Semantics, Undecidability, and Expressiveness of Predicate Logic

Bow-Yaw Wang

Institute of Information Science
Academia Sinica, Taiwan

October 15, 2018
Outline

1. Semantics of predicate logic
   - Models
   - Semantic entailment
   - Semantics of equality

2. Undecidability of predicate logic

3. Expressiveness of predicate logic
Let $\Gamma$ be a set of predicate logic formulae and $\psi$ a predicate logic formula. We know how to show $\Gamma \vdash \psi$.

Intuitively, $\psi$ “holds” when every formulae in $\Gamma$ hold.

What if we want to show $\Gamma \nvdash \psi$?

How do we show “there is no such deduction?”

Intuitively, we want to argue that $\psi$ does not hold even when every formulae in $\Gamma$ hold.

Hence we will discuss when predicate logic formulae “hold.”
1. **Semantics of predicate logic**
   - Models
   - Semantic entailment
   - Semantics of equality

2. **Undecidability of predicate logic**

3. **Expressiveness of predicate logic**
Models

- Recall that we have constant, function, and predicate symbols in predicate logic.
- The semantics of terms and atomic predicates are defined in models.

**Definition**

Let $\mathcal{F}$ and $\mathcal{P}$ be a set of function and predicate symbols respectively. A model $\mathcal{M}$ of $(\mathcal{F}, \mathcal{P})$ consists of

- A non-empty set $A$ called the universe;
- For function symbol $f \in \mathcal{F}$ with arity $n \geq 0$, a function $f^\mathcal{M} : A^n \rightarrow A$;
  - Particularly, a constant symbol $c \in \mathcal{F}$ is an element $c^\mathcal{M} \in A$.
- For predicate symbol $P \in \mathcal{P}$ with arity $n > 0$, a set $P^\mathcal{M} \subseteq A^n$. 
Example of Models

Let $\mathcal{F} = \{e, \cdot\}$ and $\mathcal{P} = \{\leq\}$ where $e$ is a constant, $\cdot$ a binary function, and $\leq$ a binary predicate symbol respectively. We use infix notation for $\cdot$ and $\leq$.

Consider the model $\mathcal{M}$:
- the universe $A$ is the set of all binary finite strings;
- $e^\mathcal{M}$ is the empty string $\epsilon$;
- $\cdot^\mathcal{M}$ is string concatenation;
- $\leq^\mathcal{M}$ is the string prefix relation.

For instance, $00 \cdot^\mathcal{M} 111 = 00111$ and $01 \leq^\mathcal{M} 011$.

In this model,
- $\forall x((x \leq x \cdot e) \land (x \cdot e \leq x))$ is true.
- $\exists y \forall x(y \leq x)$ is true.
- $\forall x \forall y \forall z((x \leq y) \implies (x \cdot z \leq y \cdot z))$ is false.
For the semantics of $\forall x \phi$ and $\exists x \phi$, we need to check whether $\phi$ is true when $x$ is assigned to an element of the universe.

A model $(\mathcal{F}, \mathcal{P})$ however does not give semantics to variables.

**Definition**

An environment for a universe $A$ is a function $l : \text{var} \rightarrow A$. If $l$ is an environment, $x \in \text{var}$, and $a \in A$, the environment $l[x \mapsto a]$ is defined as follows.

$$
l[x \mapsto a](y) = \begin{cases} 
a & \text{if } x = y \\
l(y) & \text{if } x \neq y
\end{cases}
$$
Definition

Let $\mathcal{M}$ be a model of $(\mathcal{F}, \mathcal{P})$, $l$ an environment, and $\phi$ a predicate logic formula. $\mathcal{M} \models_l \phi$ holds is defined as follows.

- $\mathcal{M} \models_l P(t_1, t_2, \ldots, t_n)$ holds if $(a_1, a_2, \ldots, a_n) \in P^\mathcal{M}$ where $a_1, a_2, \ldots, a_n \in A$ are computed for $t_1, t_2, \ldots, t_n$ by $\mathcal{F}$ and $l$;
- $\mathcal{M} \models_l \forall x \psi$ holds if $\mathcal{M} \models_{l[x \mapsto a]} \psi$ for every $a \in A$;
- $\mathcal{M} \models_l \exists x \psi$ holds if $\mathcal{M} \models_{l[x \mapsto a]} \psi$ for some $a \in A$;
- $\mathcal{M} \models_l \neg \psi$ holds if it is not the case $\mathcal{M} \models_l \psi$;
- $\mathcal{M} \models_l \psi_0 \lor \psi_1$ holds if $\mathcal{M} \models_l \psi_0$ holds or $\mathcal{M} \models_l \psi_1$ holds;
- $\mathcal{M} \models_l \psi_0 \land \psi_1$ holds if $\mathcal{M} \models_l \psi_0$ holds and $\mathcal{M} \models_l \psi_1$ holds;
- $\mathcal{M} \models_l \psi_0 \implies \psi_1$ holds if $\mathcal{M} \models_l \psi_1$ holds whenever $\mathcal{M} \models_l \psi_0$ holds.

If $\mathcal{M} \models_l \phi$ holds, we say $\phi$ computes to T in $\mathcal{M}$ with respect to $l$. Also, we write $\mathcal{M} \not\models_l \phi$ when it is not the case $\mathcal{M} \models_l \phi$. 
Let $\phi$ be a predicate logic formula, $l$ and $l'$ two environments that agree on free variables of $\phi$.

- That is, $l(x) = l'(x)$ for every free variable $x$ in $\phi$.

By induction on $\phi$, it is straightforward to show $\mathcal{M} \models _l \phi$ holds if and only if $\mathcal{M} \models _{l'} \phi$.

A sentence is a predicate logic formula without free variables.

Let $\phi$ be a sentence. Either

- $\mathcal{M} \models _l \phi$ holds for every environment $l$; or
- $\mathcal{M} \not\models _l \phi$ does not hold for every environment $l$.

Hence we write $\mathcal{M} \models \phi$ (or $\mathcal{M} \not\models \phi$) for a sentence $\phi$ since the choice of $l$ does not matter.
Example

- Consider \((\mathcal{F}, \mathcal{P}) = (\{\text{alma}\}, \{\text{loves}\})\) where alma is a constant and loves is a binary predicate.

- Let \(\mathcal{M}\) be a model of \((\mathcal{F}, \mathcal{P})\) with the universe \(A = \{a, b, c\}\), alma\(^\mathcal{M}\) = \(a\), and loves\(^\mathcal{M}\) = \(\{(a, a), (b, a), (c, a)\}\).

- Consider the statement:

  None of Alma’s lovers’ lovers love her.

- We first translate the statement into a predicate logic formula \(\phi\):

  \[ \forall x \forall y (\text{loves}(x, \text{alma}) \land \text{loves}(y, x) \implies \neg\text{loves}(y, \text{alma})). \]

- We have \(\mathcal{M} \not\models \phi\).
  - Choose \(a\) for \(x\) and \(b\) for \(y\). We have \((a, a) \in \text{loves}^\mathcal{M}\) and \((b, a) \in \text{loves}^\mathcal{M}\) but it is not the case \((b, a) \notin \text{loves}^\mathcal{M}\).
Outline

1. Semantics of predicate logic
   - Models
   - Semantic entailment
   - Semantics of equality

2. Undecidability of predicate logic

3. Expressiveness of predicate logic
**Definition**

Let $\Gamma$ be a (possibly infinite) set of predicate logic formulae and $\psi$ a predicate logic formula.

- $\Gamma \models \psi$ holds (or $\Gamma$ semantically entails $\psi$) if for every model $\mathcal{M}$ and environment $l$, $\mathcal{M} \models_l \psi$ holds whenever $\mathcal{M} \models_l \phi$ holds for every $\phi \in \Gamma$;

- $\psi$ is satisfiable if there is a model $\mathcal{M}$ and an environment $l$ such that $\mathcal{M} \models_l \psi$ holds;

- $\psi$ is valid if $\mathcal{M} \models_l \psi$ holds for every model $\mathcal{M}$ and environment $l$ where we can compute $\psi$;

- $\Gamma$ is consistent or satisfiable if there is a model $\mathcal{M}$ and an environment $l$ such that $\mathcal{M} \models_l \phi$ for every $\phi \in \Gamma$.

Note that “$\models$” has two different meanings:

- $\mathcal{M} \models \psi$ means “$\psi$ computes to $T$ in $\mathcal{M}$;”

- $\phi_1, \phi_2, \ldots, \phi_n \models \psi$ means “$\psi$ is semantically entailed by $\phi_1, \phi_2, \ldots, \phi_n$."

**Semantic Entailment**
Let $\psi, \phi_1, \phi_2, \ldots, \phi_n$ be sentences.

To check if $\mathcal{M} \models \psi$ holds, we need to enumerate all elements in the universe if $\psi$ contains $\forall$ or $\exists$.

To check if $\phi_1, \phi_2, \ldots, \phi_n \models \psi$ holds, we need to consider all possible models satisfying $\phi_1, \phi_2, \ldots, \phi_n$.

Both sound difficult since a model may contain an infinite number of elements in its universe.

However, we may still prove semantic entailments.
Example

Show $\forall x (P(x) \implies Q(x)) \models \forall x P(x) \implies \forall x Q(x)$.

Proof.

Let $\mathcal{M}$ be a model that $\mathcal{M} \models \forall x (P(x) \implies Q(x))$. There are two cases:

- $\mathcal{M} \not\models \forall x P(x)$. Then $\mathcal{M} \models \forall x P(x) \implies \forall x Q(x)$.

- $\mathcal{M} \models \forall x P(x)$. Let $a$ be an element in the universe of $\mathcal{M}$. We have $a \in P^\mathcal{M}$ since $\mathcal{M} \models \forall x P(x)$ and hence $a \in Q^\mathcal{M}$ since $\mathcal{M} \models \forall x (P(x) \implies Q(x))$. That is, $\mathcal{M} \models \forall x Q(x)$. We conclude $\mathcal{M} \models \forall x P(x) \implies \forall x Q(x)$. 

Bow-Yaw Wang (Academia Sinica) Semantics, Undecidability, and Expressiveness October 15, 2018 14 / 44
Example

Show $\forall x P(x) \implies \forall x Q(x) \not\equiv \forall x (P(x) \implies Q(x))$.

Proof.

Let $\mathcal{M}'$ be a model where $A' = \{a, b\}$, $P^{\mathcal{M}'} = \{a\}$, and $Q^{\mathcal{M}'} = \{b\}$. Since $\mathcal{M}' \not\models \forall x P(x)$, $\mathcal{M}' \models \forall x P(x) \implies \forall x Q(x)$. Since $a \in P^{\mathcal{M}'}$ but $a \not\in Q^{\mathcal{M}'}$, $\mathcal{M}' \not\models \forall x (P(x) \implies Q(x))$. 
Outline

1. Semantics of predicate logic
   - Models
   - Semantic entailment
   - Semantics of equality

2. Undecidability of predicate logic

3. Expressiveness of predicate logic
Observe that $=$ is also a binary predicate.

But the symbol “$=$” is somewhat special.
  - We did not say $= \in \mathcal{P}$.
  - Rather, we explicitly say that $=$ denotes the equality.

This is because we do not want to interpret the equality arbitrarily.
  - It sounds absurd if $a = b$ means $a$ is not $b$.

In all model $\mathcal{M}$, we always have $=^\mathcal{M} = \{(a, a) : a \in A\}$. 
Outline

1. Semantics of predicate logic
   - Models
   - Semantic entailment
   - Semantics of equality

2. Undecidability of predicate logic

3. Expressiveness of predicate logic
   - Existential second-order logic
   - Universal second-order logic
Validity Problem for Predicate Logic

**Definition**
Given a predicate logic formula $\phi$, the **validity problem** for predicate logic is to check whether $\models \phi$ holds or not.

- For a propositional logic formula $\phi$, it is decidable to check whether $\models \phi$ holds.
  - The validity problem for propositional logic is coNP-complete.
- For a predicate logic formula $\phi$, it is unclear how to design an algorithm.
- We will show the validity problem for predicate logic is undecidable.
Post Correspondence Problem

**Definition**

Given $C = ((s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k))$ where $s_i$, $t_i$ are non-empty binary strings for every $1 \leq i \leq k$. The **Post correspondence problem (PCP)** is to check whether there are $1 \leq i_1, i_2, \ldots, i_n \leq k$ such that $s_{i_1} s_{i_2} \cdots s_{i_n} = t_{i_1} t_{i_2} \cdots t_{i_n}$.

- For example, consider $C = ((1, 101), (10, 00), (011, 11))$. We have

  $$1\ 011\ 10\ 011 = 101\ 11\ 00\ 11.$$  

- The Post correspondence problem is undecidable.
  - For details, study **computational complexity**.
The validity problem for predicate logic is undecidable.

Proof.
Let $C = ((s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k))$ be an instance of PCP. We build a predicate logic formula $\phi$ so that $C$ has a solution iff $\models \phi$ holds.

Let $\mathcal{F} = \{e, f_0, f_1\}$ and $\mathcal{P} = \{P\}$. The function symbols $e, f_0(), f_1()$ encode binary strings. The binary predicate symbol $P(s, t)$ means “there are $i_1, i_2, \ldots, i_m$ so that $s = s_{i_1}s_{i_2}\ldots s_{i_m}$ and $t = t_{i_1}t_{i_2}\ldots t_{i_m}$.”

For instance, $1011 = f_1(f_1(f_0(f_1(e)))) = f_{1011}(e)$. Moreover, we write $f_{b_1b_2\ldots b_h}(v)$ for $f_{b_h}(f_{b_{h-1}}\ldots f_{b_1}(v))$ where $b_1b_2\ldots b_h$ is a binary string.
Proof (cont’d).

Define

\[ \phi_1 \triangleq k \bigwedge_{i=1}^{k} P(f_{s_i}(e), f_{t_i}(e)) \]
\[ \phi_2 \triangleq \forall v \forall w (P(v, w) \implies \bigwedge_{i=1}^{k} P(f_{s_i}(v), f_{t_i}(w))) \]
\[ \phi_3 \triangleq \exists z P(z, z) \]

We claim \( \models \phi_1 \land \phi_2 \implies \phi_3 \) iff \( C \) has a solution.

Suppose \( \models \phi_1 \land \phi_2 \implies \phi_3 \). Consider the model \( \mathcal{M} \) for \( (\mathcal{F}, \mathcal{P}) \) as follows.

The universe \( A \) is the set of all finite binary strings. \( e^\mathcal{M} \triangleq \epsilon, f_0^\mathcal{M}(s) \triangleq s_0 \), and \( f_1^\mathcal{M}(s) \triangleq s_1 \). Finally, \( P^\mathcal{M} = \{(s, t) : \text{there are } i_1, i_2, \ldots, i_m \text{ so that } s = s_{i_1}s_{i_2}\cdots s_{i_m} \text{ and } t = t_{i_1}t_{i_2}\cdots t_{i_m} \} \). We have \( \mathcal{M} \models \phi_1 \land \phi_2 \implies \phi_3 \). Moreover, since \( \mathcal{M} \models \phi_1 \) and \( \mathcal{M} \models \phi_2 \) (why?), \( \mathcal{M} \models \phi_3 \). That is, there is a binary string \( z \) and \( i_1, i_2, \ldots, i_n \) such that \( z = s_{i_1}s_{i_2}\cdots s_{i_n} = t_{i_1}t_{i_2}\cdots t_{i_n} \).
Proof (cont’d).

Conversely, suppose $C$ has a solution $i_1, i_2, \ldots, i_n$ that $s_{i_1}s_{i_2}\ldots s_{i_n} = t_{i_1}t_{i_2}\ldots t_{i_n}$. We need to show $M' \models \phi_1 \land \phi_2 \implies \phi_3$ for every model $M'$ defining $e^M$, $f_0^M$, $f_1^M$, and $P^M$. Clearly, $M' \models \phi_1 \land \phi_2 \implies \phi_3$ when $M' \not\models \phi_1 \land \phi_2$.

It suffices to consider $M' \models \phi_1 \land \phi_2$, and show $M' \models \phi_3$ as well.

Let $A'$ be the universe of $M'$. We interpret finite binary strings in $A'$ as follows.

- \[
\text{interpret}(\epsilon) \triangleq e^M,
\]
- \[
\text{interpret}(s0) \triangleq f_0^M(\text{interpret}(s)),
\]
- \[
\text{interpret}(s1) \triangleq f_1^M(\text{interpret}(s)).
\]

Hence, for instance, the string 1011 is interpreted as the element $f_1^M(f_1^M(f_0^M(f_1^M(e^M))))$. Generally, a finite binary string $s$ is interpreted as $f_s^M(e^M)$ in $A'$. 
Proof (cont’d).

Since $\mathcal{M}' \models \phi_1$, we have

$$(\text{interpret}(s_i), \text{interpret}(t_i)) \in P^{\mathcal{M}'} \text{ for } 1 \leq i \leq k.$$ 

Since $\mathcal{M}' \models \phi_2$, we have for every $(\text{interpret}(s), \text{interpret}(t)) \in P^{\mathcal{M}'}$

$$(\text{interpret}(ss_i), \text{interpret}(tt_i)) \in P^{\mathcal{M}'} \text{ for } 1 \leq i \leq k.$$ 

Thus,

$$(\text{interpret}(s_{i_1}s_{i_2}\ldots s_{i_n}), \text{interpret}(t_{i_1}t_{i_2}\ldots t_{i_n})) \in P^{\mathcal{M}'}.$$ 

Moreover, $s_{i_1}s_{i_2}\ldots s_{i_n} = t_{i_1}t_{i_2}\ldots t_{i_n}$ since $i_1, i_2, \ldots, i_n$ is a solution to $C$. Hence $\text{interpret}(s_{i_1}s_{i_2}\ldots s_{i_n}) = \text{interpret}(t_{i_1}t_{i_2}\ldots t_{i_n})$. In other words, $\mathcal{M}' \models \phi_3$. \qed
Corollary

The satisfiability problem for predicate logic is undecidable.

Proof.

Observe $\models \phi$ holds iff $\neg \phi$ is not satisfiable.

Theorem

For any predicate logic sentence $\phi$, $\vdash \phi$ iff $\models \phi$.

Corollary

It is undecidable to check whether $\vdash \phi$ for any predicate logic sentence $\phi$.

- The undecidability of provability problem for predicate logic means it is impossible to build a perfect automatic theorem prover.
- Just like art, human creativity is still important in mathematics!
Similar to propositional logic, the natural deduction proof system for predicate logic is both sound and complete. Proving completeness however is much harder for predicate logic.

▶ There is no truth table for predicate logic.

We will give the first step to establish completeness.
Lemma

Let $\Gamma$ be a set of predicate logic formulae. The following are equivalent:

1. $\Gamma \models \phi$ implies $\Gamma \vdash \phi$;
2. $\Gamma \models \bot$ implies $\Gamma \vdash \bot$.

Proof.

(1) to (2). Suppose $\Gamma \models \bot$. Then $\Gamma \vdash \bot$ by (1).

(2) to (1). Suppose $\Gamma \models \phi$. Then $\Gamma \cup \{\neg \phi\} \models \bot$. Hence $\Gamma \cup \{\neg \phi\} \vdash \bot$.

Therefore $\Gamma \vdash \phi$ using PBC.

To show completeness, it suffices to show that for every $\Gamma$, $\Gamma \not\models \bot$ implies $\Gamma$ is satisfiable.
We have two facts about predicate logic formulae. 
- \( \models \phi \) implies \( \vdash \phi \); and 
- it is undecidable to check if \( \vdash \phi \).

If a predicate logic formula is valid, then there is a natural deduction proof.

On the other hand, it is impossible to have a program which checks whether there is a natural deduction proof.
Outline

1. Semantics of predicate logic
2. Undecidability of predicate logic
3. Expressiveness of predicate logic
   - Existential second-order logic
   - Universal second-order logic
Example

Let $A = \{s_0, s_1, s_2, s_3\}$ and $R^M = \{(s_0, s_1), (s_1, s_0), (s_1, s_1), (s_1, s_2), (s_2, s_0), (s_3, s_0), (s_3, s_2)\}$. We write $s \rightarrow s'$ if $(s, s') \in R^M$, and say there is a transition from $s$ to $s'$.

Definition

Given a directed graph $G$ and nodes $n, n'$ in $G$, the reachability problem for $G$ is to check whether there is a path of transition from $n$ to $n'$. 
Let \((\mathcal{F}, \mathcal{P}) = (\emptyset, \{ R \})\) with a binary predicate \(R\).

A model of \((\mathcal{F}, \mathcal{P})\) denotes a directed graph.

Can we write a predicate logic formula \(\phi\) with free variables \(u\) and \(v\) to express \(u \rightarrow \cdots \rightarrow v\)?

Consider

\[
\begin{align*}
  u &= v \lor \\
  R(u, v) \lor \\
  \exists x_0 (R(u, x_0) \land R(x_0, v)) \lor \\
  \exists x_0 \exists x_1 (R(u, x_0) \land R(x_0, x_1) \land R(x_1, v)) \lor \cdots
\end{align*}
\]

But this is not a predicate logic formula since it is infinite.

We will show it is impossible to express reachability in predicate logic.
Compactness Theorem

**Theorem**

Let $\Gamma$ be a set of predicate logic sentences. If all finite subset of $\Gamma$ is satisfiable, $\Gamma$ is satisfiable.

**Proof.**

Assume $\Gamma$ is not satisfiable. Then $\Gamma \vDash \bot$. By the completeness theorem for predicate logic, $\Gamma \vDash \bot$. Since deductions are finite, we have $\Delta \vDash \bot$ for some finite subset $\Delta$ of $\Gamma$. By the soundness theorem for predicate logic, $\Delta \vDash \bot$. $\Delta$ is not satisfiable, a contraction.
Let $\psi$ be a predicate logic sentence. If $\psi$ has a model with at least $n$ elements for every $n \geq 1$, $\psi$ has a model with infinitely many elements.

Proof.

Define $\phi_n \triangleq \exists x_1 \exists x_2 \ldots \exists x_n \bigwedge_{1 \leq i < j \leq n} \neg (x_i = x_j)$. Let $\Gamma = \{ \psi \} \cup \{ \phi_n : n > 1 \}$. For every finite subset $\Delta$ of $\Gamma$, $\Delta$ is satisfiable. By the compactness theorem, $\Gamma$ is satisfiable by some model $M$. Particularly, $M \models \psi$ holds. Since $M \models \phi_n$ for every $n \geq 1$, $M$ has infinitely many elements.
Theorem

There is no predicate logic formula $\phi$ with exactly two free variables $u, v$ and exactly one binary predicate $R$ such that $\phi$ holds in directed graphs iff there is a path in the graph from the node associated with $u$ to the node associated with $v$.

Proof.

Suppose $\phi$ is a predicate logic formula expressing a path from $u$ to $v$. Let $c$ and $c'$ be constants. Define $\phi_0 \triangleq c = c'$ and

$$\phi_n \triangleq \exists x_1 \exists x_2 \cdots \exists x_{n-1} (R(c, x_1) \land R(x_1, x_2) \land \cdots \land R(x_{n-1}, c')) .$$

Then $\phi_n$ expresses that there is a path of length $n$ from $c$ to $c'$. Let $\Gamma = \{ \phi[c/u][c'/v] \} \cup \{ \neg \phi_i : i \geq 0 \}$. For every finite subset $\Delta$ of $\Gamma$, $\Delta$ is satisfiable since there is always a path of an arbitrary finite length from $c$ to $c'$. By the compactness theorem, $\Gamma$ is satisfiable. A contradiction. \qed
Outline

1. Semantics of predicate logic
2. Undecidability of predicate logic
3. Expressiveness of predicate logic
   - Existential second-order logic
   - Universal second-order logic
Existential Second-Order Logic

- In predicate logic, we can ask if there is an element with a certain property.
  - Predicate logic is also called first-order logic.

- We can generalize the concept and ask if there is a predicate with a certain property in existential second-order logic.

- Let $P$ be an $n$-ary predicate symbol.

- $\exists P \phi$ is an existential second-order logic formula.

- Let $\mathcal{M}$ be a model for all function and predicate symbols except $P$ and $\mathcal{M}_T$ the same model with an additional $n$-ary relation $T(= P^{\mathcal{M}_T}) \subseteq A^n$. Define

$$\mathcal{M} \models I \exists P \phi$$

if $\mathcal{M}_T \models I \phi$ for some $T(= P^{\mathcal{M}_T}) \subseteq A^n$. 

Bow-Yaw Wang (Academia Sinica)
Semantics, Undecidability, and Expressiveness
October 15, 2018 36 / 44
Consider the existential second-order logic formula
\[ \exists P \forall x \forall y \forall z (C_1 \land C_2 \land C_3 \land C_4) \]
where
\[
C_1 \triangleq P(x, x) \\
C_2 \triangleq P(x, y) \land P(y, z) \implies P(x, z) \\
C_3 \triangleq P(u, v) \implies \bot \\
C_4 \triangleq R(x, y) \implies P(x, y).
\]
\[ \quad \text{\textbullet ~ } C_i \text{'s are Horn clauses.} \]
Consider the directed graph $\mathcal{M}$ in the previous slide.

Let $l(u) = s_0$ and $l(v) = s_3$.

Does $\mathcal{M} \models I \exists P \forall x \forall y \forall z (C_1 \land C_2 \land C_3 \land C_4)$ hold?

- Take $T \triangleq \{(s, s') \in A \times A : s' \neq s_3\} \cup \{(s_3, s_3)\}$. 
Outline

1. Semantics of predicate logic
2. Undecidability of predicate logic
3. Expressiveness of predicate logic
   - Existential second-order logic
   - Universal second-order logic
Let $P$ be an $n$-ary predicate symbol.

$\forall P \phi$ is a universal second-order logic formula.

Let $\mathcal{M}$ be a model for all function and predicate symbols except $P$. Define

$$\mathcal{M} \models I \forall P \phi \text{ if } \mathcal{M}_T \models I \phi \text{ for every } T (= P^{\mathcal{M}_T}) \subseteq A^n.$$
**Theorem**

Let $\mathcal{M}$ be a model of $(\emptyset, \{R\})$ with a binary predicate symbol $R$. $\mathcal{M} \models I \forall P \exists x \exists y \exists z (\neg C_1 \lor \neg C_2 \lor \neg C_3 \lor \neg C_4)$ holds iff $l(v)$ is $R$-reachable from $l(u)$ in $\mathcal{M}$, where $C_1 \triangleq P(x, x), C_2 \triangleq P(x, y) \land P(y, z) \implies P(x, z), C_3 \triangleq P(u, v) \implies \bot$, and $C_4 \triangleq R(x, y) \implies P(x, y)$.

**Proof.**

Assume $\mathcal{M}_T \models I \exists x \exists y \exists z (\neg C_1 \lor \neg C_2 \lor \neg C_3 \lor \neg C_4)$ for every $T \subseteq A \times A$. Consider the reflexive and transitive closure $T^*$ of $R^\mathcal{M}$. Then $\mathcal{M}_{T^*} \models I' C_1 \land C_2 \land C_4$ where $I' = I[x, y, z \mapsto a, b, c]$ for some $a, b, c \in A^{\mathcal{M}_T}$. Hence $\mathcal{M}_{T^*} \models I' \neg C_3$ and so $\mathcal{M}_{T^*} \models I' P(u, v)$. In other words, $(I'(u), I'(v)) = (I(u), I(v)) \in T^*$. There is a finite path from $l(u)$ to $l(v)$. 
Proof (cont’d).

Conversely, assume there is a finite path from \( l(u) \) to \( l(v) \). Let \( T \subseteq A \times A \). There are two cases.

- **\( T \) is not reflexive, not transitive, or does not contain \( R^M \).** Then \( M_T \models l' \neg C_1, M_T \models l' \neg C_2 \), or \( M_T \models l' \neg C_4 \) for some \( l' = l[x, y, z \mapsto a, b, c] \) for some \( a, b, c \in A^M_T \).

- **\( T \) is reflexive, transitive, and contains \( R^M \).** Then \( T \) contains the reflexive, transitive closure of \( R^M \). Note that \((l(u), l(v))\) is in the reflexive, transitive closure of \( R^M \). Hence \( M_T \models l' \neg C_3 \).

In all cases, we have \( M_T \models \exists x \exists y \exists z (\neg C_1 \lor \neg C_2 \lor \neg C_3 \lor \neg C_4) \).

Reachability is in fact expressible in existential second-order logic.
Given an existential second-order logic formula $\phi$, whether there is an existential second-order logic formula $\psi$ such that $\psi$ and $\neg \phi$ are equivalent is an open problem.
If we allow both quantifiers in a formula, we get second-order logic.

For instance, \( \exists P \forall Q(\forall x \forall y(Q(x, y) \implies Q(y, x)) \implies \forall u \forall v(Q(u, v) \implies P(u, v))) \) is a second-order logic sentence.

Furthermore, if we allow quantifiers over relations of relations, we get third-order logic.

Designing higher-order logic need be careful.

- Nice properties such as compactness and completeness often fail.
- Soundness theorem can also fail!
  
  Consider \( A \triangleq \{ x : x \notin x \} \).

Many theorem provers (Coq, Isabelle, HOL etc) are in fact based on higher-order logics.