## syn.1 Variable Assignments

fol:syn:ass:sec

A variable assignment $s$ provides a value for *every* variable—and there are infinitely many of them. This is of course not necessary. We require variable assignments to assign values to all variables simply because it makes things a lot easier. The value of a term $t$, and whether or not a formula $\varphi$ is satisfied in a structure with respect to $s$, only depend on the assignments $s$ makes to the variables in $t$ and the free variables of $\varphi$. This is the content of the next two propositions. To make the idea of "depends on" precise, we show that any two variable assignments that agree on all the variables in $t$ give the same value, and that $\varphi$ is satisfied relative to one iff it is satisfied relative to the other if two variable assignments agree on all free variables of $\varphi$. explanation

fol:syn:ass:prop:valindep

**Proposition syn.1.** *If the variables in a term $t$ are among $x_1, \dots, x_n$, and $s_1(x_i) = s_2(x_i)$ for $i = 1, \dots, n$, then $\mathrm{Val}^{\mathfrak{M}}_{s_1}(t) = \mathrm{Val}^{\mathfrak{M}}_{s_2}(t)$.*

*Proof.* By induction on the complexity of $t$. For the base case, $t$ can be a constant symbol or one one of the variables $x_1, \dots, x_n$. If $t = c$, then $\mathrm{Val}^{\mathfrak{M}}_{s_1}(t) = c^{\mathfrak{M}} = \mathrm{Val}^{\mathfrak{M}}_{s_2}(t)$. If $t = x_i$, $s_1(x_i) = s_2(x_i)$ by the hypothesis of the proposition, and so $\mathrm{Val}^{\mathfrak{M}}_{s_1}(t) = s_1(x_i) = s_2(x_i) = \mathrm{Val}^{\mathfrak{M}}_{s_2}(t)$.

For the inductive step, assume that $t = f(t_1, \dots, t_k)$ and that the claim holds for $t_1, \dots, t_k$. Then

$$\begin{aligned}\mathrm{Val}^{\mathfrak{M}}_{s_1}(t) &= \mathrm{Val}^{\mathfrak{M}}_{s_1}(f(t_1, \dots, t_k)) = \\ &= f^{\mathfrak{M}}(\mathrm{Val}^{\mathfrak{M}}_{s_1}(t_1), \dots, \mathrm{Val}^{\mathfrak{M}}_{s_1}(t_k))\end{aligned}$$

For $j = 1, \dots, k$, the variables of $t_j$ are among $x_1, \dots, x_n$. So by induction hypothesis, $\mathrm{Val}^{\mathfrak{M}}_{s_1}(t_j) = \mathrm{Val}^{\mathfrak{M}}_{s_2}(t_j)$. So,

$$\begin{aligned}\mathrm{Val}^{\mathfrak{M}}_{s_1}(t) &= \mathrm{Val}^{\mathfrak{M}}_{s_2}(f(t_1, \dots, t_k)) = \\ &= f^{\mathfrak{M}}(\mathrm{Val}^{\mathfrak{M}}_{s_1}(t_1), \dots, \mathrm{Val}^{\mathfrak{M}}_{s_1}(t_k)) = \\ &= f^{\mathfrak{M}}(\mathrm{Val}^{\mathfrak{M}}_{s_2}(t_1), \dots, \mathrm{Val}^{\mathfrak{M}}_{s_2}(t_k)) = \\ &= \mathrm{Val}^{\mathfrak{M}}_{s_2}(f(t_1, \dots, t_k)) = \mathrm{Val}^{\mathfrak{M}}_{s_2}(t).\end{aligned}$$

□

fol:syn:ass:prop:satindep

**Proposition syn.2.** *If the free variables in $\varphi$ are among $x_1, \dots, x_n$, and $s_1(x_i) = s_2(x_i)$ for $i = 1, \dots, n$, then $\mathfrak{M}, s_1 \vDash \varphi$ iff $\mathfrak{M}, s_2 \vDash \varphi$.*

*Proof.* We use induction on the complexity of $\varphi$. For the base case, where $\varphi$ is atomic, $\varphi$ can be: $\top$, $\bot$, $R(t_1, \dots, t_k)$ for a $k$-place predicate $R$ and terms $t_1$, $\dots$, $t_k$, or $t_1 = t_2$ for terms $t_1$ and $t_2$.

1. $\varphi \equiv \top$: both $\mathfrak{M}, s_1 \vDash \varphi$ and $\mathfrak{M}, s_2 \vDash \varphi$.

2. $\varphi \equiv \bot$: both $\mathfrak{M}, s_1 \nvDash \varphi$ and $\mathfrak{M}, s_2 \nvDash \varphi$.

3. $\varphi \equiv R(t_1, \dots, t_k)$: let $\mathfrak{M}, s_1 \vDash \varphi$. Then

$$\langle \text{Val}^{\mathfrak{M}}_{s_1}(t_1), \dots, \text{Val}^{\mathfrak{M}}_{s_1}(t_k) \rangle \in R^{\mathfrak{M}}.$$

   For $i = 1, \dots, k$, $\text{Val}^{\mathfrak{M}}_{s_1}(t_i) = \text{Val}^{\mathfrak{M}}_{s_2}(t_i)$ by Proposition syn.1. So we also have $\langle \text{Val}^{\mathfrak{M}}_{s_2}(t_i), \dots, \text{Val}^{\mathfrak{M}}_{s_2}(t_k) \rangle \in R^{\mathfrak{M}}$.

4. $\varphi \equiv t_1 = t_2$: suppose $\mathfrak{M}, s_1 \vDash \varphi$. Then $\text{Val}^{\mathfrak{M}}_{s_1}(t_1) = \text{Val}^{\mathfrak{M}}_{s_1}(t_2)$. So,

$$\begin{aligned} \text{Val}^{\mathfrak{M}}_{s_2}(t_1) &= \text{Val}^{\mathfrak{M}}_{s_1}(t_1) && \text{(by Proposition syn.1)} \\ &= \text{Val}^{\mathfrak{M}}_{s_1}(t_2) && \text{(since } \mathfrak{M}, s_1 \vDash t_1 = t_2\text{)} \\ &= \text{Val}^{\mathfrak{M}}_{s_2}(t_2) && \text{(by Proposition syn.1)}, \end{aligned}$$

   so $\mathfrak{M}, s_2 \vDash t_1 = t_2$.

Now assume $\mathfrak{M}, s_1 \vDash \psi$ iff $\mathfrak{M}, s_2 \vDash \psi$ for all formulas $\psi$ less complex than $\varphi$. The induction step proceeds by cases determined by the main operator of $\varphi$. In each case, we only demonstrate the forward direction of the biconditional; the proof of the reverse direction is symmetrical. In all cases except those for the quantifiers, we apply the induction hypothesis to sub-formulas $\psi$ of $\varphi$. The free variables of $\psi$ are among those of $\varphi$. Thus, if $s_1$ and $s_2$ agree on the free variables of $\varphi$, they also agree on those of $\psi$, and the induction hypothesis applies to $\psi$.

1. $\varphi \equiv \neg\psi$: if $\mathfrak{M}, s_1 \vDash \varphi$, then $\mathfrak{M}, s_1 \nvDash \psi$, so by the induction hypothesis, $\mathfrak{M}, s_2 \nvDash \psi$, hence $\mathfrak{M}, s_2 \vDash \varphi$.

2. $\varphi \equiv \psi \land \chi$: if $\mathfrak{M}, s_1 \vDash \varphi$, then $\mathfrak{M}, s_1 \vDash \psi$ and $\mathfrak{M}, s_1 \vDash \chi$, so by induction hypothesis, $\mathfrak{M}, s_2 \vDash \psi$ and $\mathfrak{M}, s_2 \vDash \chi$. Hence, $\mathfrak{M}, s_2 \vDash \varphi$.

3. $\varphi \equiv \psi \lor \chi$: if $\mathfrak{M}, s_1 \vDash \varphi$, then $\mathfrak{M}, s_1 \vDash \psi$ or $\mathfrak{M}, s_1 \vDash \chi$. By induction hypothesis, $\mathfrak{M}, s_2 \vDash \psi$ or $\mathfrak{M}, s_2 \vDash \chi$, so $\mathfrak{M}, s_2 \vDash \varphi$.

4. $\varphi \equiv \psi \rightarrow \chi$: if $\mathfrak{M}, s_1 \vDash \varphi$, then $\mathfrak{M}, s_1 \nvDash \psi$ or $\mathfrak{M}, s_1 \vDash \chi$. By the induction hypothesis, $\mathfrak{M}, s_2 \nvDash \psi$ or $\mathfrak{M}, s_2 \vDash \chi$, so $\mathfrak{M}, s_2 \vDash \varphi$.

5. $\varphi \equiv \psi \leftrightarrow \chi$: if $\mathfrak{M}, s_1 \vDash \varphi$, then either $\mathfrak{M}, s_1 \vDash \psi$ and $\mathfrak{M}, s_1 \vDash \chi$, or $\mathfrak{M}, s_1 \nvDash \psi$ and $\mathfrak{M}, s_1 \nvDash \chi$. By the induction hypothesis, either $\mathfrak{M}, s_2 \vDash \psi$ and $\mathfrak{M}, s_2 \vDash \chi$ or $\mathfrak{M}, s_2 \nvDash \psi$ and $\mathfrak{M}, s_2 \nvDash \chi$. In either case, $\mathfrak{M}, s_2 \vDash \varphi$.

6. $\varphi \equiv \exists x\, \psi$: if $\mathfrak{M}, s_1 \vDash \varphi$, there is an $x$-variant $s'_1$ of $s_1$ so that $\mathfrak{M}, s'_1 \vDash \psi$. Let $s'_2$ be the $x$-variant of $s_2$ that assigns the same thing to $x$ as does $s'_1$. The free variables of $\psi$ are among $x_1, \dots, x_n$, and $x$. $s'_1(x_i) = s'_2(x_i)$, since $s'_1$ and $s'_2$ are $x$-variants of $s_1$ and $s_2$, respectively, and by hypothesis $s_1(x_i) = s_2(x_i)$. $s'_1(x) = s'_2(x)$ by the way we have defined $s'_2$. Then the induction hypothesis applies to $\psi$ and $s'_1$, $s'_2$, so $\mathfrak{M}, s'_2 \vDash \psi$. Hence, there is an $x$-variant of $s_2$ that satisfies $\psi$, and so $\mathfrak{M}, s_2 \vDash \varphi$.

7. $\varphi \equiv \forall x\, \psi$: if $\mathfrak{M}, s_1 \vDash \varphi$, then for every $x$-variant $s_1'$ of $s_1$, $\mathfrak{M}, s_1' \vDash \psi$. Take an arbitrary $x$-variant $s_2'$ of $s_2$, let $s_1'$ be the $x$-variant of $s_1$ which assigns the same thing to $x$ as does $s_2'$. The free variables of $\psi$ are among $x_1, \ldots, x_n$, and $x$. $s_1'(x_i) = s_2'(x_i)$, since $s_1'$ and $s_2'$ are $x$-variants of $s_1$ and $s_2$, respectively, and by hypothesis $s_1(x_i) = s_2(x_i)$. $s_1'(x) = s_2'(x)$ by the way we have defined $s_1'$. Then the induction hypothesis applies to $\psi$ and $s_1'$, $s_2$, and we have $\mathfrak{M}, s_2' \vDash \psi$. Since $s_2'$ is an arbitrary $x$-variant of $s_2$, every $x$-variant of $s_2$ satisfies $\psi$, and so $\mathfrak{M}, s_2 \vDash \varphi$.

By induction, we get that $\mathfrak{M}, s_1 \vDash \varphi$ iff $\mathfrak{M}, s_2 \vDash \varphi$ whenever the free variables in $\varphi$ are among $x_1, \ldots, x_n$ and $s_1(x_i) = s_2(x_i)$ for $i = 1, \ldots, n$. □

**Problem syn.1.** Complete the proof of Proposition syn.2.

Sentences have no free variables, so any two variable assignments assign the same things to all the (zero) free variables of any sentence. The proposition just proved then means that whether or not a sentence is satisfied in a structure relative to a variable assignment is completely independent of the assignment. We'll record this fact. It justifies the definition of satisfaction of a sentence in a structure (without mentioning a variable assignment) that follows. explanation

*fol:syn:ass: cor:sat-sentence* **Corollary syn.3.** *If $\varphi$ is a sentence and $s$ a variable assignment, then $\mathfrak{M}, s \vDash \varphi$ iff $\mathfrak{M}, s' \vDash \varphi$ for every variable assignment $s'$.*

*Proof.* Let $s'$ be any variable assignment. Since $\varphi$ is a sentence, it has no free variables, and so every variable assignment $s'$ trivially assigns the same things to all free variables of $\varphi$ as does $s$. So the condition of Proposition syn.2 is satisfied, and we have $\mathfrak{M}, s \vDash \varphi$ iff $\mathfrak{M}, s' \vDash \varphi$. □

fol:syn:ass: defn:satisfaction **Definition syn.4.** If $\varphi$ is a sentence, we say that a structure $\mathfrak{M}$ *satisfies* $\varphi$, $\mathfrak{M} \vDash \varphi$, iff $\mathfrak{M}, s \vDash \varphi$ for all variable assignments $s$.

If $\mathfrak{M} \vDash \varphi$, we also simply say that $\varphi$ *is true in* $\mathfrak{M}$.

*fol:syn:ass: prop:sentence-sat-true* **Proposition syn.5.** *Let $\mathfrak{M}$ be a structure, $\varphi$ be a sentence, and $s$ a variable assignment. $\mathfrak{M} \vDash \varphi$ iff $\mathfrak{M}, s \vDash \varphi$.*

*Proof.* Exercise. □

**Problem syn.2.** Prove Proposition syn.5

*fol:syn:ass: prop:sat-quant* **Proposition syn.6.** *Suppose $\varphi(x)$ only contains $x$ free, and $\mathfrak{M}$ is a structure. Then:*

1. *$\mathfrak{M} \vDash \exists x\, \varphi(x)$ iff $\mathfrak{M}, s \vDash \varphi(x)$ for at least one variable assignment $s$.*
2. *$\mathfrak{M} \vDash \forall x\, \varphi(x)$ iff $\mathfrak{M}, s \vDash \varphi(x)$ for all variable assignments $s$.*

*Proof.* Exercise. □

**Problem syn.3.** Prove Proposition syn.6.

**Problem syn.4.** Suppose $\mathcal{L}$ is a language without function symbols. Given a structure $\mathfrak{M}$, $c$ a constant symbol and $a \in |\mathfrak{M}|$, define $\mathfrak{M}[a/c]$ to be the structure that is just like $\mathfrak{M}$, except that $c^{\mathfrak{M}[a/c]} = a$. Define $\mathfrak{M} \mid\models \varphi$ for sentences $\varphi$ by:

1. $\varphi \equiv \bot$: not $\mathfrak{M} \mid\models \varphi$.
2. $\varphi \equiv \top$: $\mathfrak{M} \mid\models \varphi$.
3. $\varphi \equiv R(d_1, \dots, d_n)$: $\mathfrak{M} \mid\models \varphi$ iff $\langle d_1^{\mathfrak{M}}, \dots, d_n^{\mathfrak{M}} \rangle \in R^{\mathfrak{M}}$.
4. $\varphi \equiv d_1 = d_2$: $\mathfrak{M} \mid\models \varphi$ iff $d_1^{\mathfrak{M}} = d_2^{\mathfrak{M}}$.
5. $\varphi \equiv \neg\psi$: $\mathfrak{M} \mid\models \varphi$ iff not $\mathfrak{M} \mid\models \psi$.
6. $\varphi \equiv (\psi \wedge \chi)$: $\mathfrak{M} \mid\models \varphi$ iff $\mathfrak{M} \mid\models \psi$ and $\mathfrak{M} \mid\models \chi$.
7. $\varphi \equiv (\psi \vee \chi)$: $\mathfrak{M} \mid\models \varphi$ iff $\mathfrak{M} \mid\models \psi$ or $\mathfrak{M} \mid\models \chi$ (or both).
8. $\varphi \equiv (\psi \to \chi)$: $\mathfrak{M} \mid\models \varphi$ iff not $\mathfrak{M} \mid\models \psi$ or $\mathfrak{M} \mid\models \chi$ (or both).
9. $\varphi \equiv (\psi \leftrightarrow \chi)$: $\mathfrak{M} \mid\models \varphi$ iff either both $\mathfrak{M} \mid\models \psi$ and $\mathfrak{M} \mid\models \chi$, or neither $\mathfrak{M} \mid\models \psi$ nor $\mathfrak{M} \mid\models \chi$.
10. $\varphi \equiv \forall x\, \psi$: $\mathfrak{M} \mid\models \varphi$ iff for all $a \in |\mathfrak{M}|$, $\mathfrak{M}[a/c] \mid\models \psi[c/x]$, if $c$ does not occur in $\psi$.
11. $\varphi \equiv \exists x\, \psi$: $\mathfrak{M} \mid\models \varphi$ iff there is an $a \in |\mathfrak{M}|$ such that $\mathfrak{M}[a/c] \mid\models \psi[c/x]$, if $c$ does not occur in $\psi$.

Let $x_1, \dots, x_n$ be all free variables in $\varphi$, $c_1, \dots, c_n$ constant symbols not in $\varphi$, $a_1, \dots, a_n \in |\mathfrak{M}|$, and $s(x_i) = a_i$.

Show that $\mathfrak{M}, s \models \varphi$ iff $\mathfrak{M}[a_1/c_1, \dots, a_n/c_n] \mid\models \varphi[c_1/x_1] \dots [c_n/x_n]$.

(This problem shows that it is possible to give a semantics for first-order logic that makes do without variable assignments.)

**Problem syn.5.** Suppose that $f$ is a function symbol not in $\varphi(x, y)$. Show that there is a structure $\mathfrak{M}$ such that $\mathfrak{M} \models \forall x\, \exists y\, \varphi(x, y)$ iff there is an $\mathfrak{M}'$ such that $\mathfrak{M}' \models \forall x\, \varphi(x, f(x))$.

(This problem is a special case of what's known as Skolem's Theorem; $\forall x\, \varphi(x, f(x))$ is called a *Skolem normal form* of $\forall x\, \exists y\, \varphi(x, y)$.)

## Photo Credits

## Bibliography