SUMO Example with Interpretation

September 18, 2019

We will consider ten propositions in a fragment of the language of SUMO. The ten propositions are about the obligation to brush one's teeth.

1. $(\mathsf{modalAttribute}\, p\, \mathsf{Obligation} \Leftrightarrow (\neg \mathsf{modalAttribute}\, (\neg p)\, \mathsf{Permission}))$.
2. $(\mathsf{confersObligation}\, (\forall d.((\mathsf{instance}\, d\, \mathsf{Day}) \Rightarrow (\exists b.((\mathsf{instance}\, b\, \mathsf{BrushingTeeth}) \wedge ((\mathsf{during}\, b\, (\mathsf{WhenFn}\, d)) \wedge (\mathsf{agent}\, b\, \mathsf{Me}))))))\, \mathsf{MyMom}\, \mathsf{Me})$.
3. $(\mathsf{hasAuthorityOver}\, \mathsf{MyMom}\, \mathsf{Me})$.
4. $(\forall a_1.(\forall a_2.((\mathsf{confersObligation}\, p\, a_1\, a_2) \Rightarrow (\mathsf{holdsObligation}\, p\, a_2))))$.
5. $(\forall u.(\forall a.((\mathsf{holdsDuring}\, t\, ((\mathsf{hasAuthorityOver}\, u\, a) \wedge (\mathsf{confersObligation}\, f\, u\, a))) \Rightarrow (\mathsf{holdsDuring}\, t\, (\mathsf{desires}\, a\, f)))))$.
6. $(\forall u.(\forall a.(((\mathsf{hasAuthorityOver}\, u\, a) \wedge (\mathsf{confersObligation}\, (\mathsf{holdsDuring}\, t\, f)\, u\, a)) \Rightarrow (\mathsf{desires}\, a\, (\mathsf{holdsDuring}\, t\, f)))))$.
7. $(\forall u.(\forall a.(((\mathsf{hasAuthorityOver}\, u\, a) \wedge (\mathsf{confersObligation}\, f\, u\, a)) \Rightarrow (\mathsf{desires}\, a\, f))))$.
8. $(\forall a.((\mathsf{holdsObligation}\, f\, a) \Rightarrow \mathsf{modalAttribute}\, f\, \mathsf{Obligation}))$.
9. $(\forall a.((\mathsf{holdsRight}\, f\, a) \Rightarrow \mathsf{modalAttribute}\, f\, \mathsf{Permission}))$.
10. $(\neg(\mathsf{holdsRight}\, (\exists d.((\mathsf{instance}\, d\, \mathsf{Day}) \wedge (\neg(\exists b.((\mathsf{instance}\, b\, \mathsf{BrushingTeeth}) \wedge ((\mathsf{during}\, b\, (\mathsf{WhenFn}\, d)) \wedge (\mathsf{agent}\, b\, \mathsf{Me}))))))\, \mathsf{Me}))$.

A sufficient fragment to cover these ten propositions is given by the following grammar for propositions and individuals where we allow variables (p) ranging over propositions, variables (t) ranging over time and variables ($a, u, x, \ldots$) ranging over individuals.

$$
\begin{array}{rcl}
\text{Time} & (\tau) & ::= t \text{ (only variables for now)} \\
\text{Ind} & (\mu, \nu) & ::= \mathsf{Me} | \mathsf{MyMom} | \mathsf{Day} | \mathsf{BrushingTeeth} | \mathsf{WhenFn}\ \mu \\
\text{Prop} & (\varphi, \psi) & \top | \bot | \neg\varphi | \varphi \wedge \psi | \varphi \Rightarrow \psi | \varphi \Leftrightarrow \psi | \forall x.\varphi | \exists x.\varphi \\
 & & \mathsf{instance}\ \mu\ \nu | \mathsf{during}\ \mu\ \nu | \mathsf{hasAuthorityOver}\ \mu\ \nu | \mathsf{agent}\ \mu\ \nu \\
 & & \mathsf{modalAttribute}\ \varphi\ \mathsf{Obligation} | \mathsf{modalAttribute}\ \varphi\ \mathsf{Permission} \\
 & & \mathsf{confersObligation}\ \varphi\ \mu\ \nu | \mathsf{desires}\ \mu\ \varphi | \mathsf{holdsRight}\ \varphi\ \mu | \mathsf{holdsObligation}\ \varphi\ \mu \\
 & & \mathsf{holdsDuring}\ \tau\ \varphi
\end{array}
$$

The language is not ordinary first-order logic since there are modalities. The most obvious modalities are the deontological operators $\mathsf{modalAttribute}\ \varphi\ \mathsf{Obligation}$ and $\mathsf{modalAttribute}\ \varphi\ \mathsf{Permission}$. Propositions of the form $\mathsf{holdsDuring}\ \tau\ \varphi$ can be seen as using a temporal modality. We will also use two modalities to allow implicit references to subjects and objects.

Kripke semantics gives an easy way to interpret modalities. Instead of interpreting propositions as being true or false, we interpret propositions as a set of worlds. Informally speaking, propositions are "true" if the set of worlds includes all worlds.

Since we will have four different kinds of modalities (time, subject, object and deontological) it would seem we need four different kinds of worlds. However, we will combine these together by simply thinking of a world as a point in a four-dimensional space. Each dimension of the space corresponds to the relevant modal information for that world.

With this in mind, we extend the language above to contain four new kinds of terms:

$$\begin{aligned} \mathsf{World} &::= w \text{ (only variables for now)} \\ \mathsf{Subj} &::= s \text{ (only variables for now)} \\ \mathsf{Obj} &::= o \text{ (only variables for now)} \\ \mathsf{Deon} &::= e \text{ (only variables for now)} \end{aligned}$$

The four dimensions Time, Subj, Obj and Deon will each have their own accessibility relation. We can form the compound accessibility relation on worlds by saying a world ξ is related to ξ' iff when we project onto all four dimensions the pairs of components satisfy the corresponding accessibility relations.

Now, we describe how to map the language above into a simply typed λ-calculus with multiple base types. The mapping is intended to β-normalize to give multisorted first-order formulas (at least most of the time).

As usual we will have a base type o of propositions. (This should not be confused with the notion of proposition above, which will map to $\omega \to o$, e.g., sets of worlds.) The other base types will be ι (individuals), ω (worlds), τ (time), σ (subjects), ξ (objects) and δ (deontology). All other types are of the form $\alpha \to \beta$ where α and β are types. As is usual in Church style λ-calculus, we assume every variable has a unique type. If a term s has type $\alpha \to \beta$ and a term t has type α, then $(s\ t)$ has type β. If a variable x has type α and a term t has type β, then $(\lambda x.t)$ has type $\alpha \to \beta$.

Let us start by describing the infrastructure for worlds. We have the following typed constants.

- rtime : $\tau \to \tau \to o$ (the accessibility relation for time)
- wtime : $\omega \to \tau$ (the projection function giving the time for each world)
- attime : $\tau \to \omega \to \omega$ (a function taking a time and a world and returning a world which is like the given world but at a different time) The intention is that ω is a tuple like (t, s, o, e) and attime t' ω is the world (t', s, o, e).
- reify_time : $\tau \to \iota$ (take a time and turn it into an individual)
- rsubj : $\tau \to \tau \to o$ (the accessibility relation for subjects)
- wsubj : $\omega \to \tau$ (the projection function giving the subject for each world)
- atsubj : $\tau \to \omega \to \omega$ (a function taking a subject and a world and returning a world which is like the given world but at a different subject)
- reify_subj : $\tau \to \iota$ (take a subject and turn it into an individual)
- robj : $\tau \to \tau \to o$ (the accessibility relation for objects)
- wobj : $\omega \to \tau$ (the projection function giving the object for each world)
- atobj : $\tau \to \omega \to \omega$ (a function taking an object and a world and returning a world which is like the given world but at a different object)
- reify_obj : $\tau \to \iota$ (take a obj and turn it into an individual)
- rdeon : $\tau \to \tau \to o$ (the accessibility relation for deontic dimension)
- wdeon : $\omega \to \tau$ (the projection function giving the deontic component for each world)

We can give a way to translate the original ten SUMO formulas into terms of type $\omega \to o$ by giving terms of appropriate type for each of the constructs in the fragment of the language. In many cases we will simply assume we have a corresponding typed constant and use it:

- $\mathsf{Day} : \iota$
- $\mathsf{MyMom} : \iota$
- $\mathsf{Me} : \iota$
- $\mathsf{BrushingTeeth} : \iota$
- $\mathsf{WhenFn} : \iota \to \iota$
- $\mathsf{instance} : \iota \to \iota \to \omega \to o$
- $\mathsf{during} : \iota \to \iota \to \omega \to o$
- $\mathsf{hasAuthorityOver} : \iota \to \iota \to \omega \to o$
- $\mathsf{agent} : \iota \to \iota \to \omega \to o$
- $\mathsf{desires} : \iota \to (\omega \to o) \to \omega \to o$

We will also assume a new typed constant:

- $\mathsf{confers} : (\omega \to o) \to \iota \to \iota \to \omega \to o$

For the remaining constructs of the SUMO fragment we will use special interpretations, indicated below. For the special cases of the obligation and permission modalities we will use a box and diamond operation specific to the deontological dimension δ.

For SUMO individuals above, we let $[\mu] : \iota$ be the obvious corresponding term defined recursively. For simplicity, we will also assume the SUMO individual variables correspond to variables of type ι with the same name, SUMO time variables correspond to variables of type τ with the same name, and SUMO propositional variables correspond to variables of type $\omega \to o$ with the same name.

We will define a translation on some SUMO propositions φ yielding a term $[\varphi] : \omega \to o$ below. Once we have this translation we can translate the propositions above by universally quantifying over all free variables. For propositional variables in the SUMO formula, this will correspond to quantifying over a variable of type $\omega \to o$. We then universally quantify over worlds. In the end, the term we obtain is of the form

$$\forall ... \forall w : \omega.[\varphi]w$$

We will often be able to β-normalize this formula and obtain a multisorted first-order formula.

Here is each case of the translation:

- $[\top]$ is $\lambda w.\top$, where $\top : o$ in the image is some notion of truth in Church's type theory. The same will be true for logical connectives and quantifiers below.
- $[\bot]$ is $\lambda w.\bot$.
- $[\neg\varphi]$ is $\lambda w.\neg([\varphi]w)$.
- $[\varphi \wedge \psi]$ is $\lambda w.(([\varphi]w) \wedge ([\psi]w))$.
- $[\varphi \Rightarrow \psi]$ is $\lambda w.(([\varphi]w) \Rightarrow ([\psi]w))$.
- $[\varphi \Leftrightarrow \psi]$ is $\lambda w.(([\varphi]w) \Leftrightarrow ([\psi]w))$.
- $[\forall x.\varphi]$ is $\lambda w.\forall x.[\varphi]w$
- $[\exists x.\varphi]$ is $\lambda w.\exists x.[\varphi]w$
- $[\mathsf{instance}\ \mu\ \nu]$ is $\mathsf{instance}\ [\mu]\ [\nu]$
- $[\mathsf{during}\ \mu\ \nu]$ is $\mathsf{during}\ [\mu]\ [\nu]$

- $[\mathsf{hasAuthorityOver}\ \mu\ \nu]$ is $\mathsf{hasAuthorityOver}\ [\mu]\ [\nu]$
- $[\mathsf{agent}\ \mu\ \nu]$ is $\mathsf{agent}\ [\mu]\ [\nu]$
- $[\mathsf{modalAttribute}\ \varphi\ \mathsf{Obligation}]$ is $\lambda w.\forall w' : \omega.\mathsf{rdeon}\ (\mathsf{wdeon}\ w)\ (\mathsf{wdeon}\ w') \Rightarrow [\varphi]w'$
- $[\mathsf{modalAttribute}\ \varphi\ \mathsf{Permission}]$ is $\lambda w.\exists w' : \omega.\mathsf{rdeon}\ (\mathsf{wdeon}\ w)\ (\mathsf{wdeon}\ w') \wedge [\varphi]w'$
- $[\mathsf{confersObligation}\ \varphi\ \mu\ \nu]$ is

  $\lambda w.\exists s : \sigma.\mathsf{reify\_subj} s = [\nu] \wedge \mathsf{confers}\ [\varphi]\ [\mu]\ [\nu]\ w \wedge \forall w'.\mathsf{rdeon}\ (\mathsf{wdeon}\ (\mathsf{atsubj}\ s\ w))\ (\mathsf{wdeon}\ w') \Rightarrow [\varphi]w'$

- $[\mathsf{desires}\ \mu\ \varphi]$ is $\mathsf{desires}\ [\mu]\ [\varphi]$
- $[\mathsf{holdsRight}\ \varphi\ \mu]$ is

$$\lambda w.\exists s : \sigma.\mathsf{reify\_subj} s = [\mu] \wedge \exists w'.\mathsf{rdeon}\ (\mathsf{wdeon}\ (\mathsf{atsubj}\ s\ w))\ (\mathsf{wdeon}\ w') \wedge [\varphi]w'$$

- $[\mathsf{holdsObligation}\ \varphi\ \mu]$ is

$$\lambda w.\exists s : \sigma.\mathsf{reify\_subj} s = [\mu] \wedge \forall w'.\mathsf{rdeon}\ (\mathsf{wdeon}\ (\mathsf{atsubj}\ s\ w))\ (\mathsf{wdeon}\ w') \Rightarrow [\varphi]w'$$

- $[\mathsf{holdsDuring}\ \tau\ \varphi]$ is $\lambda w.[\varphi](\mathsf{attime}\ [\tau]\ w)$

After translation and normalization five of the ten propositions become provable (under some assumptions). The specific five are 1, 4, 8, 9 and 10.