A few examples

Formalize in first-order logic:

- Some people love each other.
  \[ \exists x \exists y (\ell(x, y) \land \ell(y, x)) \]

- If two people are in love, then they are spouses.
  \[ \forall x \forall y (\ell(x, y) \land \ell(y, x) \Rightarrow s(x) = y \land s(y) = x) \]

- No one can love two distinct persons.
  \[ \forall x \forall y (\ell(x, y) \Rightarrow \forall z (\ell(x, z) \Rightarrow y = z)) \]
  \[ \forall x \forall y \forall z (\ell(x, y) \land \ell(x, z) \Rightarrow y = z) \]
Overview

Interpretation (contd.): meaning of formulae

Interpretation and substitution

Finite interpretation

Important equivalences

Conclusion
Overview

Interpretation (contd.): meaning of formulae

Interpretation and substitution

Finite interpretation

Important equivalences

Conclusion
Reminders

- $D$ is a non-empty domain.
- $I$ is an interpretation of every symbol in a formula as:
  - constants ($\in D$)
  - functions ($D^n \rightarrow D$)
  - propositional variables ($\in \{0, 1\}$)
  - relations ($\subseteq D^n$)
Example 4.3.29

Let us consider the following signature:

- \(\text{Anne}^f_0\), \(\text{Bernard}^f_0\) and \(\text{Claude}^f_0\): constants
- \(\ell^{r_2}\): a binary relation (we read \(\ell(x, y)\) as “\(x\) loves \(y\)”)
- \(s^{f_1}\): a unary function (we read \(s(x)\) as the spouse of \(x\)).

Let \(I\) be the interpretation of domain \(D = \{0, 1, 2\}\) where:

- \(\text{Anne}_I^f = 0\), \(\text{Bernard}_I^f = 1\), and \(\text{Claude}_I^f = 2\).
- \(\ell_I^{r_2} = \{(0, 1), (1, 0), (2, 0)\}\).
- \(s_I^{f_1}(0) = 1\), \(s_I^{f_1}(1) = 0\), \(s_I^{f_1}(2) = 2\).
Truth value of a formula 4.3.30

1. Propositional connectives have the same meaning as in propositional logic.

2. Let $e[x = d]$ be the state that is identical to $e$, except for $x$.

$$\left[ \forall x B \right]_{l,e} = \min_{d \in D} [B]_{l,e[x=d]} = \prod_{d \in D} [B]_{l,e[x=d]};$$

i.e. it is true if and only if $[B]_{(l,f)} = 1$ for every state $f$ identical to $e$, except for $x$.

3. 

$$\left[ \exists x B \right]_{l,e} = \max_{d \in D} [B]_{l,e[x=d]} = \sum_{d \in D} [B]_{l,e[x=d]};$$

i.e. it is true if there is a state $f$ identical to $e$, except for $x$, such that $[B]_{(l,f)} = 1$. 
Example 4.3.32

Let us use the interpretation $I$ given in example 4.3.19.
(Reminder $D = \{0, 1, 2\}$)

\[\exists x \, \ell(x, x) \] _I =
\[= \max\{[\ell(0, 0)]_I, [\ell(1, 1)]_I, [\ell(2, 2)]_I\} = \text{false}\]
\[= [\ell(0, 0)]_I + [\ell(1, 1)]_I + [\ell(2, 2)]_I = \text{false} + \text{false} + \text{false} = \text{false}.\]

\[\forall x \, \exists y \, \ell(x, y) \] _I =
\[= \min\\{\max\{\max\{[\ell(0, 0)]_I, [\ell(0, 1)]_I, [\ell(0, 2)]_I\},
\max\{[\ell(1, 0)]_I, [\ell(1, 1)]_I, [\ell(1, 2)]_I\},
\max\{[\ell(2, 0)]_I, [\ell(2, 1)]_I, [\ell(2, 2)]_I\}\}\]
\[= \min\{\max\{\text{false, true, false}\}, \max\{\text{true, false, false}\},
\max\{\text{true, false, false}\}\}\]
\[= \min\{\text{true, true, true}\} = \text{true}.\]
Example 4.3.32

\[ \exists y \forall x \, \ell(x, y) \]  

\[ = [\ell(0, 0)]_I \cdot [\ell(1, 0)]_I \cdot [\ell(2, 0)]_I + [\ell(0, 1)]_I \cdot [\ell(1, 1)]_I \cdot [\ell(2, 1)]_I \]

\[ + [\ell(0, 2)]_I \cdot [\ell(1, 2)]_I \cdot [\ell(2, 2)]_I \]

\[ = \text{false}.\text{true}.\text{true} + \text{true}.\text{false}.\text{false} + \text{false}.\text{false}.\text{false} \]

\[ = \text{false} + \text{false} + \text{false} = \text{false}. \]

Remark 4.3.33

The formulae \( \forall x \exists y \, \ell(x, y) \) and \( \exists y \forall x \, \ell(x, y) \) do not have the same value. Exchanging a \( \exists \) and a \( \forall \) does not preserve the truth value of a formula.
Model, validity, consequence, equivalence

Defined as in propositional logic but...

What’s needed to evaluate a formula

- **In propositional logic**: an assignment \( V \rightarrow \{0, 1\} \)
- **In first-order logic**: \((I, e)\) where
  - \(I\) is a symbol interpretation
  - \(e\) a variable state.

... we use an interpretation instead of an assignment. The truth value of a formula only depends on
- the state of its free variables
- and the interpretation of its symbols.
Overview

Interpretation (contd.): meaning of formulae

Interpretation and substitution

Finite interpretation

Important equivalences

Conclusion
Substitution at the propositional level

Recall that substituting a propositional variable in a valid formula yields another valid formula. This extends to first-order logic.

Example:

Let $\sigma(p) = \forall x \ q(x)$.

$p \lor \neg p$ is valid, the same holds for

$$\sigma(p \lor \neg p) = \forall x \ q(x) \lor \neg \forall x \ q(x)$$

The replacement principle extends to first-order logic as well since:

For any formulae $A$ and $B$ and any variable $x$:

- $(A \iff B) \models (\forall x A \iff \forall x B)$
- $(A \iff B) \models (\exists x A \iff \exists x B)$
Instantiation of a variable in a term

Definition 4.3.34

\( A < x := t > \) is the formula obtained by replacing in \( A \) every free occurrence of \( x \) with the term \( t \).

Example 4.3.35

Let \( A \) be the formula \( (\forall x P(x) \lor Q(x)) \), the formula \( A < x := b > \) is \( (\forall x P(x) \lor Q(b)) \) since only the bold occurrence of \( x \) is free.

But one cannot substitute any variable with anything:

Example 4.3.37

Let \( A \) be the formula \( \exists y p(x, y) \).

\[ A < x := y > = \exists y p(y, y) \] (capture phenomenon)
Capture changes the meaning of a formula

Example 4.3.37

Let $p$ be a binary relation interpreted over $\{0, 1\}$ as $p_I = \{(0, 1)\}$

Let $e$ be a state where $y = 0$.

- $[A < x := y >]_{(I, e)} =$
  
  $[\exists y p(y, y)]_{(I, e)} = [p(0, 0)]_{(I, e)} + [p(1, 1)]_{(I, e)} = false + false = false$.

- Let $d = 0$.
  
  In the assignment $(I, e[x = d])$, we have $x = 0$.
  Hence $[A]_{(I, e[x=d])} =$

  $[\exists y p(x, y)]_{(I, e[x=d])} = [p(0, 0)]_{(I, e)} + [p(0, 1)]_{(I, e)} = false + true = true$.

Thus, $[A < x := y >]_{(I, e)} \neq [A]_{(I, e[x=d])}$, for $d = [y]_{(I, e)}$. 
Precautions for the instantiation of a variable in a term

Solution: notion of a term \( t \) free for a variable

**Definition 4.3.34**

2. The term \( t \) is free for \( x \) in \( A \) if the variables of \( t \) are not bound in the free occurrences of \( x \).

**Example 4.3.35**

- The term \( f(z) \) is free for \( x \) in formula \( \exists y \, p(x, y) \).
- On the opposite, the term \( y \) is not free for \( x \) in this formula.
- By definition, the term \( x \) is free for \( x \) in any formula.
Properties

Theorem 4.3.36
Let \( A \) be a formula and \( t \) a free term for the variable \( x \) in \( A \).
For any assignment \((I, e)\) we have
\[
[A < x := t >]_{(I, e)} = [A]_{(I, e[\{x=d\})}
\]
where \( d = [t]_{(I, e)} \).

Corollary 4.3.38
Let \( A \) be a formula and \( t \) a free term for \( x \) in \( A \).
The formulae \( \forall x A \Rightarrow A < x := t > \) and \( A < x := t > \Rightarrow \exists x A \) are valid.
Overview

Interpretation (contd.): meaning of formulae

Interpretation and substitution

Finite interpretation

Important equivalences

Conclusion
Finite model

Definition

A finite model of a closed formula is an interpretation of the formula of finite domain, which makes the formula true.

Remark

- The name of the elements of the domain is not important.
- Hence for a model with $n$ elements, we’ll use the domain of integers less than $n$. 
Building a finite model

**Naive idea**: In order to know whether a closed formula has a model of domain \( \{0, \ldots, n - 1\} \), just

- **enumerate** all the possible interpretations of the associated signature of the formula
- **evaluate** the formula for these interpretations.

**Example**

Let \( \Sigma = \{a^0, f^1, P^{r2}\} \), plus possibly the equality.

Over a domain of 5 elements, \( \Sigma \) has \( 5 \times 5^5 \times 2^{25} \) interpretations!

This method is **unreadable** in practice.
Software for building a finite model

MACE

- translation of first-order formulae in propositional formulae
- performant algorithms to find the satisfiability of a propositional formula (e.g., different versions of the DPLL algorithm)

http://www.cs.unm.edu/~mccune/mace4

An actual example:
Method for finding a finite model

We look for models with $n$ elements by reduction to the propositional case.

**Base case:** a formula with no function symbol and no constant, except representations of integers less than $n$.

**Building the $n$-elements model**

1. eliminate the quantifiers by expansion over a domain of $n$ elements,
2. replace equalities with their value
3. search for a propositional assignment of atomic formulae which is a model of the formula.
Expansion of a formula

Definition 4.3.39

The $n$-expansion of $A$ consists in replacing:

- all sub-formula of $A$ of the form $\forall x B$ with the conjunction
  $$\bigwedge_{i < n} B < x := i >$$
- all sub-formula of $A$ of the form $\exists x B$ with the disjunction
  $$\bigvee_{i < n} B < x := i >$$

where $i$ is a term interpreted as the integer $i$.

Example 4.3.40

The 2-expansion of the formula $\exists x P(x) \Rightarrow \forall x P(x)$ is

$$P(0) + P(1) \Rightarrow P(0).P(1)$$
Property of the $n$-expansion

**Theorem 4.3.41**

Let $A$ be a formula containing only integers $< n$.
Let $B$ be the $n$-expansion of $A$.
Any interpretation over the domain $\{0, \ldots, n-1\}$ assigns the same value to $A$ and $B$.

Proof: by induction on the height of formulae.
From the assignment to the interpretation

Let $A$ be a formula:
- closed,
- with no quantifier,
- with no equality nor function symbol,
- with no constant except the representations of integers less than $n$.

Let $P$ be the set of atomic formulae in $A$ (except $\top$ and $\bot$).

**Theorem 4.3.42**

For any propositional assignment $\nu : P \to \{false, true\}$ there exists an interpretation $I$ of $A$ such that $[A]_I = [A]_{\nu}$. 
Example 4.3.43

Let \( \nu \) be the assignment defined by \([p(0)]_\nu = true\) and \([p(1)]_\nu = false\).

\( \nu \) gives the value \( false \) to the formula \((p(0) + p(1)) \Rightarrow (p(0) \cdot p(1))\).

The interpretation \( I \) defined by \( p_I = \{0\} \) gives the same value to the same formulae.

This example shows that \( \nu \) and \( I \) are two analogous ways of presenting an interpretation.
From the interpretation to the assignment

Let $A$ be a closed formula, with no quantifier, no equality, no function symbol, no constant except for the representations of integers $< n$. Let $P$ be the set of atomic formulae in $A$.

**Theorem 4.3.44**

For any interpretation $I$ there exists an assignment $\nu : P \rightarrow \{false, true\}$ such that

$$[A]_I = [A]_\nu.$$
Finding a finite model of a closed formula without function symbol

Procedure under the same hypotheses.

1. Replace $A$ by its $n$-expansion $B$
2. In $B$,
   - replace equalities by their truth value
     \((i = j \text{ is true iff } i \text{ and } j \text{ are identical})\)
   - Apply the usual simplifications

Let $C$ be the obtained formula.
3. Look for a propositional assignment $\nu$ of the atomic formulae of $C$, which is a model of $C$. 
Correctness of the method

\[
\begin{align*}
A & \rightarrow B & \rightarrow C & \rightarrow C \\
(1\text{st order}) & \equiv_n (\text{without } \forall \exists) & \equiv (\text{without const.}) & \simeq (\text{propos.})
\end{align*}
\]

- \( [A]_I = [B]_I \) for any \( I \) over a domain of \( n \) elements
- \( B \equiv C \) by construction (hence \( [B]_I = [C]_I \) for any \( I \))
  - For any \( v \) there is an \( I \) such that \( [C]_I = [C]_v \).
  - For any \( I \) there is a \( v \) such that \( [C]_I = [C]_v \).

Thus \( A \) has a model \( I \) over a domain of \( n \) elements if and only if \( C \) has a model \( v \) (and we can find \( I \) from \( v \) if need be).
Example 4.3.45

\[ A = \exists x P(x) \land \exists x \neg P(x) \land \forall x \forall y (P(x) \land P(y) \Rightarrow x = y) \]

A has no model of one element, since we have \( \exists x P(x) \) and \( \exists x \neg P(x) \).

2-expansion of A

\[
(P(0) + P(1)). (\overline{P(0)} + \overline{P(1)}). (P(0).P(0) \Rightarrow 0 = 0).(P(0).P(1) \Rightarrow 0 = 1).
(P(1).P(0) \Rightarrow 1 = 0).(P(1).P(1) \Rightarrow 1 = 1)
\]

We replace equalities by their values

\[
(P(0) + P(1)). (\overline{P(0)} + \overline{P(1)}).
(P(0).P(0) \Rightarrow \top). (P(0).P(1) \Rightarrow \bot). (P(1).P(0) \Rightarrow \bot). (P(1).P(1) \Rightarrow \top).
\]

Which simplifies to \((P(0) + P(1)).(\overline{P(0)} + \overline{P(1)})\)

The assignment \( P(0) \mapsto true, \ P(1) \mapsto false \) is a propositional model of that, hence the interpretation \( I \) of domain \( \{0, 1\} \) where \( P_I = \{0\} \) is a model of \( A \).
Finding a finite model of a closed formula with a function symbol

Let $A$ be a closed formula which can contain representations of integers of value less than $n$.

Procedure:
- Replace $A$ by its expansion
- Enumerate the choices of symbol values, by propagating as much as possible each of the realized choices.

Similar to DPLL algorithm.
Example 4.3.46 : \( A = \exists y P(y) \Rightarrow P(a) \)

Look for a counter-model with 2 elements.

2-expansion of \( A \)

\[
P(0) + P(1) \Rightarrow P(a)
\]

Find the values of \( P(0), P(1), a \).
We (arbitrarily) choose \( a = 0 \).

\[
P(0) + P(1) \Rightarrow P(0)
\]

\( P(0) \leftrightarrow false, P(1) \leftrightarrow true \) is a propositional counter-model, we deduce an interpretation \( I \) such that \( P_I = \{1\} \).

A counter-model is \( I \) over domain \( \{0, 1\} \) such that \( P_I = \{1\} \) and \( a_I = 0 \).
Example 4.3.47 : $P(a), \forall x(P(x) \Rightarrow P(f(x))), \neg P(f(b))$

1. 2-expansion:
   
   $$F = \{ P(a), (P(0) \Rightarrow P(f(0))), (P(1) \Rightarrow P(f(1))), \neg P(f(b)) \}.$$ 

2. Find values for $P(0), P(1), a, b, f(0)$ and $f(1)$ which provide a model of $F$.

3. Let us choose $a = 0$
   
   ▶ From $P(a) = true$ and $a = 0$, we deduce: $P(0) = true$
   
   ▶ From $P(0) = true$ and $(P(0) \Rightarrow P(f(0))) = true$, we deduce: $P(f(0)) = true$
   
   ▶ From $P(f(b)) = false$ and $P(f(0)) = true$, we deduce $f(0) \neq f(b)$ therefore $b \neq 0$, hence: $b = 1$ and $P(f(1)) = false$.
   
   ▶ From $P(f(1)) = false$ and $P(0) = true$, we deduce $f(1) \neq 0$ hence: $f(1) = 1$ and $P(1) = false$
   
   ▶ From $P(f(0)) = true$ and $P(1) = false$, we deduce: $f(0) = 0$

4. Model: $a = 0, b = 1, P = \{0\}, f(0) = 0, f(1) = 1$
Overview

Interpretation (contd.): meaning of formulae

Interpretation and substitution

Finite interpretation

Important equivalences

Conclusion
Relation between $\forall$ and $\exists$

Lemma 4.4.1

Let $A$ be a formula and $x$ be a variable.

1. $\neg \forall x A \equiv \exists x \neg A$
2. $\forall x A \equiv \neg \exists x \neg A$
3. $\neg \exists x A \equiv \forall x \neg A$
4. $\exists x A \equiv \neg \forall x \neg A$

Let us prove the first two equivalences, the other are in exercise 78
Proof of $\neg \forall xA \equiv \exists x \neg A$

Let us evaluate $[\neg \forall xA](l,e)$

- $= 1 - [\forall xA](l,e)$
- $= 1 - \min_{d \in D} [A](l,e[x=d])$
- $= \max_{d \in D} (1 - [A](l,e[x=d]))$
- $= \max_{d \in D} [\neg A](l,e[x=d])$
- $= [\exists x \neg A](l,e)$

Proof of $\forall xA \equiv \neg \exists x \neg A$:

Let us evaluate $\forall xA$

- $\equiv \neg \neg \forall xA$
- $\equiv \neg \exists x \neg A$ (see above)
Moving quantifiers

Let $x, y$ be two variables and $A, B$ be two formulae.

1. $\forall x \forall y A \equiv \forall y \forall x A$
2. $\exists x \exists y A \equiv \exists y \exists x A$
3. $\forall x (A \land B) \equiv (\forall x A \land \forall x B)$
4. $\exists x (A \lor B) \equiv (\exists x A \lor \exists x B)$

5. Let $Q$ be a quantifier and let $\circ$ be a connective among $\land$, $\lor$.
   If $x$ is not a free variable of $A$ then:
   5.1 $Qx A \equiv A$,
   5.2 $Qx(A \circ B) \equiv A \circ QxB$
Example 4.4.2

Let us eliminate useless quantifiers from these two formulae:

1. \( \forall x \exists x P(x) \equiv \exists x P(x) \)
2. \( \forall x (\exists x P(x) \lor Q(x)) \equiv \exists x P(x) \lor \forall x Q(x) \)
Renaming of bound variables (1/3)

Theorem 4.4.3

Let $Q$ be a quantifier. If $y$ does not occur in $Qx\ A$ then:

$$Qx\ A \equiv Qy\ A < x := y >.$$

Example 4.4.4

$\forall x\ p(x, z) \equiv \forall y\ p(y, z)$

$\forall x\ p(x, z) \not\equiv \forall z\ p(z, z)$
Overview

Interpretation (contd.): meaning of formulae

Interpretation and substitution

Finite interpretation

Important equivalences

Conclusion
Today

- To evaluate a formula = to choose an interpretation for its **symbols** and a state for its **variables**
- Method for finding **(counter-)model** by finite interpretation and expansion
- Important equivalences about quantifiers
  (beware, **no usable notion of normal form**)
Next time

- Skolemisation
- Semi-algorithm to prove that a formula is unsatisfiable.

Homework

*Every man is mortal.*
*Socrates is a man.*
*Hence Socrates is mortal.*

- Look for a counter-model using 1-expansion then 2-expansion.
- What is your conclusion?