

syn.1 Satisfaction of a Formula in a Structure

fol:syn:sat:
sec

The basic notion that relates expressions such as terms and **formulas**, on the one hand, and **structures** on the other, are those of **value** of a term and **satisfaction** of a **formula**. Informally, the **value** of a term is an **element** of a **structure**—if the term is just a constant, its **value** is the object assigned to the constant by the **structure**, and if it is built up using **function symbols**, the **value** is computed from the **values** of constants and the functions assigned to the functions in the term. A **formula** is *satisfied* in a **structure** if the interpretation given to the predicates makes the **formula** true in the domain of the **structure**. This notion of satisfaction is specified inductively: the specification of the **structure** directly states when atomic **formulas** are satisfied, and we define when a complex **formula** is satisfied depending on the main connective or quantifier and whether or not the immediate **subformulas** are satisfied. The case of the quantifiers here is a bit tricky, as the immediate **subformula** of a quantified **formula** has a free **variable**, and **structures** don't specify the **values** of **variables**. In order to deal with this difficulty, we also introduce *variable assignments* and define satisfaction not with respect to a **structure** alone, but with respect to a **structure** plus a **variable** assignment.

explanation

Definition syn.1 (Variable Assignment). A *variable assignment* s for a **structure** \mathfrak{M} is a function which maps each **variable** to an element of $|\mathfrak{M}|$, i.e., $s: \text{Var} \rightarrow |\mathfrak{M}|$.

A **structure** assigns a **value** to each **constant symbol**, and a variable assignment to each variable. But we want to use terms built up from them to also name **elements** of the **domain**. For this we define the **value** of terms inductively. For **constant symbols** and variables the value is just as the **structure** or the variable assignment specifies it; for more complex terms it is computed recursively using the functions the **structure** assigns to the **function symbols**.

explanation

Definition syn.2 (Value of Terms). If t is a term of the language \mathcal{L} , \mathfrak{M} is a **structure** for \mathcal{L} , and s is a **variable** assignment for \mathfrak{M} , the *value* $\text{Val}_s^{\mathfrak{M}}(t)$ is defined as follows:

1. $t \equiv c$: $\text{Val}_s^{\mathfrak{M}}(t) = c^{\mathfrak{M}}$.
2. $t \equiv x$: $\text{Val}_s^{\mathfrak{M}}(t) = s(x)$.
3. $t \equiv f(t_1, \dots, t_n)$:

$$\text{Val}_s^{\mathfrak{M}}(t) = f^{\mathfrak{M}}(\text{Val}_s^{\mathfrak{M}}(t_1), \dots, \text{Val}_s^{\mathfrak{M}}(t_n)).$$

Definition syn.3 (x -Variant). If s is a **variable** assignment for a **structure** \mathfrak{M} , then any **variable** assignment s' for \mathfrak{M} which differs from s at most in what it assigns to x is called an *x -variant* of s . If s' is an x -variant of s we write $s \sim_x s'$.

explanation

Note that an x -variant of an assignment s does not *have* to assign something different to x . In fact, every assignment counts as an x -variant of itself.

Definition syn.4 (Satisfaction). Satisfaction of a **formula** φ in a **structure** \mathfrak{M} relative to a **variable** assignment s , in symbols: $\mathfrak{M}, s \models \varphi$, is defined recursively as follows. (We write $\mathfrak{M}, s \not\models \varphi$ to mean “not $\mathfrak{M}, s \models \varphi$.”)

fol:syn:sat:
defn:satisfaction

1. $\varphi \equiv \perp$: $\mathfrak{M}, s \not\models \varphi$.
2. $\varphi \equiv \top$: $\mathfrak{M}, s \models \varphi$.
3. $\varphi \equiv R(t_1, \dots, t_n)$: $\mathfrak{M}, s \models \varphi$ iff $\langle \text{Val}_s^{\mathfrak{M}}(t_1), \dots, \text{Val}_s^{\mathfrak{M}}(t_n) \rangle \in R^{\mathfrak{M}}$.
4. $\varphi \equiv t_1 = t_2$: $\mathfrak{M}, s \models \varphi$ iff $\text{Val}_s^{\mathfrak{M}}(t_1) = \text{Val}_s^{\mathfrak{M}}(t_2)$.
5. $\varphi \equiv \neg\psi$: $\mathfrak{M}, s \models \varphi$ iff $\mathfrak{M}, s \not\models \psi$.
6. $\varphi \equiv (\psi \wedge \chi)$: $\mathfrak{M}, s \models \varphi$ iff $\mathfrak{M}, s \models \psi$ and $\mathfrak{M}, s \models \chi$.
7. $\varphi \equiv (\psi \vee \chi)$: $\mathfrak{M}, s \models \varphi$ iff $\mathfrak{M}, s \models \psi$ or $\mathfrak{M}, s \models \chi$ (or both).
8. $\varphi \equiv (\psi \rightarrow \chi)$: $\mathfrak{M}, s \models \varphi$ iff $\mathfrak{M}, s \not\models \psi$ or $\mathfrak{M}, s \models \chi$ (or both).
9. $\varphi \equiv (\psi \leftrightarrow \chi)$: $\mathfrak{M}, s \models \varphi$ iff either both $\mathfrak{M}, s \models \psi$ and $\mathfrak{M}, s \models \chi$, or neither $\mathfrak{M}, s \models \psi$ nor $\mathfrak{M}, s \models \chi$.
10. $\varphi \equiv \forall x \psi$: $\mathfrak{M}, s \models \varphi$ iff for every x -variant s' of s , $\mathfrak{M}, s' \models \psi$.
11. $\varphi \equiv \exists x \psi$: $\mathfrak{M}, s \models \varphi$ iff there is an x -variant s' of s so that $\mathfrak{M}, s' \models \psi$.

explanation

The variable assignments are important in the last two clauses. We cannot define satisfaction of $\forall x \psi(x)$ by “for all $a \in |\mathfrak{M}|$, $\mathfrak{M} \models \psi(a)$.” We cannot define satisfaction of $\exists x \psi(x)$ by “for at least one $a \in |\mathfrak{M}|$, $\mathfrak{M} \models \psi(a)$.” The reason is that a is not symbol of the language, and so $\psi(a)$ is not a **formula** (that is, $\psi[a/x]$ is undefined). We also cannot assume that we have **constant symbols** or terms available that name every **element** of \mathfrak{M} , since there is nothing in the definition of **structures** that requires it. Even in the standard language the set of **constant symbols** is **denumerable**, so if $|\mathfrak{M}|$ is not **enumerable** there aren't even enough **constant symbols** to name every object.

Example syn.5. Let $=\{a, b, f, R\}$ where a and b are **constant symbols**, f is a two-place **function symbol**, and R is a two-place **predicate symbol**. Consider the **structure** \mathfrak{M} defined by:

1. $|\mathfrak{M}| = \{1, 2, 3, 4\}$
2. $a^{\mathfrak{M}} = 1$
3. $b^{\mathfrak{M}} = 2$
4. $f^{\mathfrak{M}}(x, y) = x + y$ if $x + y \leq 3$ and $= 3$ otherwise.

$$5. R^{\mathfrak{M}} = \{\langle 1, 1 \rangle, \langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 2, 4 \rangle\}$$

The function $s(x) = 1$ that assigns $1 \in |\mathfrak{M}|$ to every **variable** is a variable assignment for \mathfrak{M} .

Then

$$\text{Val}_s^{\mathfrak{M}}(f(a, b)) = f^{\mathfrak{M}}(\text{Val}_s^{\mathfrak{M}}(a), \text{Val}_s^{\mathfrak{M}}(b)).$$

Since a and b are **constant symbols**, $\text{Val}_s^{\mathfrak{M}}(a) = a^{\mathfrak{M}} = 1$ and $\text{Val}_s^{\mathfrak{M}}(b) = b^{\mathfrak{M}} = 2$. So

$$\text{Val}_s^{\mathfrak{M}}(f(a, b)) = f^{\mathfrak{M}}(1, 2) = 1 + 2 = 3.$$

To compute the value of $f(f(a, b), a)$ we have to consider

$$\text{Val}_s^{\mathfrak{M}}(f(f(a, b), a)) = f^{\mathfrak{M}}(\text{Val}_s^{\mathfrak{M}}(f(a, b)), \text{Val}_s^{\mathfrak{M}}(a)) = f^{\mathfrak{M}}(3, 1) = 3,$$

since $3 + 1 > 3$. Since $s(x) = 1$ and $\text{Val}_s^{\mathfrak{M}}(x) = s(x)$, we also have

$$\text{Val}_s^{\mathfrak{M}}(f(f(a, b), x)) = f^{\mathfrak{M}}(\text{Val}_s^{\mathfrak{M}}(f(a, b)), \text{Val}_s^{\mathfrak{M}}(x)) = f^{\mathfrak{M}}(3, 1) = 3,$$

An atomic **formula** $R(t_1, t_2)$ is satisfied if the tuple of values of its arguments, i.e., $\langle \text{Val}_s^{\mathfrak{M}}(t_1), \text{Val}_s^{\mathfrak{M}}(t_2) \rangle$, is an **element** of $R^{\mathfrak{M}}$. So, e.g., we have $\mathfrak{M}, s \models R(b, f(a, b))$ since $\langle \text{Val}_s^{\mathfrak{M}}(b), \text{Val}_s^{\mathfrak{M}}(f(a, b)) \rangle = \langle 2, 3 \rangle \in R^{\mathfrak{M}}$, but $\mathfrak{M}, s \not\models R(x, f(a, b))$ since $\langle 1, 3 \rangle \notin R^{\mathfrak{M}}[s]$.

To determine if a non-atomic formula φ is satisfied, you apply the clauses in the inductive definition that applies to the main connective. For instance, the main connective in $R(a, a) \rightarrow (R(b, x) \vee R(x, b))$ is the \rightarrow , and

$$\begin{aligned} \mathfrak{M}, s \models R(a, a) \rightarrow (R(b, x) \vee R(x, b)) \text{ iff} \\ \mathfrak{M}, s \not\models R(a, a) \text{ or } \mathfrak{M}, s \models R(b, x) \vee R(x, b) \end{aligned}$$

Since $\mathfrak{M}, s \models R(a, a)$ (because $\langle 1, 1 \rangle \in R^{\mathfrak{M}}$) we can't yet determine the answer and must first figure out if $\mathfrak{M}, s \models R(b, x) \vee R(x, b)$:

$$\begin{aligned} \mathfrak{M}, s \models R(b, x) \vee R(x, b) \text{ iff} \\ \mathfrak{M}, s \models R(b, x) \text{ or } \mathfrak{M}, s \models R(x, b) \end{aligned}$$

And this is the case, since $\mathfrak{M}, s \models R(x, b)$ (because $\langle 1, 2 \rangle \in R^{\mathfrak{M}}$).

Recall that an x -variant of s is a variable assignment that differs from s at most in what it assigns to x . For every **element** of $|\mathfrak{M}|$, there is an x -variant of s : $s_1(x) = 1$, $s_2(x) = 2$, $s_3(x) = 3$, $s_4(x) = 4$, and with $s_i(y) = s(y) = 1$ for all variables y other than x . These are all the x -variants of s for the structure \mathfrak{M} ,

since $|\mathfrak{M}| = \{1, 2, 3, 4\}$. Note, in particular, that $s_1 = s$ is also an x -variant of s , i.e., s is always an x -variant of itself.

To determine if an existentially quantified **formula** $\exists x \varphi(x)$ is satisfied, we have to determine if $\mathfrak{M}, s' \models \varphi(x)$ for at least one x -variant s' of s . So,

$$\mathfrak{M}, s \models \exists x (R(b, x) \vee R(x, b)),$$

since $\mathfrak{M}, s_1 \models R(b, x) \vee R(x, b)$ (s_3 would also fit the bill). But,

$$\mathfrak{M}, s \not\models \exists x (R(b, x) \wedge R(x, b))$$

since for none of the s_i , $\mathfrak{M}, s_i \models R(b, x) \wedge R(x, b)$.

To determine if a universally quantified **formula** $\forall x \varphi(x)$ is satisfied, we have to determine if $\mathfrak{M}, s' \models \varphi(x)$ for all x -variants s' of s . So,

$$\mathfrak{M}, s \models \forall x (R(x, a) \rightarrow R(a, x)),$$

since $\mathfrak{M}, s_i \models R(x, a) \rightarrow R(a, x)$ for all s_i ($\mathfrak{M}, s_1 \models R(a, x)$ and $\mathfrak{M}, s_j \not\models R(x, a)$ for $j = 2, 3$, and 4). But,

$$\mathfrak{M}, s \not\models \forall x (R(a, x) \rightarrow R(x, a))$$

since $\mathfrak{M}, s_2 \not\models R(a, x) \rightarrow R(x, a)$ (because $\mathfrak{M}, s_2 \models R(a, x)$ and $\mathfrak{M}, s_2 \not\models R(x, a)$).

For a more complicated case, consider

$$\forall x (R(a, x) \rightarrow \exists y R(x, y)).$$

Since $\mathfrak{M}, s_3 \not\models R(a, x)$ and $\mathfrak{M}, s_4 \not\models R(a, x)$, the interesting cases where we have to worry about the consequent of the conditional are only s_1 and s_2 . Does $\mathfrak{M}, s_1 \models \exists y R(x, y)$ hold? It does if there is at least one y -variant s'_1 of s_1 so that $\mathfrak{M}, s'_1 \models R(x, y)$. In fact, s_1 is such a y -variant ($s_1(x) = 1$, $s_1(y) = 1$, and $\langle 1, 1 \rangle \in R^{\mathfrak{M}}$), so the answer is yes. To determine if $\mathfrak{M}, s_2 \models \exists y R(x, y)$ we have to look at the y -variants of s_2 . Here, s_2 itself does not satisfy $R(x, y)$ ($s_2(x) = 2$, $s_2(y) = 1$, and $\langle 2, 1 \rangle \notin R^{\mathfrak{M}}$). However, consider $s'_2 \sim_y s_2$ with $s'_2(y) = 3$. $\mathfrak{M}, s'_2 \models R(x, y)$ since $\langle 2, 3 \rangle \in R^{\mathfrak{M}}$, and so $\mathfrak{M}, s_2 \models \exists y R(x, y)$. In sum, for every x -variant s_i of s , either $\mathfrak{M}, s_i \not\models R(a, x)$ ($i = 3, 4$) or $\mathfrak{M}, s_i \models \exists y R(x, y)$ ($i = 1, 2$), and so

$$\mathfrak{M}, s \models \forall x (R(a, x) \rightarrow \exists y R(x, y)).$$

On the other hand,

$$\mathfrak{M}, s \not\models \exists x (R(a, x) \wedge \forall y R(x, y)).$$

The only x -variants s_i of s with $\mathfrak{M}, s_i \models R(a, x)$ are s_1 and s_2 . But for each, there is in turn a y -variant $s'_i \sim_y s_i$ with $s'_i(y) = 4$ so that $\mathfrak{M}, s'_i \not\models R(x, y)$ and so $\mathfrak{M}, s_i \not\models \forall y R(x, y)$ for $i = 1, 2$. In sum, none of the x -variants $s_i \sim_x s$ are such that $\mathfrak{M}, s_i \models R(a, x) \wedge \forall y R(x, y)$.

Problem syn.1. Let $\mathcal{L} = \{c, f, A\}$ with one **constant symbol**, one one-place **function symbol** and one two-place **predicate symbol**, and let the **structure** \mathfrak{M} be given by

1. $|\mathfrak{M}| = \{1, 2, 3\}$
2. $c^{\mathfrak{M}} = 3$
3. $f^{\mathfrak{M}}(1) = 2, f^{\mathfrak{M}}(2) = 3, f^{\mathfrak{M}}(3) = 2$
4. $A^{\mathfrak{M}} = \{\langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 3, 3 \rangle\}$

(a) Let $s(v) = 1$ for all variables v . Find out whether

$$\mathfrak{M}, s \models \exists x (A(f(z), c) \rightarrow \forall y (A(y, x) \vee A(f(y), x)))$$

Explain why or why not.

(b) Give a different structure and variable assignment in which the formula is not satisfied.

Photo Credits

Bibliography