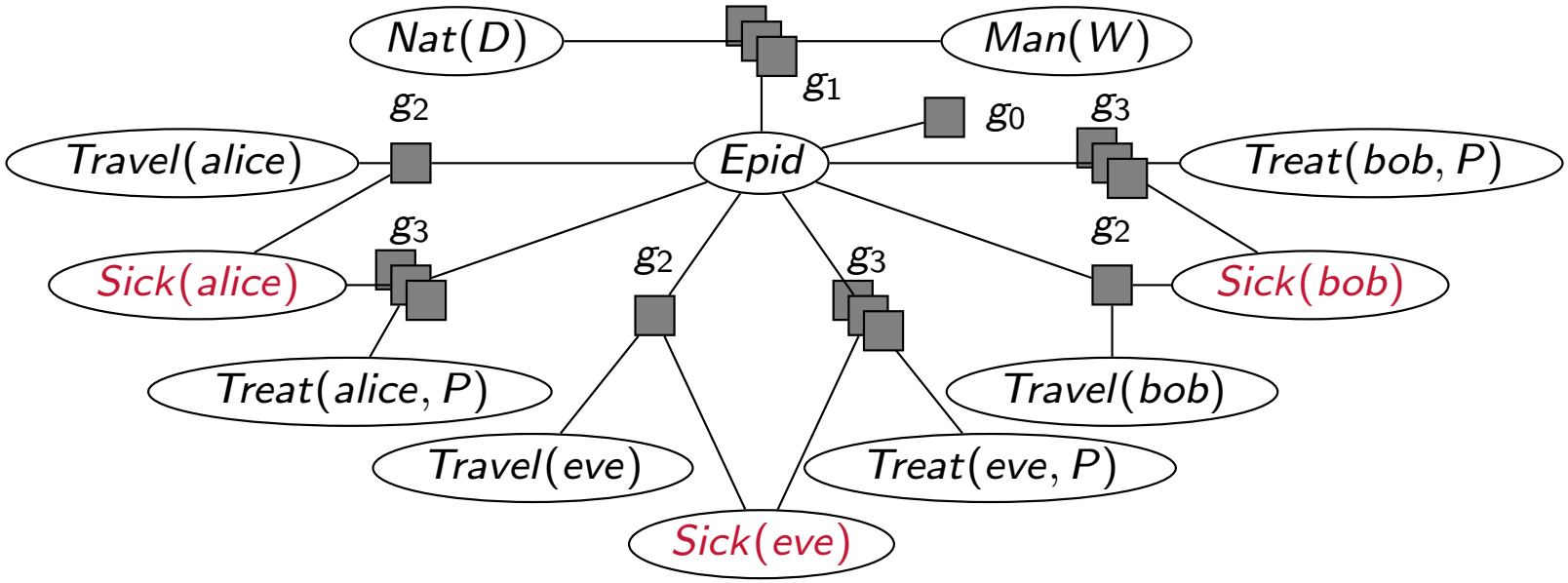


Problem: Isomorphic Instances in Queries

$P(Sick(eve), Sick(alice), Sick(bob))$

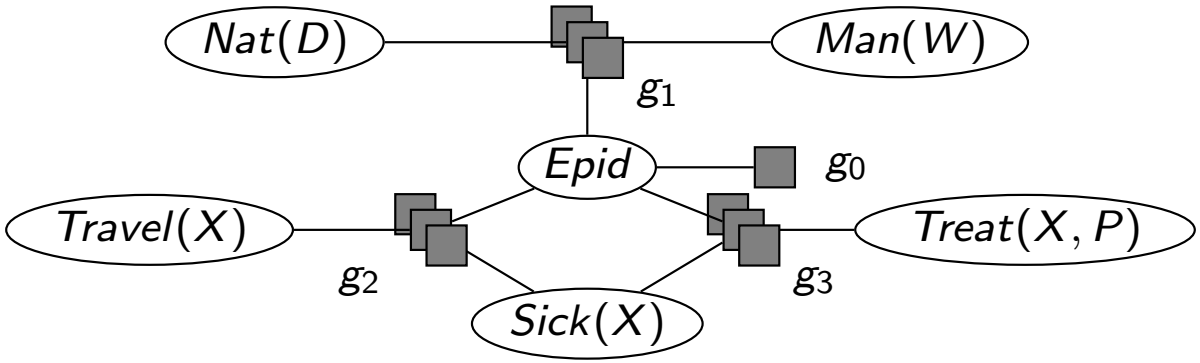


Preemptive shattering

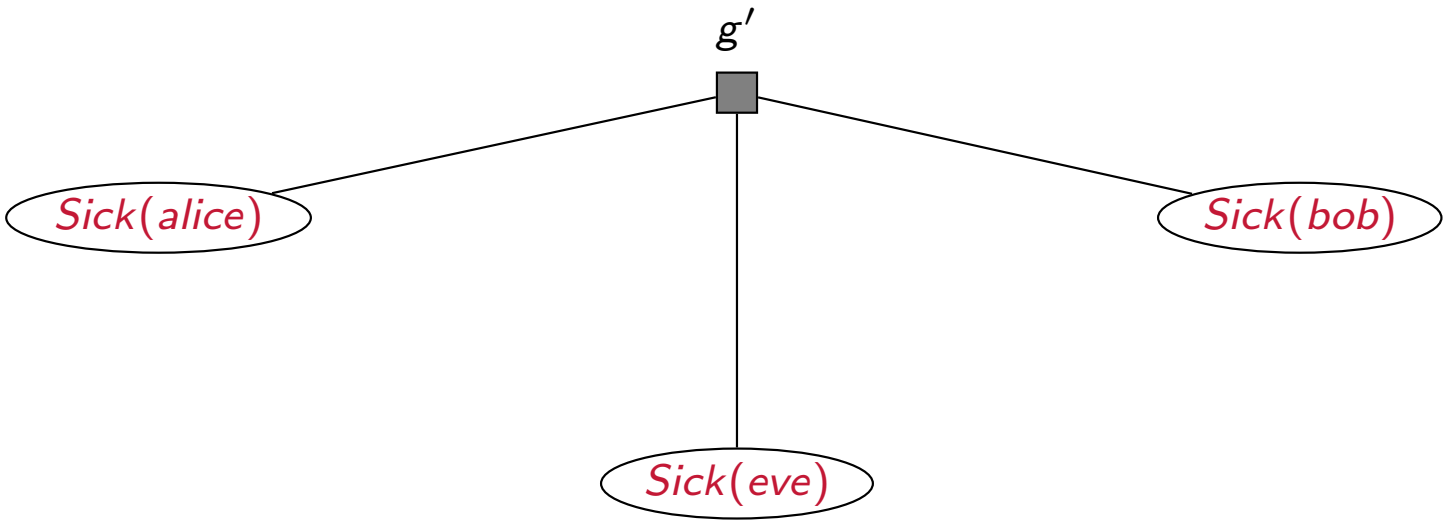


Problem: Isomorphic Instances in Queries

$P(Sick(eve), Sick(alice), Sick(bob))$

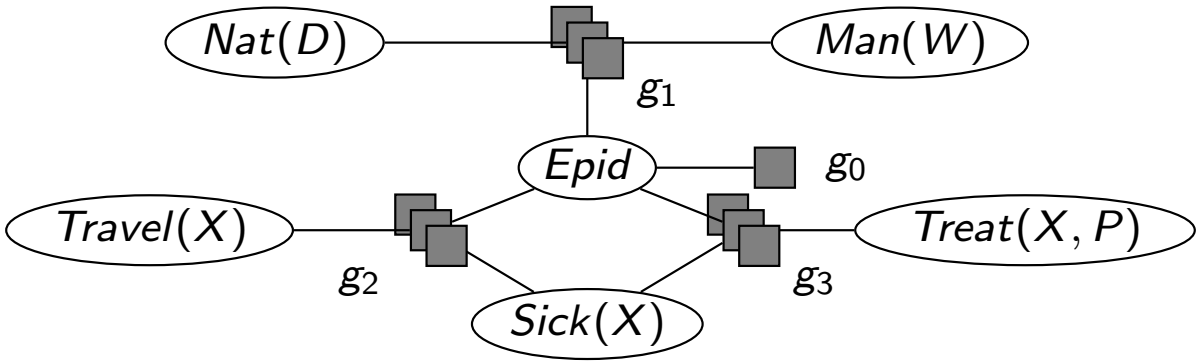


Elimination



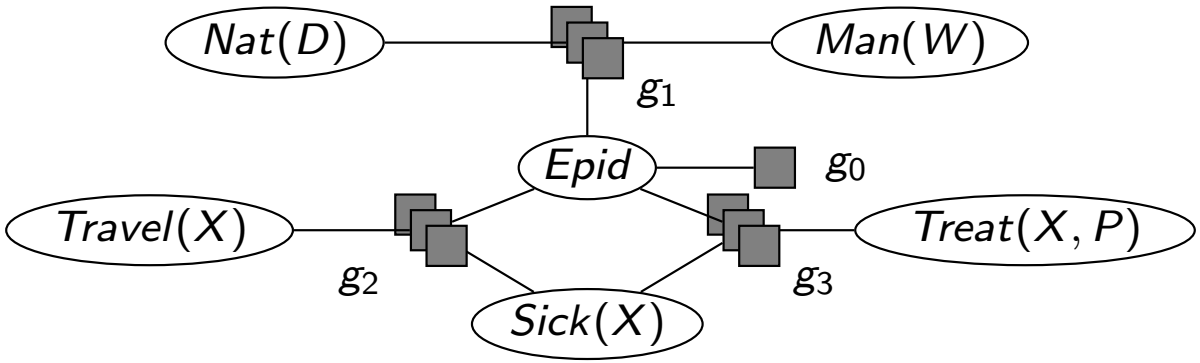
Patch 1: Smaller Models for Query Answering

Lifted Junction Tree Algorithm (LJT)

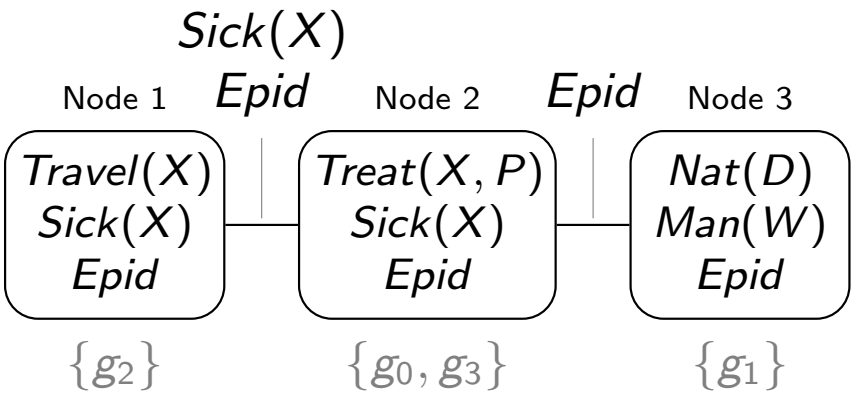


Patch 1: Smaller Models for Query Answering

Lifted Junction Tree Algorithm (LJT)

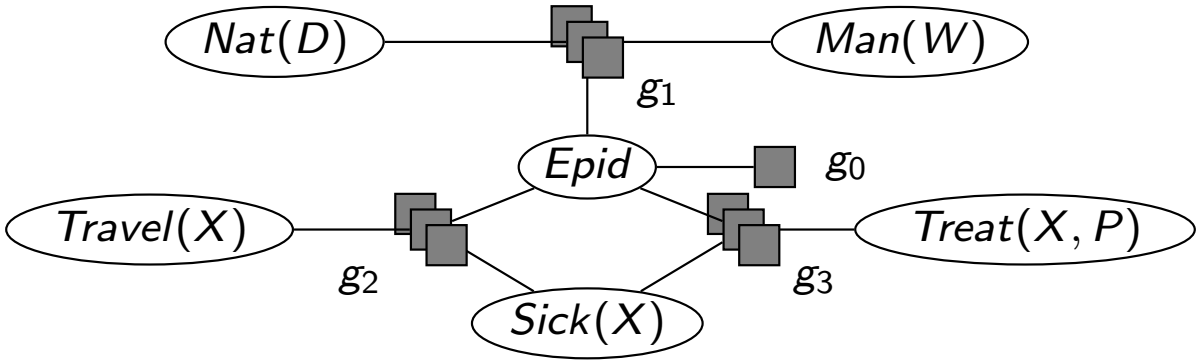


Construction

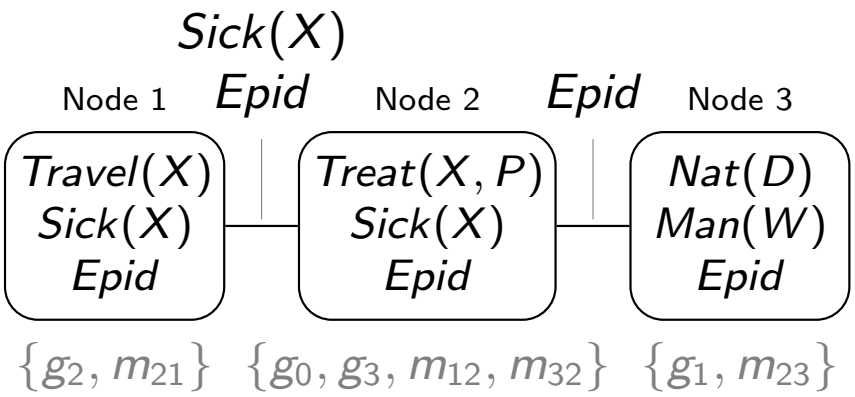


Patch 1: Smaller Models for Query Answering

Lifted Junction Tree Algorithm (LJT)

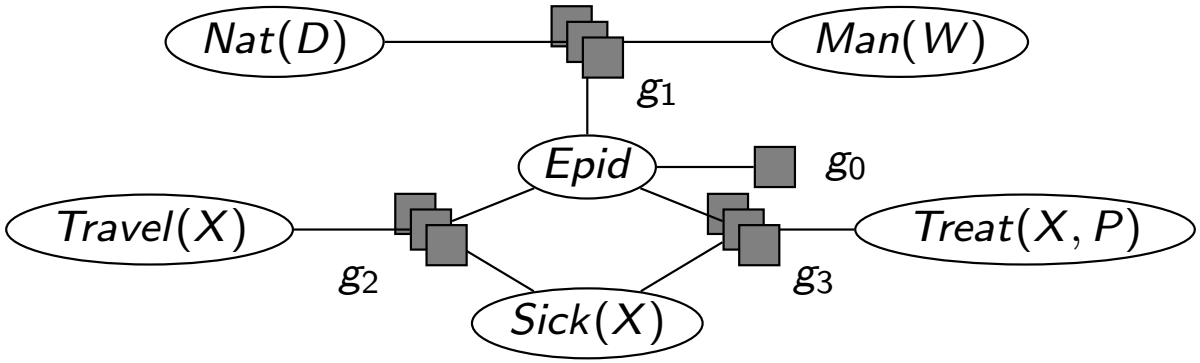


Message passing

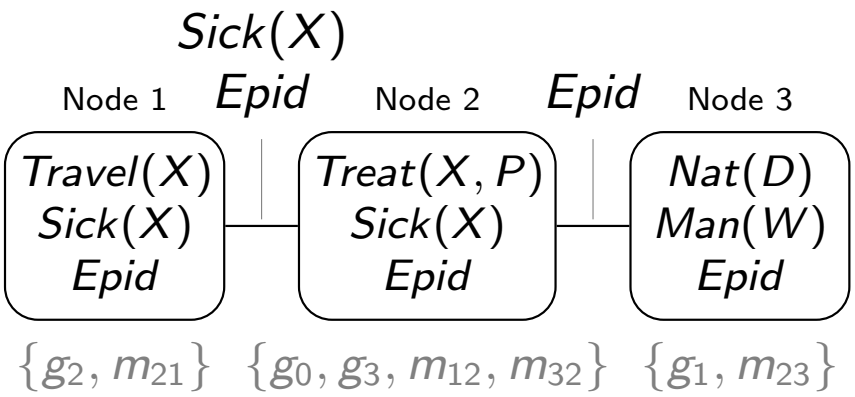


Patch 1: Smaller Models for Query Answering

$P(Sick(eve), Sick(alice), Sick(bob))$

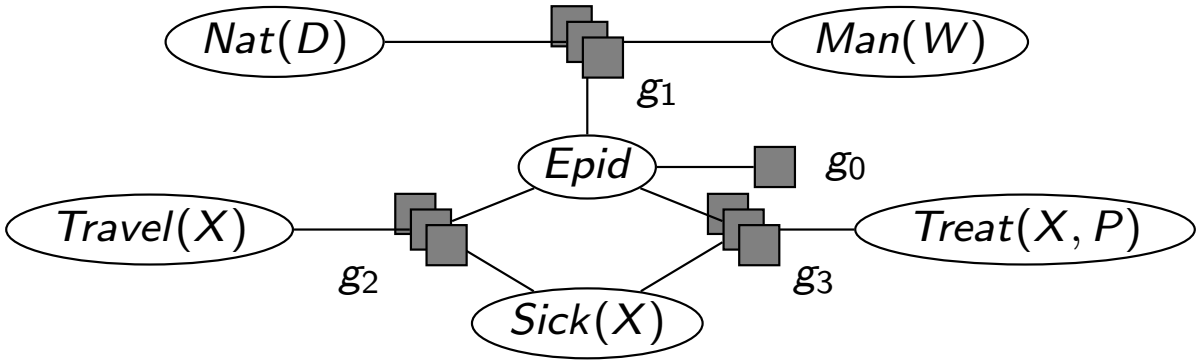


Query answering

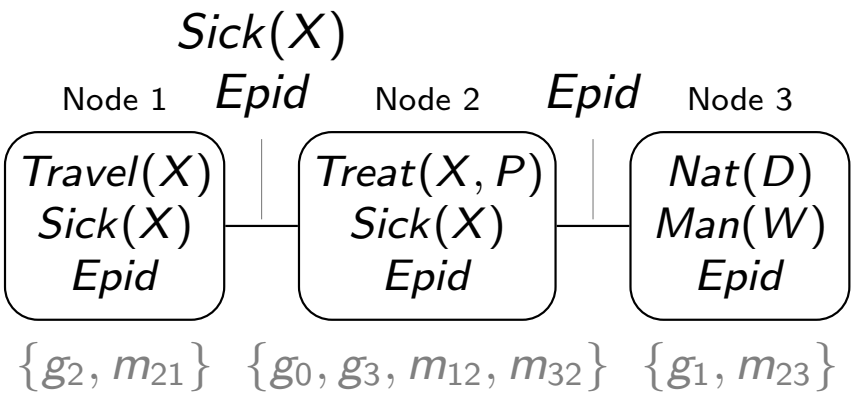


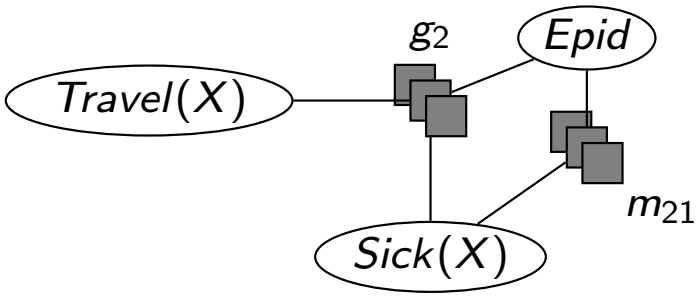
Patch 1: Smaller Models for Query Answering

$P(Sick(eve), Sick(alice), Sick(bob))$



Query answering

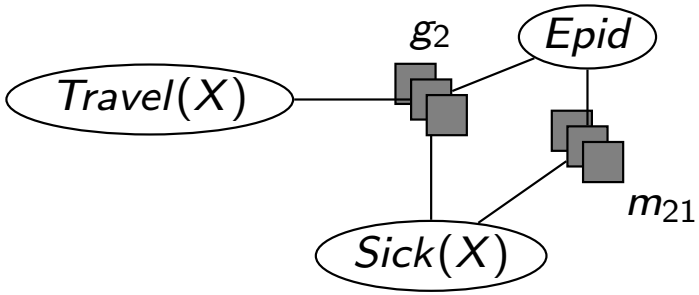




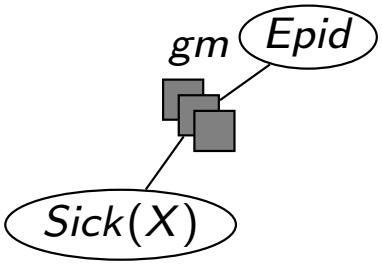
Node 1 model

Patch 2: Ondemand Shattering

$P(Sick(eve), Sick(alice), Sick(bob))$

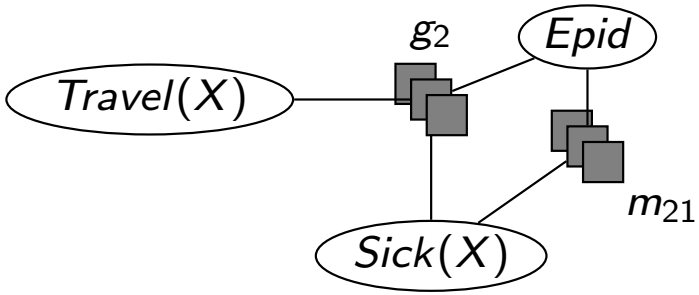


Elimination

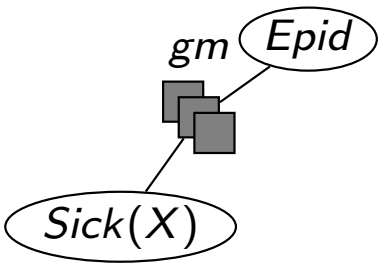


Patch 2: Ondemand Shattering

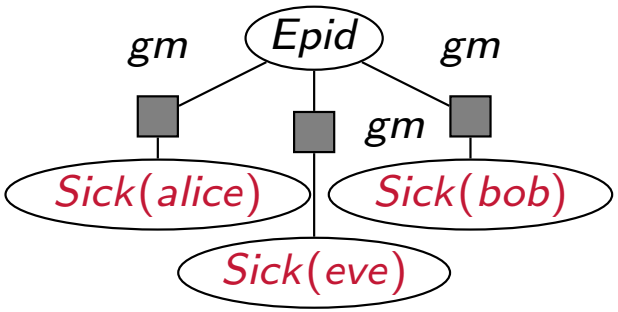
$P(Sick(eve), Sick(alice), Sick(bob))$



Elimination

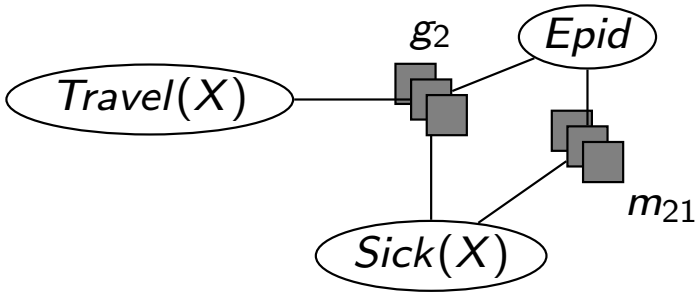


Ondemand shattering

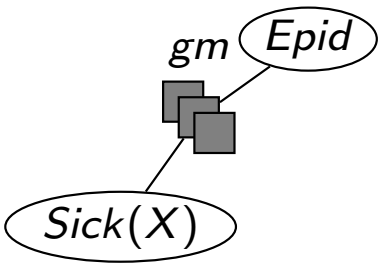


Patch 2: Ondemand Shattering

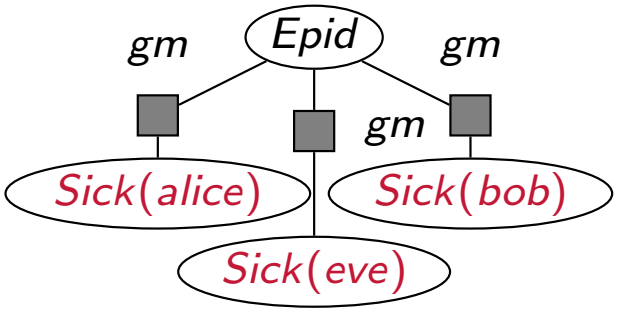
$P(Sick(eve), Sick(alice), Sick(bob))$



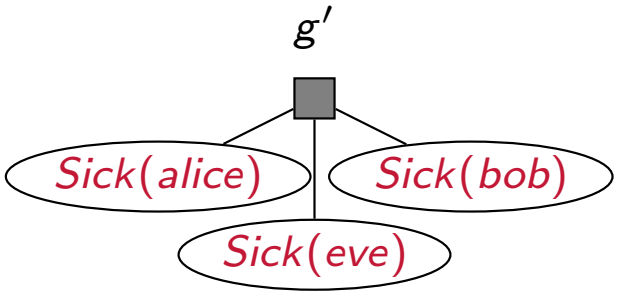
Elimination



Ondemand shattering

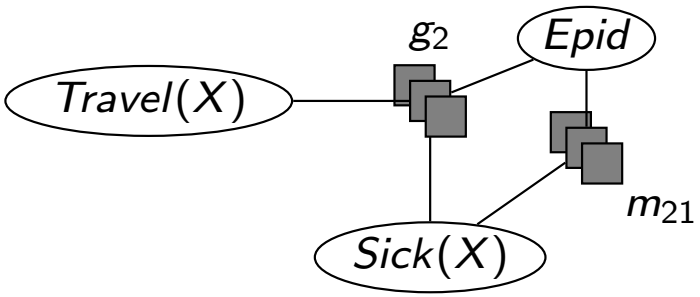


Elimination



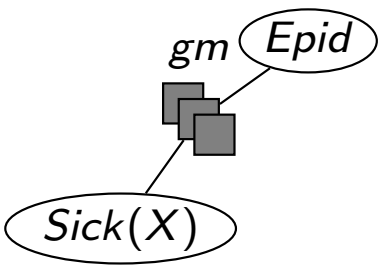
Patch 2: Ondemand Shattering

$P(Sick(eve), Sick(alice), Sick(bob))$

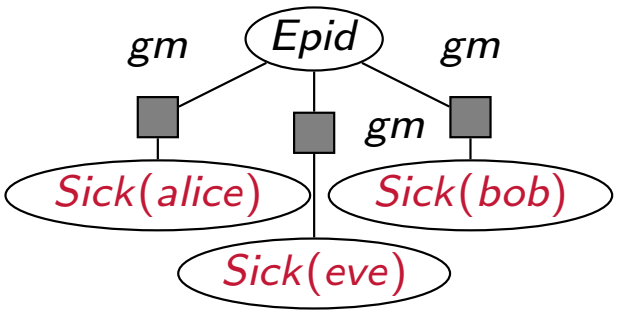


$Sick(alice)$	$Sick(eve)$	$Sick(bob)$	g'
0	0	0	1
0	0	1	2
0	1	0	2
1	0	0	2
1	1	0	3
1	0	1	3
1	1	0	3
1	1	1	4

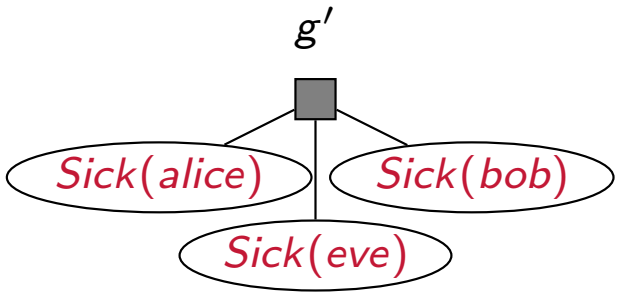
Elimination



Ondemand shattering

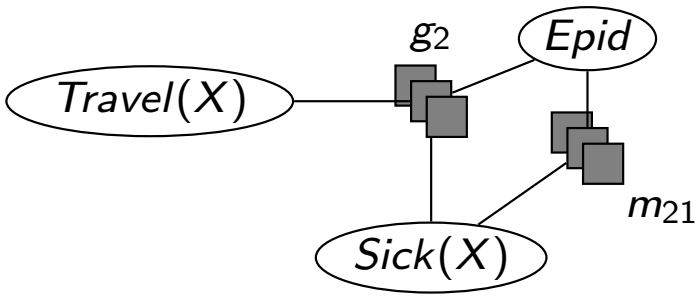


Elimination



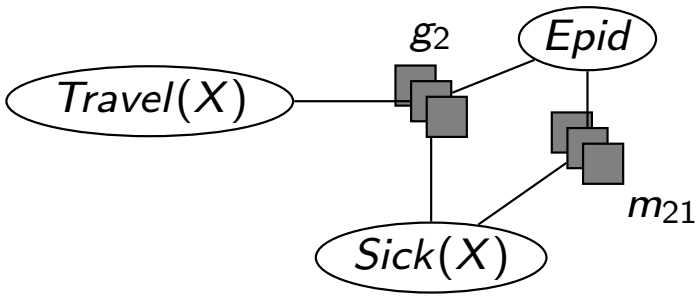
Contribution: Parameterised Queries

$P(Sick(X))|\top$

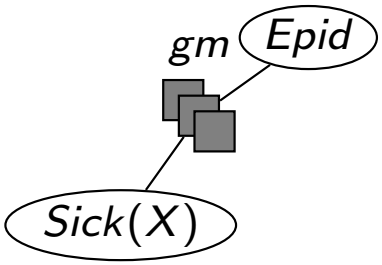


Contribution: Parameterised Queries

$P(Sick(X))|\top$

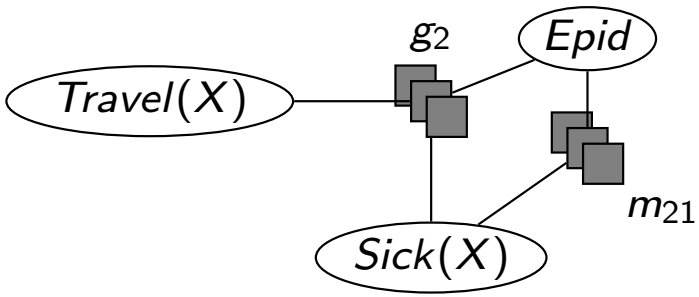


Elimination

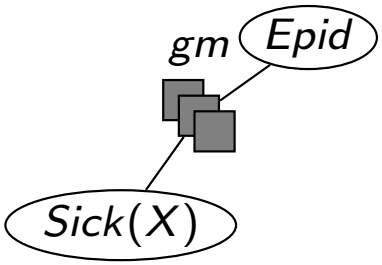


Contribution: Parameterised Queries

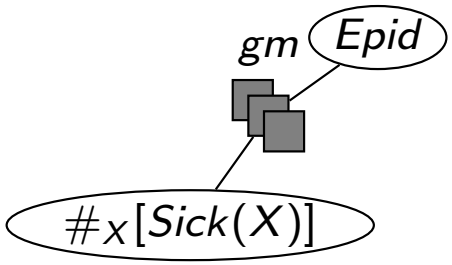
$P(Sick(X))|\top$



Elimination

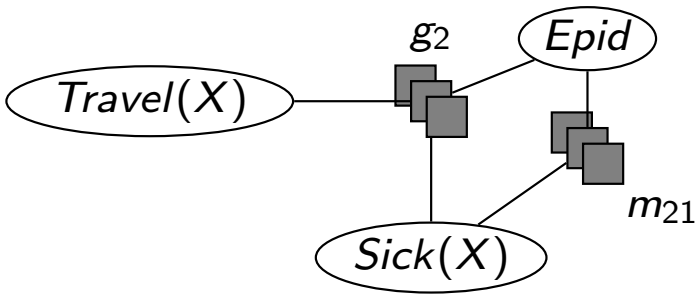


Counting conversion

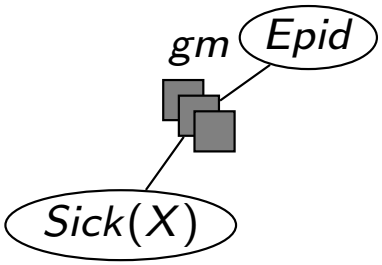


Contribution: Parameterised Queries

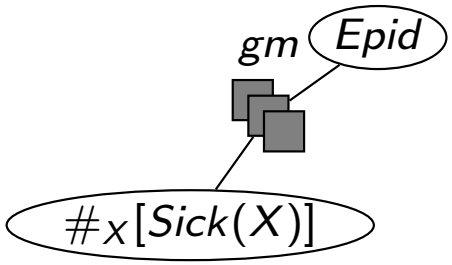
$P(Sick(X))|\top$



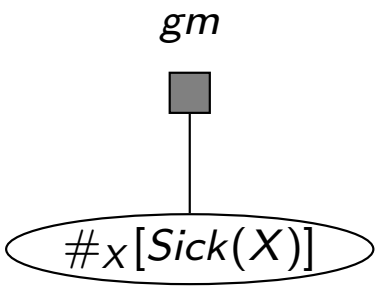
Elimination



Counting conversion

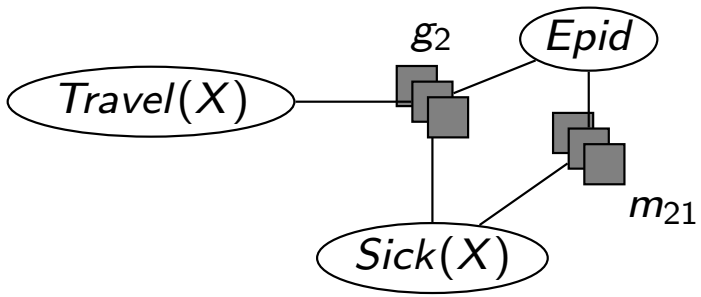


Elimination



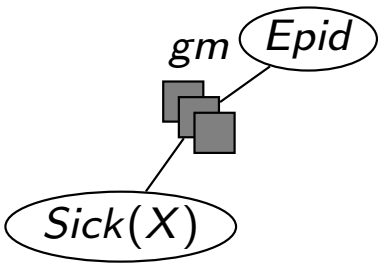
Contribution: Parameterised Queries

$P(Sick(X))|\top$

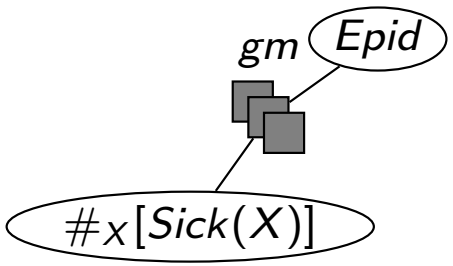


$\#_X[Sick(X)]$	g'
$[0, 3]$	1
$[1, 2]$	2
$[2, 1]$	3
$[3, 0]$	4

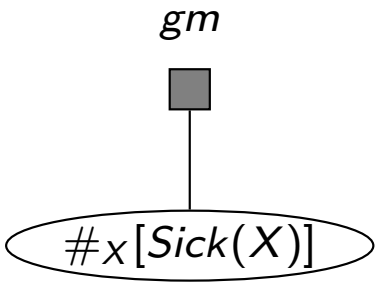
Elimination



Counting conversion



Elimination



Test Run

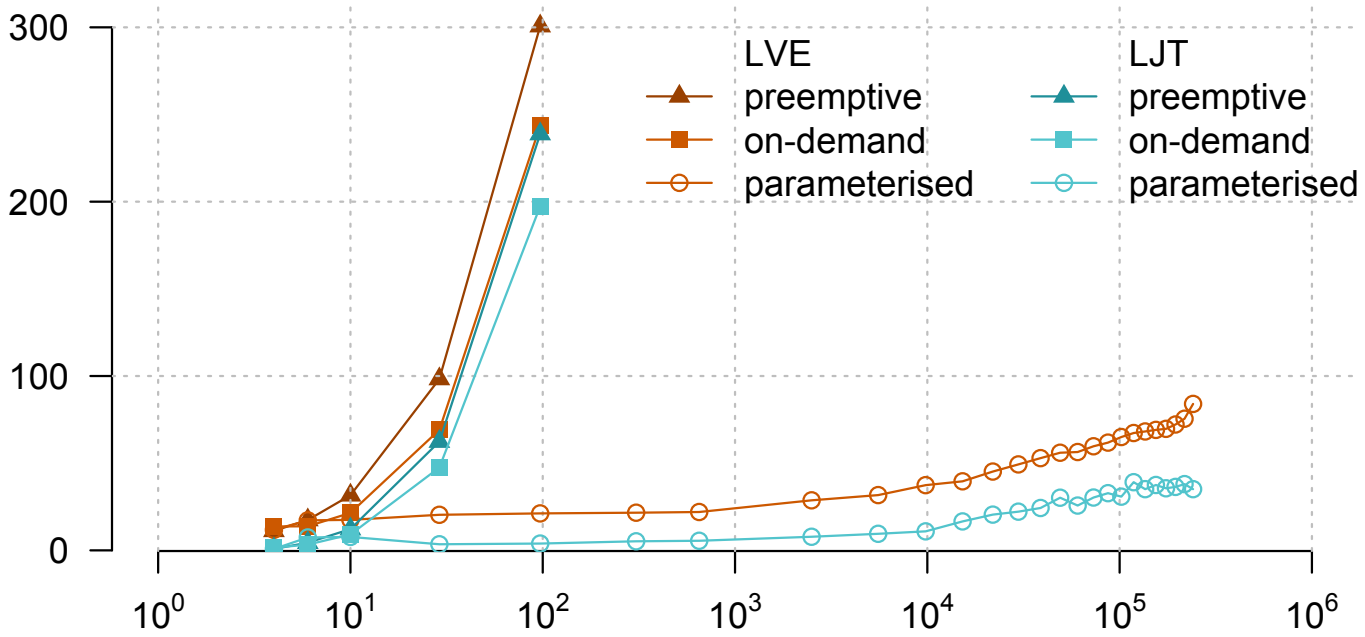


Figure: Runtimes [ms] for one query with grounded model sizes from 4 to 241,001 (points connected for readability)

`lve`: Implementation by Taghipour (2013), extended for on-demand shattering and parameterised queries

`ljt`: Our implementation of LJT

Analysis: LVE for Parameterised Queries (in LJT)

Algorithm steps

1. Eliminate non-query random variables
2. Count-convert remaining parameters
3. Normalise

Lifted query answering

Evidence for subset of query parameters

$\rightarrow$ Groups in result: $Sick(X_1), Sick(X_2)$

Correctness of calculations

$\rightarrow$ Splits of query parameter possible: $Sick(X)$ gets grounded

Limitations for parameterised queries

$\rightarrow$ Groundings unavoidable: $\phi(Sick(X), Treat(X, P), Eff(P))$

Conference Contribution

Parameterised Queries

- Compact query representation
  - with parameters
- Lifted query answering with LVE
  - exploiting counting conversions from LVE
- Compact result representation if possible
  - using counting random variables with histograms as range

Conference Contribution

Parameterised Queries

- Compact query representation
  - with parameters
- Lifted query answering with LVE
  - exploiting counting conversions from LVE
- Compact result representation if possible
  - using counting random variables with histograms as range

Current work

- Incrementally changing models
- Decision modelling