

# Conjunctive Queries on Probabilistic Graphs: Combined Complexity

---

Antoine Amarilli<sup>1</sup>, **Mikaël Monet**<sup>1,2</sup>, Pierre Senellart<sup>2,3</sup>

May 16th, 2017

<sup>1</sup>LTCI, Télécom ParisTech, Université Paris-Saclay; Paris, France

<sup>2</sup>Inria Paris; Paris, France

<sup>3</sup>École normale supérieure, PSL Research University; Paris, France

# Tuple-independent databases (TID)

- **Probabilistic databases:** model **uncertainty** about data
- Simplest model: **tuple-independent databases (TID)**
  - A **relational database**  $I$
  - A **probability valuation**  $\pi$  mapping each fact of  $I$  to  $[0, 1]$
- **Semantics** of a TID  $(I, \pi)$ : a **probability distribution** on  $I' \subseteq I$ :
  - Each fact  $F \in I$  is either **present** or **absent** with probability  $\pi(F)$
  - Assume **independence** across facts

## Example: TID

<b>S</b>		
<i>a</i>	<i>b</i>	<i>.5</i>
<i>a</i>	<i>c</i>	<i>.2</i>

## Example: TID

S		
<i>a</i>	<i>b</i>	.5
<i>a</i>	<i>c</i>	.2

This TID  $(I, \pi)$  represents the following **probability distribution**:

## Example: TID

<b>S</b>		
<i>a</i>	<i>b</i>	.5
<i>a</i>	<i>c</i>	.2

This TID  $(I, \pi)$  represents the following **probability distribution**:

<b>S</b>	
<i>a</i>	<i>b</i>
<i>a</i>	<i>c</i>

## Example: TID

<b>S</b>		
<i>a</i>	<i>b</i>	.5
<i>a</i>	<i>c</i>	.2

This TID  $(I, \pi)$  represents the following **probability distribution**:

$.5 \times .2$		$.5 \times (1 - .2)$	
<b>S</b>		<b>S</b>	
<i>a</i>	<i>b</i>	<i>a</i>	<i>b</i>
<i>a</i>	<i>c</i>		

## Example: TID

<b>S</b>		
<i>a</i>	<i>b</i>	.5
<i>a</i>	<i>c</i>	.2

This TID  $(I, \pi)$  represents the following **probability distribution**:

$.5 \times .2$	$.5 \times (1 - .2)$	$(1 - .5) \times .2$
<b>S</b>	<b>S</b>	<b>S</b>
<i>a</i> <i>b</i>	<i>a</i> <i>b</i>	
<i>a</i> <i>c</i>		<i>a</i> <i>c</i>

## Example: TID

<b>S</b>		
<i>a</i>	<i>b</i>	.5
<i>a</i>	<i>c</i>	.2

This TID  $(I, \pi)$  represents the following **probability distribution**:

$.5 \times .2$	$.5 \times (1 - .2)$	$(1 - .5) \times .2$	$(1 - .5) \times (1 - .2)$
<b>S</b>	<b>S</b>	<b>S</b>	<b>S</b>
<i>a</i> <i>b</i>	<i>a</i> <i>b</i>		
<i>a</i> <i>c</i>		<i>a</i> <i>c</i>	



# Probabilistic query evaluation (PQE)

Let us fix:

- **Relational signature**  $\sigma$
- Class  $\mathcal{I}$  of **relational instances** on  $\sigma$  (e.g., acyclic, treelike)
- Class  $\mathcal{Q}$  of **Boolean queries** (e.g., paths, trees)

# Probabilistic query evaluation (PQE)

Let us fix:

- **Relational signature**  $\sigma$
- Class  $\mathcal{I}$  of **relational instances** on  $\sigma$  (e.g., acyclic, treelike)
- Class  $\mathcal{Q}$  of **Boolean queries** (e.g., paths, trees)

**Probabilistic query evaluation** (PQE) problem for  $\mathcal{Q}$  and  $\mathcal{I}$ :

- Given a **query**  $q \in \mathcal{Q}$
- Given an **instance**  $I \in \mathcal{I}$  and a **probability valuation**  $\pi$
- Compute the **probability** that  $(I, \pi)$  satisfies  $q$

# Probabilistic query evaluation (PQE)

Let us fix:

- **Relational signature**  $\sigma$
- Class  $\mathcal{I}$  of **relational instances** on  $\sigma$  (e.g., acyclic, treelike)
- Class  $\mathcal{Q}$  of **Boolean queries** (e.g., paths, trees)

**Probabilistic query evaluation** (PQE) problem for  $\mathcal{Q}$  and  $\mathcal{I}$ :

- Given a **query**  $q \in \mathcal{Q}$
- Given an **instance**  $I \in \mathcal{I}$  and a **probability valuation**  $\pi$
- Compute the **probability** that  $(I, \pi)$  satisfies  $q$

$$\rightarrow \Pr((I, \pi) \models q) = \sum_{J \subseteq I, J \models q} \Pr(J)$$

# Complexity of probabilistic query evaluation (PQE)

**Question:** what is the (data, combined) **complexity** of PQE depending on the class  $\mathcal{Q}$  of **queries** and class  $\mathcal{I}$  of **instances**?

# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q} = \text{UCQs}$
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**

# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q} = \text{UCQs}$
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**
- PQE is **PTIME** for any  $q \in \mathcal{S}$

# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q} = \text{UCQs}$
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**
  - PQE is **PTIME** for any  $q \in \mathcal{S}$
  - PQE is **#P-hard** for any  $q \in \mathcal{Q} \setminus \mathcal{S}$

# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q} = \text{UCQs}$
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**
  - PQE is **PTIME** for any  $q \in \mathcal{S}$
  - PQE is **#P-hard** for any  $q \in \mathcal{Q} \setminus \mathcal{S}$
- Existing **data dichotomy result** on instances



# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q} = \text{UCQs}$
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**
  - PQE is **PTIME** for any  $q \in \mathcal{S}$
  - PQE is **#P-hard** for any  $q \in \mathcal{Q} \setminus \mathcal{S}$
- Existing **data dichotomy result** on instances
  - PQE for **MSO** on **bounded-treewidth** instances has **linear** data complexity [Amarilli, Bourhis, & Senellart, 2015]

# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q}$  = UCQs
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**
  - PQE is **PTIME** for any  $q \in \mathcal{S}$
  - PQE is **#P-hard** for any  $q \in \mathcal{Q} \setminus \mathcal{S}$
- Existing **data dichotomy result** on instances
  - PQE for **MSO** on **bounded-treewidth** instances has **linear** data complexity [Amarilli, Bourhis, & Senellart, 2015]
  - There is an FO query for which PQE is **#P-hard** on **any** unbounded-treewidth graph family  $\mathcal{I}$  (under some assumptions) [Amarilli, Bourhis, & Senellart, 2016]

# Data complexity results

- Existing **data dichotomy result** on queries [Dalvi & Suciu, 2012]
  - $\mathcal{Q} = \text{UCQs}$
  - $\mathcal{I}$  is all instances
  - There is a class  $\mathcal{S} \subseteq \mathcal{Q}$  of **safe queries**
  - PQE is **PTIME** for any  $q \in \mathcal{S}$
  - PQE is **#P-hard** for any  $q \in \mathcal{Q} \setminus \mathcal{S}$
- Existing **data dichotomy result** on instances
  - PQE for **MSO** on **bounded-treewidth** instances has **linear** data complexity [Amarilli, Bourhis, & Senellart, 2015]
  - There is an FO query for which PQE is **#P-hard** on **any** unbounded-treewidth graph family  $\mathcal{I}$  (under some assumptions) [Amarilli, Bourhis, & Senellart, 2016]

What about **combined** complexity?

## Restrict to CQs on graph signatures

$\exists x y z t R(x, y) \wedge S(y, z) \wedge S(t, z)$

<b>R</b>		
<i>a</i>	<i>b</i>	.1
<i>b</i>	<i>c</i>	.1
<i>c</i>	<i>d</i>	.05
<i>d</i>	<i>a</i>	1.
<i>d</i>	<i>b</i>	.8

<b>S</b>		
<i>b</i>	<i>d</i>	.7

## Restrict to CQs on graph signatures

$$\exists x y z t R(x, y) \wedge S(y, z) \wedge S(t, z) \quad \rightarrow \quad x \xrightarrow{R} y \xrightarrow{S} z \xleftarrow{S} t$$

---

**R**

---

<i>a</i>	<i>b</i>	.1
<i>b</i>	<i>c</i>	.1
<i>c</i>	<i>d</i>	.05
<i>d</i>	<i>a</i>	1.
<i>d</i>	<i>b</i>	.8

---

---

**S**

---

<i>b</i>	<i>d</i>	.7
----------	----------	----

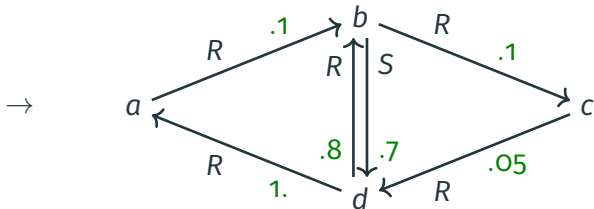
---

## Restrict to CQs on graph signatures

$$\exists xyz t R(x,y) \wedge S(y,z) \wedge S(t,z) \quad \rightarrow \quad x \xrightarrow{R} y \xrightarrow{S} z \xleftarrow{S} t$$

R		
a	b	.1
b	c	.1
c	d	.05
d	a	1.
d	b	.8

S		
b	d	.7



## Restrict instances to trees

$\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = polytrees (PT)

## Restrict instances to trees

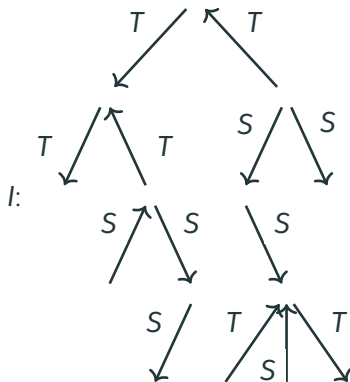
$\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = polytrees (PT)

Q:  $\xrightarrow{T} \xrightarrow{S} \xrightarrow{S} \xrightarrow{S} \xrightarrow{T}$



# Restrict instances to trees

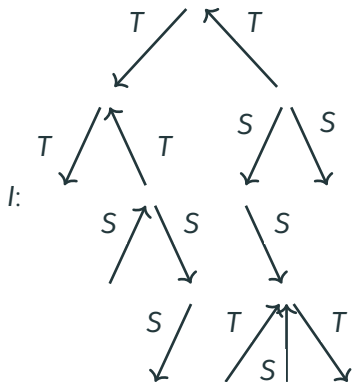
$\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = polytrees (PT)



+ prob. for each edge

# Restrict instances to trees

$\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = polytrees (PT)



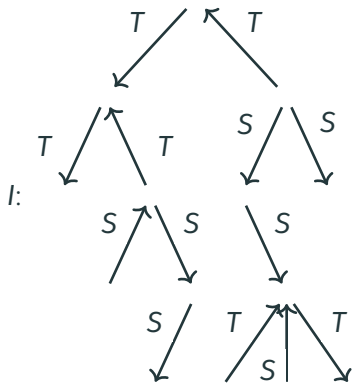
## Proposition

PQE of 1WP on PT is **#P-hard**

+ prob. for each edge

# $\mathcal{Q} = \text{one-way paths}, \mathcal{I} = \text{polytrees, without labels}$

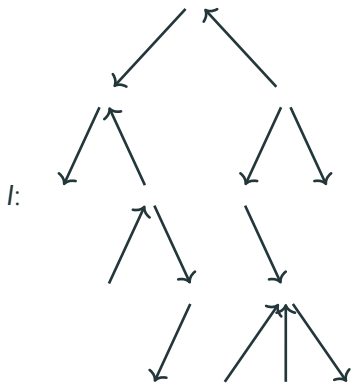
- What if we **do not have labels**?



+ prob. for each edge

# $\mathcal{Q}$ = one-way paths, $\mathcal{I}$ = polytrees, without labels

- What if we **do not have labels**?

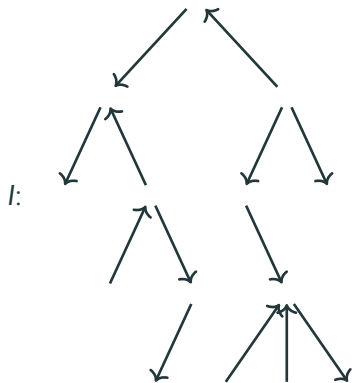


+ prob. for each edge



## $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$ , without labels

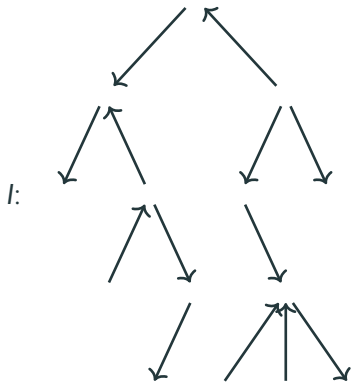
- What if we **do not have labels**?
- Probability that the instance graph has a path of length  $|Q|$
- **PTIME**: Bottom-up, e.g., tree automaton



+ prob. for each edge

## $Q = \textit{one-way paths}$ , $\mathcal{I} = \textit{polytrees}$ , without labels

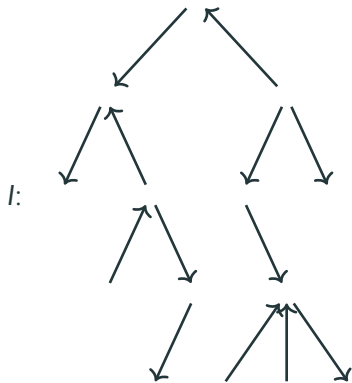
- What if we **do not have labels**?
- Probability that the instance graph has a path of length  $|Q|$
- **PTIME**: Bottom-up, e.g., tree automaton
- **Labels** have an impact!



+ prob. for each edge

# $\mathcal{Q}$ = two-way paths, $\mathcal{I}$ = polytrees, without labels

- $\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = polytrees (PT)

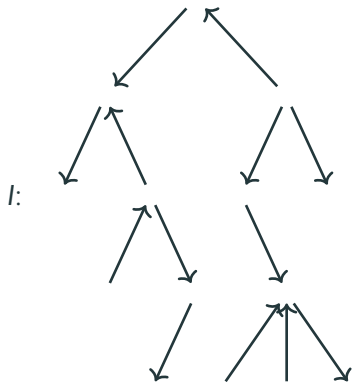


+ prob. for each edge



# $\mathcal{Q}$ = two-way paths, $\mathcal{I}$ = polytrees, without labels

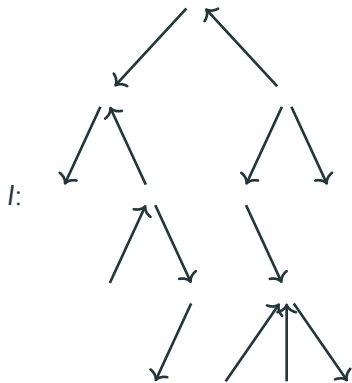
- $\mathcal{Q}$  = **two-way paths** (2WP),  $\mathcal{I}$  = **polytrees** (PT)



+ prob. for each edge

# $\mathcal{Q}$ = two-way paths, $\mathcal{I}$ = polytrees, without labels

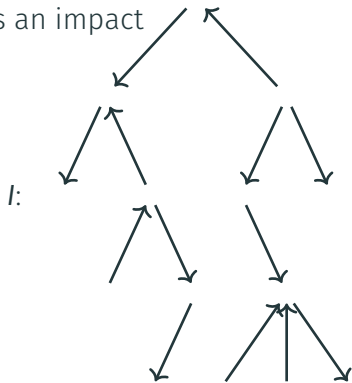
- $\mathcal{Q}$  = **two**-way paths (2WP),  $\mathcal{I}$  = **polytrees** (PT)
- **#P-hard**



+ prob. for each edge

# $\mathcal{Q}$ = two-way paths, $\mathcal{I}$ = polytrees, without labels

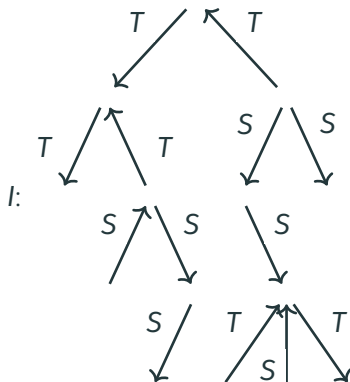
- $\mathcal{Q}$  = **two**-way paths (2WP),  $\mathcal{I}$  = **polytrees** (PT)
- **#P-hard**
- **Global orientation** of the **query** has an impact



+ prob. for each edge

$\mathcal{Q} = \text{one-way paths}, \mathcal{I} = \text{downwards trees}$

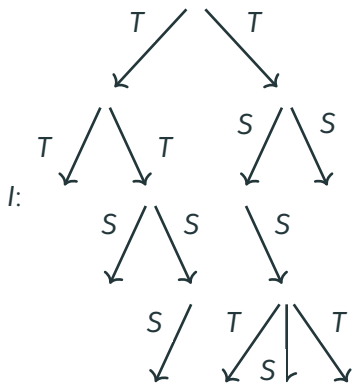
- $\mathcal{Q} = \text{one-way paths (1WP)}, \mathcal{I} = \text{polytrees (PT)}$



+ prob. for each edge

$\mathcal{Q} = \text{one-way paths}, \mathcal{I} = \text{downwards trees}$

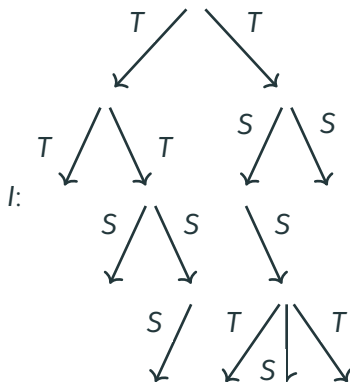
- $\mathcal{Q} = \text{one-way paths (1WP)}, \mathcal{I} = \text{downwards trees (DWT)}$



+ prob. for each edge

$\mathcal{Q} = \text{one-way paths}, \mathcal{I} = \text{downwards trees}$

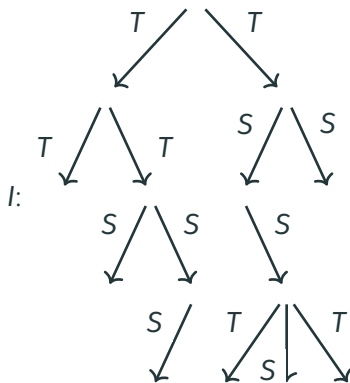
- $\mathcal{Q} = \text{one-way paths}$  (1WP),  $\mathcal{I} = \text{downwards trees}$  (DWT)
- **PTIME** also:  $\beta$ -acyclicity of the lineage



+ prob. for each edge

## $\mathcal{Q}$ = one-way paths, $\mathcal{I}$ = downwards trees

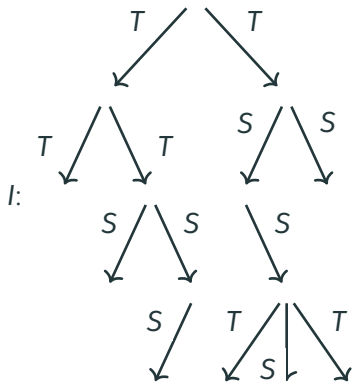
- $\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = **downwards trees** (DWT)
- **PTIME** also:  $\beta$ -acyclicity of the lineage
- **Global orientation** of the **instance** also has an impact!



+ prob. for each edge

$\mathcal{Q}$  = downwards trees,  $\mathcal{I}$  = downwards trees, with labels

- $\mathcal{Q}$  = one-way paths (1WP),  $\mathcal{I}$  = downwards trees

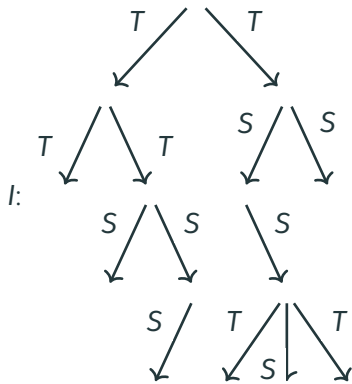
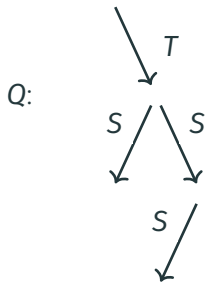


+ prob. for each edge



# $\mathcal{Q}$ = downwards trees, $\mathcal{I}$ = downwards trees, with labels

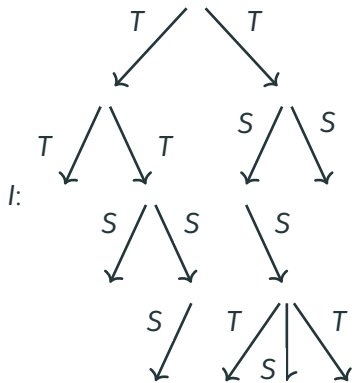
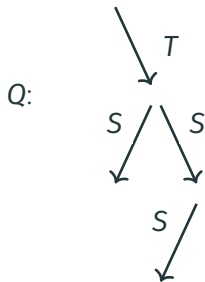
- $\mathcal{Q}$  = **downwards trees** (DWT),  $\mathcal{I}$  = downwards trees



+ prob. for each edge

# $\mathcal{Q}$ = downwards trees, $\mathcal{I}$ = downwards trees, with labels

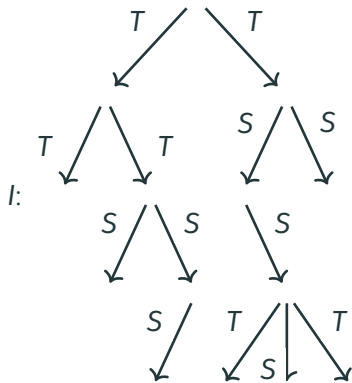
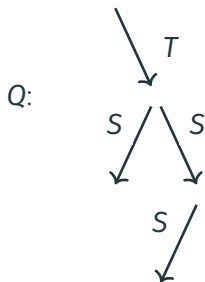
- $\mathcal{Q}$  = **downwards trees** (DWT),  $\mathcal{I}$  = **downwards trees**
- **#P-hard**



+ prob. for each edge

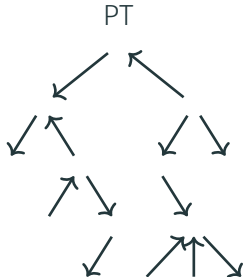
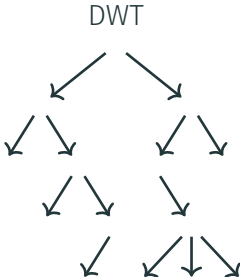
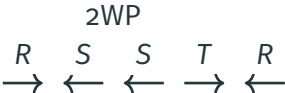
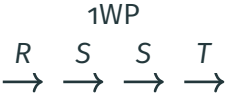
# $\mathcal{Q}$ = downwards trees, $\mathcal{I}$ = downwards trees, with labels

- $\mathcal{Q}$  = **downwards trees** (DWT),  $\mathcal{I}$  = **downwards trees**
- **#P-hard**
- **Branching** has an impact!



+ prob. for each edge

# Our graph classes



$1WP \subsetneq 2WP$   
 $1WP \subsetneq DWT$   
 $PT \subseteq \text{Connected} \subseteq \text{All}$

# Results

$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected
1WP						
2WP						
DWT			PTIME			
PT						
Connected						

$\geq 2$  labels

#P-hard

# Results

$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected	
1WP							$\geq 2$ labels
2WP							
DWT			PTIME				
PT						#P-hard	
Connected							
$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected	
1WP							No labels
2WP							
DWT			PTIME				
PT						#P-hard	
Connected							

# Results

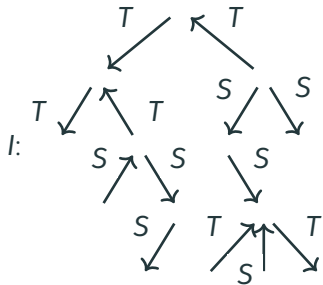
$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected	
1WP				•	•		$\geq 2$ labels
2WP				•			
DWT			PTIME	•			
PT						#P-hard	
Connected			•				
$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected	
1WP						•	No labels
2WP					•		
DWT			PTIME		•		
PT						#P-hard	
Connected			•	•			

# Results

$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected	
1WP					•		$\geq 2$ labels
2WP							
DWT			PTIME				
PT						#P-hard	
Connected							
$\downarrow Q$	$I \rightarrow$	1WP	2WP	DWT	PT	Connected	
1WP							No labels
2WP							
DWT			PTIME				
PT						#P-hard	
Connected							

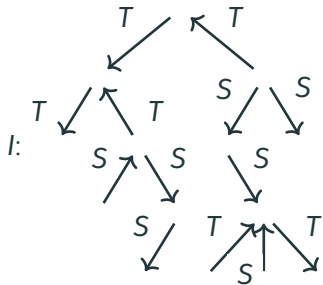


## Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$



+ prob. for each edge

## Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

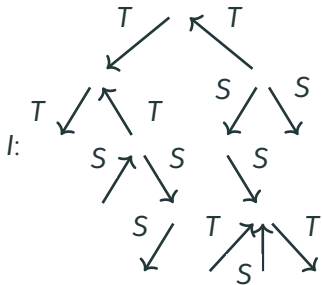


+ prob. for each edge

Reduction from **#P-hard** problem **#PP2DNF**:

- INPUT:** Boolean formula  $\varphi = \bigvee_{j=1 \dots m} (X_{x_j} \wedge Y_{y_j})$  on variables  $\{X_1, \dots, X_{n_1}\} \sqcup \{Y_1, \dots, Y_{n_2}\}$

## Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$



+ prob. for each edge

Reduction from **#P-hard** problem **#PP2DNF**:

- INPUT:** Boolean formula  $\varphi = \bigvee_{j=1 \dots m} (X_{x_j} \wedge Y_{y_j})$  on variables  $\{X_1, \dots, X_{n_1}\} \sqcup \{Y_1, \dots, Y_{n_2}\}$
- OUTPUT:** number of satisfying assignments of  $\varphi$

## Reduction for $\mathcal{Q} = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$

## Reduction for $\mathcal{Q} = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$

$I$ :

$Q$ :

## Reduction for $\mathcal{Q} = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$

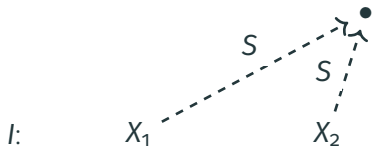
•

I:

Q:

## Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

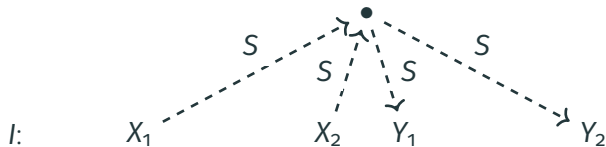
$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



Q:

## Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$

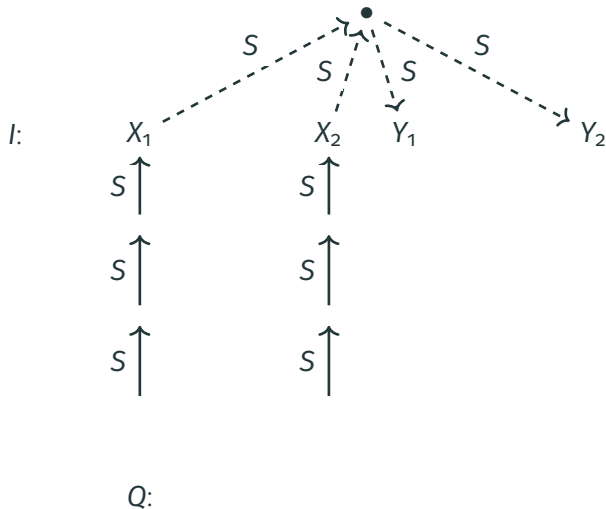


Q:



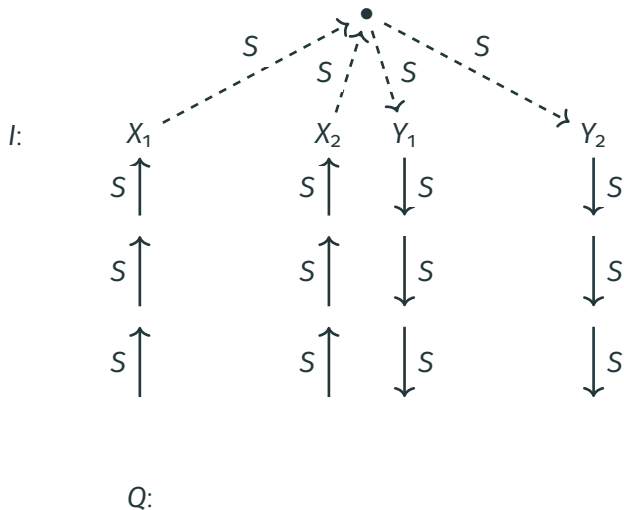
# Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



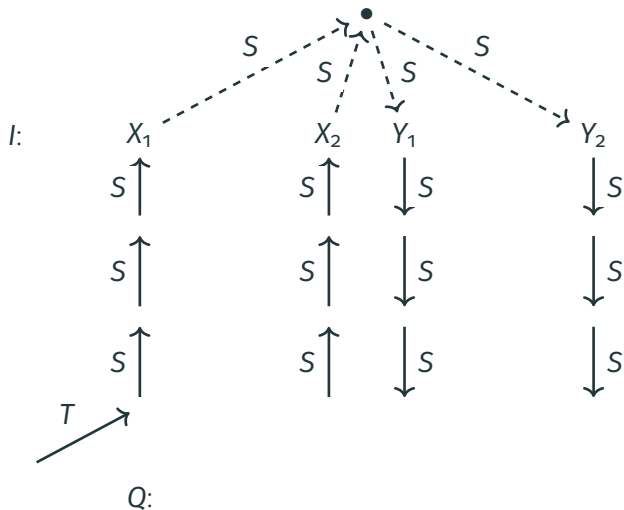
# Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



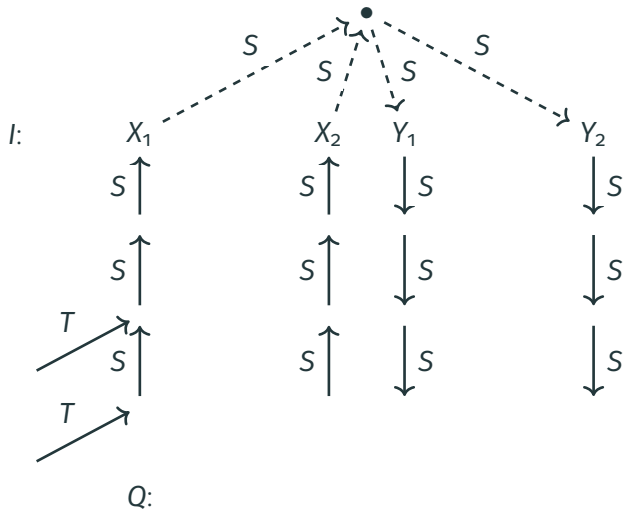
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



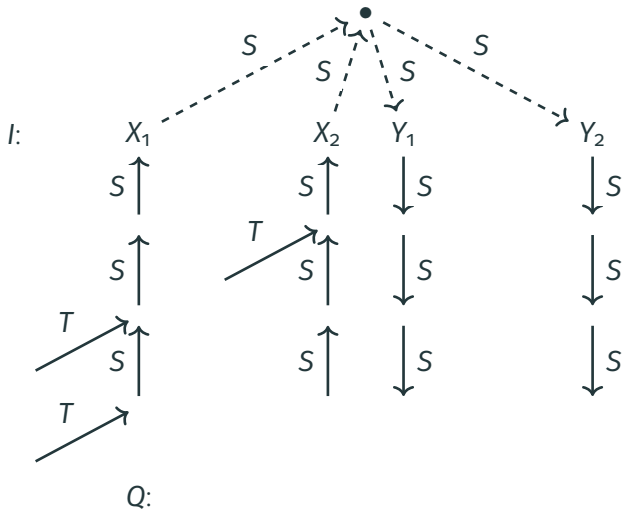
# Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



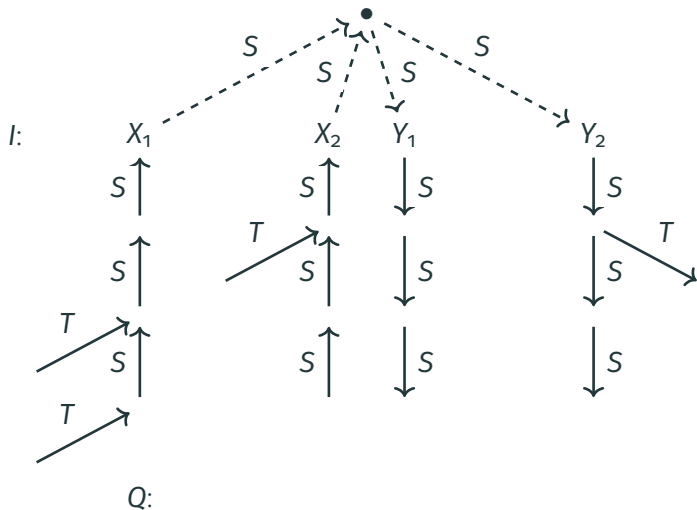
# Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



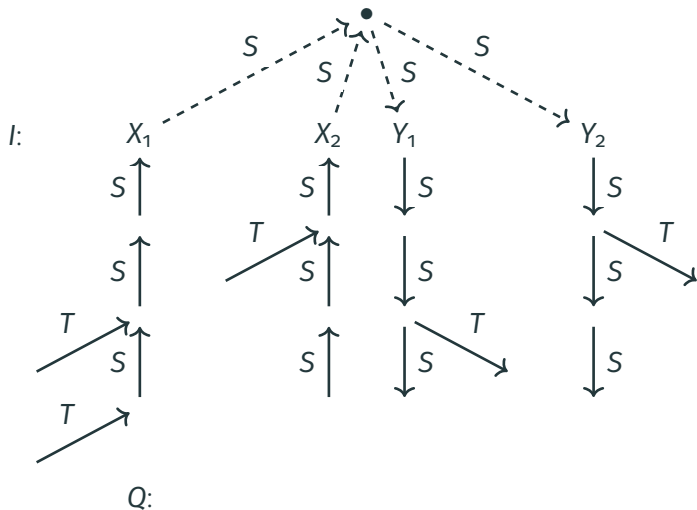
# Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



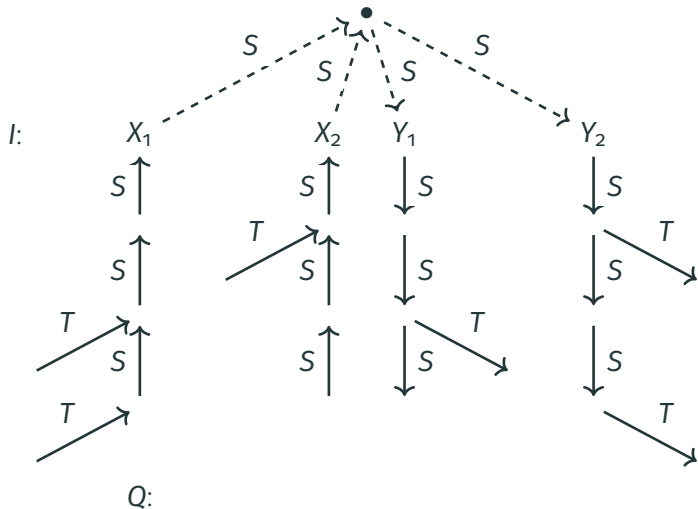
# Reduction for $Q = \text{one-way paths}$ , $\mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

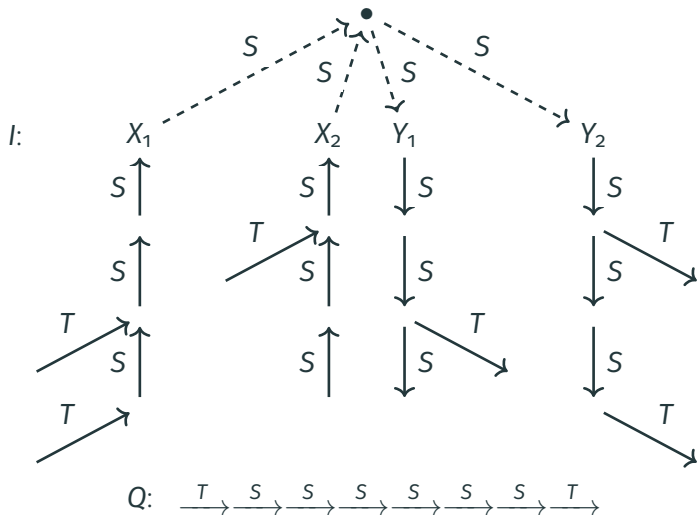
$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$





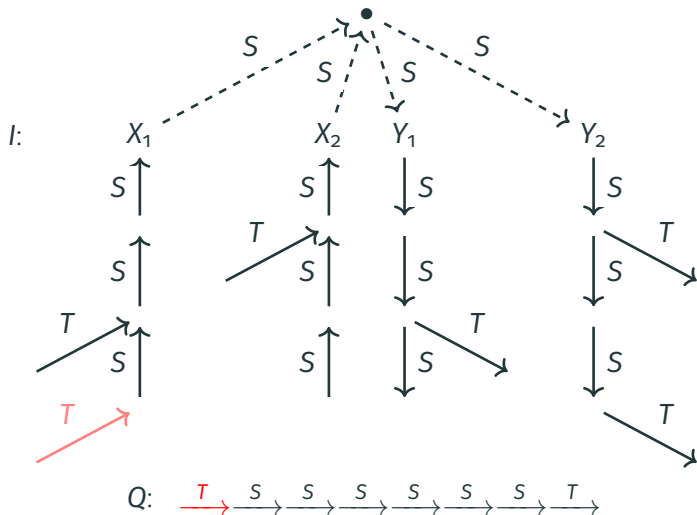
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



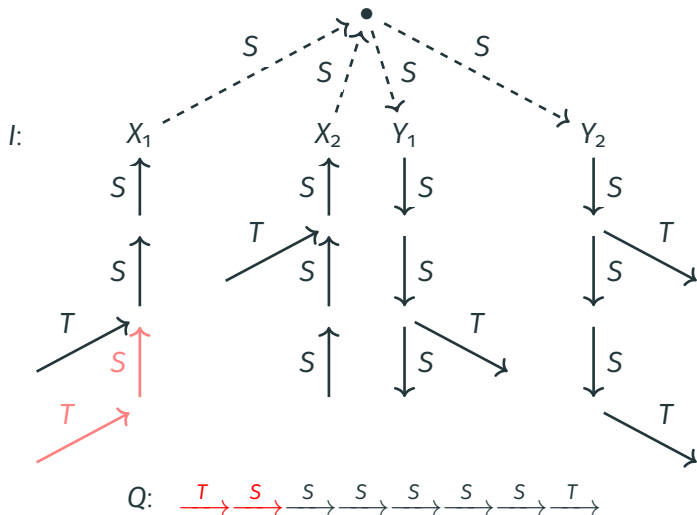
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



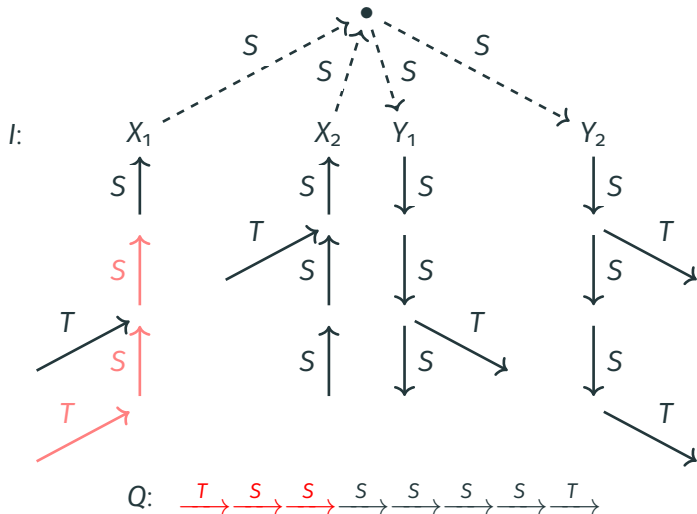
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



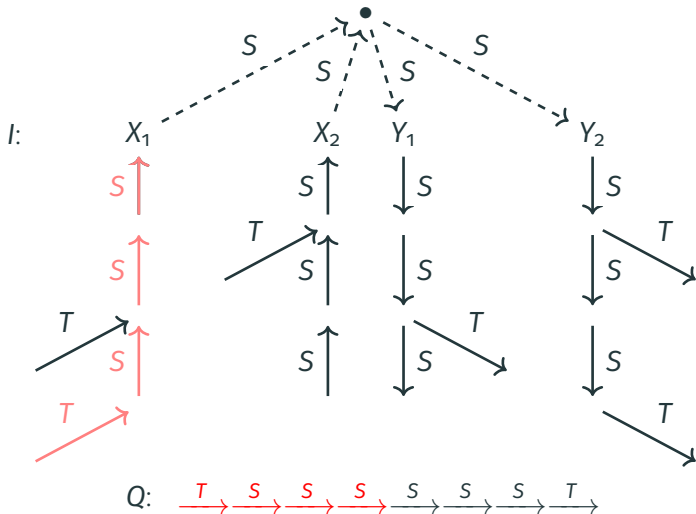
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



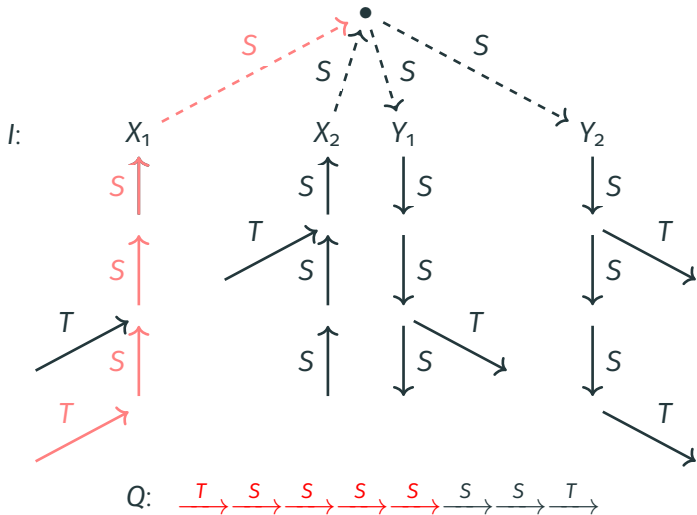
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



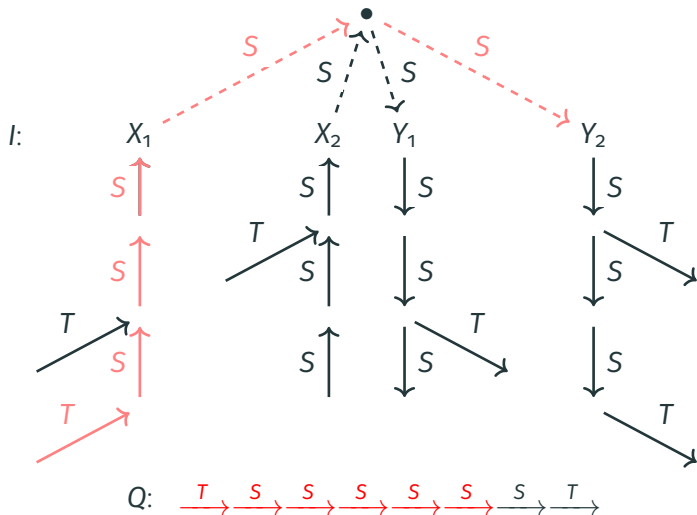
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



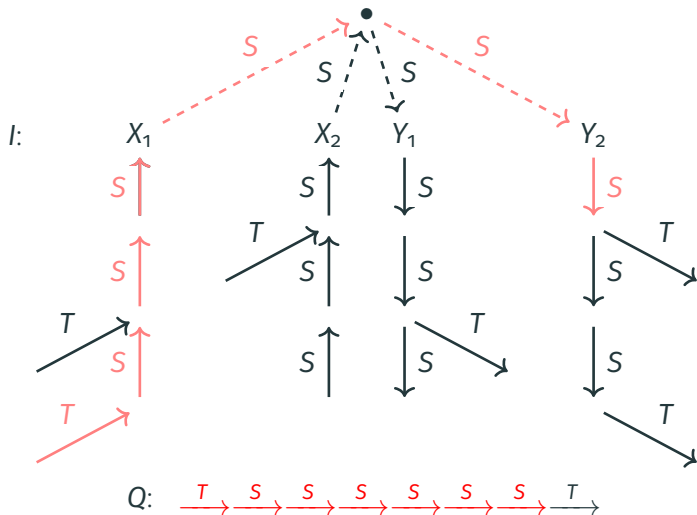
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

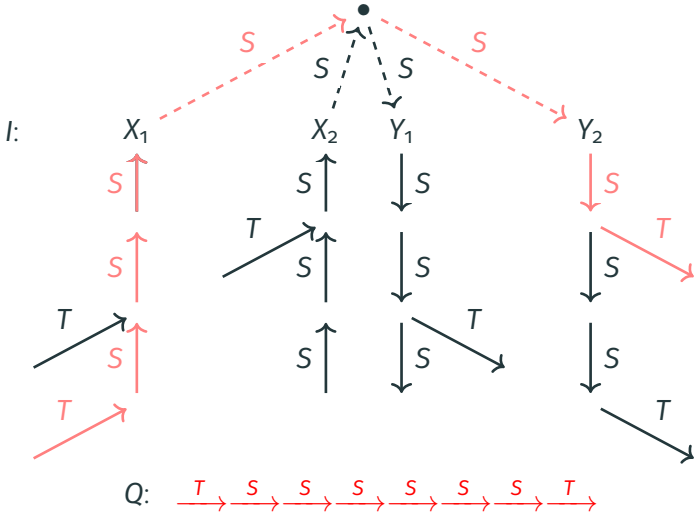
$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$





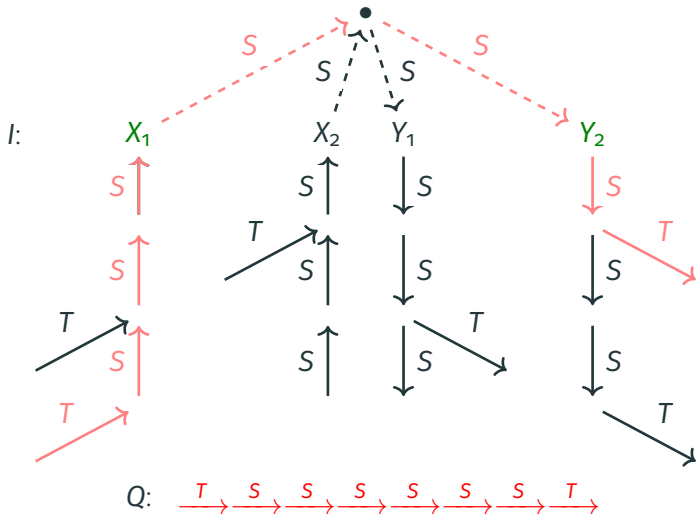
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



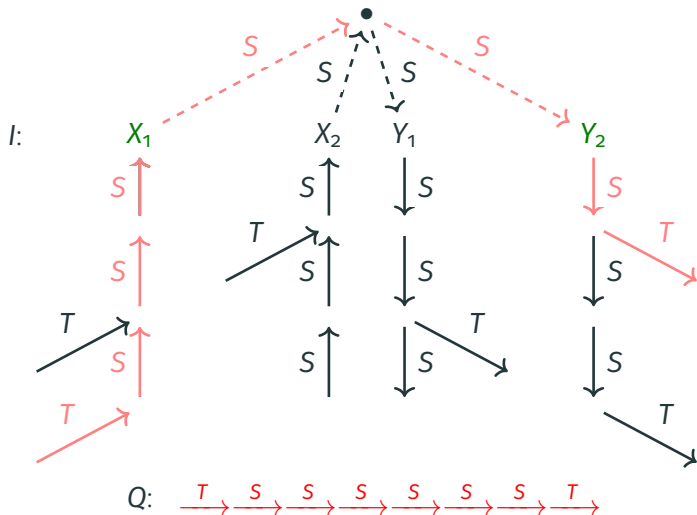
# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

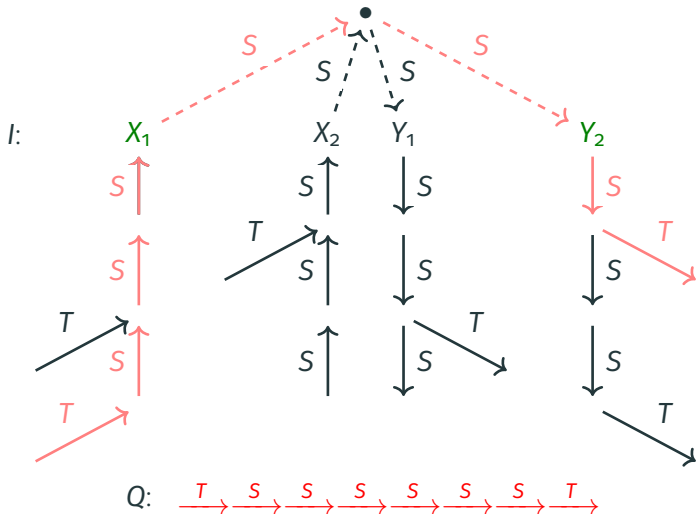
$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$



# Reduction for $Q = \text{one-way paths}, \mathcal{I} = \text{polytrees}$

$$\varphi = X_1 Y_2 \vee X_1 Y_1 \vee X_2 Y_2$$

$$\#\varphi = \Pr((I, \pi) \models Q) \times 2^{|\text{vars}(\varphi)|}$$



## Disconnected graphs

We also introduce the classes  $\sqcup 1WP$  (resp.,  $\sqcup 2WP$ ,  $\sqcup DWT$ ,  $\sqcup PT$ ) of graphs that are *disjoint unions of 1WP (resp., 2WP, DWT, PT)*

# Disconnected graphs

We also introduce the classes  $\sqcup 1WP$  (resp.,  $\sqcup 2WP$ ,  $\sqcup DWT$ ,  $\sqcup PT$ ) of graphs that are *disjoint unions* of  $1WP$  (resp.,  $2WP$ ,  $DWT$ ,  $PT$ )

$\downarrow G$	$H \rightarrow$	1WP	2WP	DWT	PT	Connected
$\sqcup 1WP$						No labels
$\sqcup 2WP$						
$\sqcup DWT$						
$\sqcup PT$						
All						

# Disconnected graphs

We also introduce the classes  $\sqcup 1WP$  (resp.,  $\sqcup 2WP$ ,  $\sqcup DWT$ ,  $\sqcup PT$ ) of graphs that are *disjoint unions* of  $1WP$  (resp.,  $2WP$ ,  $DWT$ ,  $PT$ )

$\downarrow G$	$H \rightarrow$	1WP	2WP	DWT	PT	Connected
$\sqcup 1WP$						No labels
$\sqcup 2WP$						
$\sqcup DWT$						
$\sqcup PT$						
All						

With labels, PQE of  $\sqcup 1WP$  on  $1WP$  is already **#P-hard!**

# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE



# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE
- Focus on CQs on arity-two signatures

# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE
- Focus on CQs on arity-two signatures
- Showed the importance of various features on the problem:  
**labels, global orientation, branching, connectedness**

# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE
- Focus on CQs on arity-two signatures
- Showed the importance of various features on the problem:  
**labels, global orientation, branching, connectedness**
- Established the complexity for all combinations of the graph classes we considered

# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE
- Focus on CQs on arity-two signatures
- Showed the importance of various features on the problem:  
**labels, global orientation, branching, connectedness**
- Established the complexity for all combinations of the graph classes we considered

## Drawbacks and future work:

- Our graph classes may seem “arbitrary”

# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE
- Focus on CQs on arity-two signatures
- Showed the importance of various features on the problem: **labels, global orientation, branching, connectedness**
- Established the complexity for all combinations of the graph classes we considered

## Drawbacks and future work:

- Our graph classes may seem “arbitrary”
- Not yet a dichotomy, just starting to understand the problem
- Practical applications?

# Conclusion

## Contributions:

- Detailed study of the **combined** complexity of PQE
- Focus on CQs on arity-two signatures
- Showed the importance of various features on the problem: **labels, global orientation, branching, connectedness**
- Established the complexity for all combinations of the graph classes we considered

## Drawbacks and future work:

- Our graph classes may seem “arbitrary”
- Not yet a dichotomy, just starting to understand the problem
- Practical applications?

Thanks for your attention!