CompMeta

By Martin Lundfall, Denis Erfurt

June 21, 2016

Contents

theory *QMLS5U*
imports *Main*

begin

1 Preliminaris

We present a semantic embedding of quantified modal logic (QML) in classical higher-order logic (HOL). Quantifiers are provided for Boolean, first-, second- and higher-order variables (for all types). The theoretical background of the work presented here has been discussed in [**?**]. This file is intended for reuse by further AFP articles on QML.

We begin by introducing type i for the set of possible worlds and type μ for the set of individuals. Formulae in quantified modal logic (QML) are functions from the set of possible worlds to Booleans. For convenience, their type is written as σ.

typedecl i — type for possible worlds
typedecl μ — type for individuals
type-synonym $\sigma = (i \Rightarrow bool)$

2 Embedding of Base Logic *K*

In Kripke semantics, a modal formula is interpreted over an arbitrary accessibility relation, a binary relation between possible worlds.

consts $r :: i \Rightarrow i \Rightarrow bool$ (**infixr** r *70*) — accessibility relation r

The set of classical connectives and quantifiers is *lifted* to the modal level by passing an additional parameter w, representing the current world, to the connectives' subformulae or binders' scope. This parameter is only used actively in the definition of both modalities $\{\Box, \diamond\}$, where it is applied to the accessibility relation r.

Modal connectives are typeset in bold font.[1] Abbreviations are used in place of definitions to avoid explicit mention of the embeddings' definitions when invoking automated tools via *Sledgehammer*.

abbreviation *mtrue* $:: \sigma$ ($\boldsymbol{\top}$)
 where $\boldsymbol{\top} \equiv \lambda w.\ True$
abbreviation *mfalse* $:: \sigma$ ($\boldsymbol{\bot}$)
 where $\boldsymbol{\bot} \equiv \lambda w.\ False$
abbreviation *mnot* $:: \sigma \Rightarrow \sigma$ ($\boldsymbol{\neg}$-[*52*]*53*)
 where $\boldsymbol{\neg}\varphi \equiv \lambda w.\ \neg\varphi(w)$
abbreviation *mnegpred* $:: (\mu \Rightarrow \sigma) \Rightarrow (\mu \Rightarrow \sigma)$ ($^{\neg}$-[*52*]*53*)
 where $^{\neg}\Phi \equiv \lambda x.\lambda w.\ \neg\Phi(x)(w)$
abbreviation *mand* $:: \sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr**$\boldsymbol{\wedge}$*51*)
 where $\varphi \boldsymbol{\wedge} \psi \equiv \lambda w.\ \varphi(w) \wedge \psi(w)$
abbreviation *mor* $:: \sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr**$\boldsymbol{\vee}$*50*)
 where $\varphi \boldsymbol{\vee} \psi \equiv \lambda w.\ \varphi(w) \vee \psi(w)$
abbreviation *mimp* $:: \sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr**$\boldsymbol{\rightarrow}$*49*)
 where $\varphi \boldsymbol{\rightarrow} \psi \equiv \lambda w.\ \varphi(w) \longrightarrow \psi(w)$
abbreviation *mequ* $:: \sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr**$\boldsymbol{\leftrightarrow}$*48*)
 where $\varphi \boldsymbol{\leftrightarrow} \psi \equiv \lambda w.\ \varphi(w) \longleftrightarrow \psi(w)$
abbreviation *mforall* $:: ('a \Rightarrow \sigma) \Rightarrow \sigma$ ($\boldsymbol{\forall}$)
 where $\boldsymbol{\forall}\,\Phi \equiv \lambda w.\forall\, x.\ \Phi(x)(w)$
abbreviation *mforallB* $:: ('a \Rightarrow \sigma) \Rightarrow \sigma$ (**binder**$\boldsymbol{\forall}$ [*8*]*9*)
 where $\boldsymbol{\forall}\, x.\ \varphi(x) \equiv \boldsymbol{\forall}\,\varphi$
abbreviation *mexists* $:: ('a \Rightarrow \sigma) \Rightarrow \sigma$ ($\boldsymbol{\exists}$)
 where $\boldsymbol{\exists}\,\Phi \equiv \lambda w.\exists\, x.\ \Phi(x)(w)$
abbreviation *mexistsB* $:: ('a \Rightarrow \sigma) \Rightarrow \sigma$ (**binder**$\boldsymbol{\exists}$ [*8*]*9*)

[1] In Isabelle/jEdit, bold characters can be entered by typing `\bol` before entering the actual character.

where $\exists\, x.\ \varphi(x) \equiv \exists\, \varphi$
abbreviation *meq* :: $\mu \Rightarrow \mu \Rightarrow \sigma$ (**infixr=**52) — Equality
where $x{=}y \equiv \lambda w.\ x = y$
abbreviation *meqL* :: $\mu \Rightarrow \mu \Rightarrow \sigma$ (**infixr=**$^L 52$) — Leibniz Equality
where $x{=}^L y \equiv \forall\, \varphi.\ \varphi(x) \rightarrow \varphi(y)$
abbreviation *mbox* :: $\sigma \Rightarrow \sigma$ ($\Box$-[52]53)
where $\Box\varphi \equiv \lambda w. \forall\, v.\ \varphi(v)$
abbreviation *mdia* :: $\sigma \Rightarrow \sigma$ ($\Diamond$-[52]53)
where $\Diamond\varphi \equiv \lambda w. \exists\, v.\ \varphi(v)$

Finally, a formula is valid if and only if it is satisfied in all worlds.

abbreviation *valid* :: $\sigma \Rightarrow bool$ ($\lfloor$-$\rfloor$[8]109)
where $\lfloor p \rfloor \equiv \forall\, w.\ p\ w$

3 Axiomatizations of Further Systems

Different modal logics can be axiomatized through adding a choice of the following definitions as axioms:

abbreviation M
where $M \equiv \forall\, \varphi.\ \Box\varphi \rightarrow \varphi$
abbreviation B
where $B \equiv \forall\, \varphi.\ \varphi \rightarrow \Box\Diamond\varphi$
abbreviation D
where $D \equiv \forall\, \varphi.\ \Box\varphi \rightarrow \Diamond\varphi$
abbreviation IV
where $IV \equiv \forall\, \varphi.\ \Box\varphi \rightarrow \Box\Box\varphi$
abbreviation V
where $V \equiv \forall\, \varphi.\ \Diamond\varphi \rightarrow \Box\Diamond\varphi$

Because the embedding is of a semantic nature, it is more efficient to instead make use of the well-known *Sahlqvist correspondence*, which links these axioms to constraints on a model's accessibility relation: axioms M, B, D, IV, V impose reflexivity, symmetry, seriality, transitivity and euclideanness respectively.

abbreviation *reflexive*
where *reflexive* $\equiv (\forall\, x.\ x\ r\ x)$
abbreviation *symmetric*
where *symmetric* $\equiv (\forall\, x\ y.\ x\ r\ y \longrightarrow y\ r\ x)$
abbreviation *serial* :: *bool*
where *serial* $\equiv (\forall\, x.\ \exists\, y.\ x\ r\ y)$
abbreviation *transitive* :: *bool*
where *transitive* $\equiv (\forall\, x\ y\ z.\ ((x\ r\ y) \wedge (y\ r\ z) \longrightarrow (x\ r\ z)))$
abbreviation *euclidean* :: *bool*
where *euclidean* $\equiv (\forall\, x\ y\ z.\ ((x\ r\ y) \wedge (x\ r\ z) \longrightarrow (y\ r\ z)))$

Using these definitions, we can derive axioms for the most common modal logics. Thereby we are free to use either the semantic constraints or the related Sahlqvist axioms. Here we provide both versions. We recommend to use the semantic constraints.

abbreviation *D-sem* :: *bool*
where *D-sem* ≡ *serial*
abbreviation *D-ax* :: *bool*
where *D-ax* ≡ $\lfloor D \rfloor$
abbreviation *B-sem* :: *bool*
where *B-sem* ≡ *symmetric*
abbreviation *B-ax* :: *bool*
where *B-ax* ≡ $\lfloor B \rfloor$
abbreviation *T-sem* :: *bool*
where *T-sem* ≡ *reflexive*
abbreviation *T-ax* :: *bool*
where *T-ax* ≡ $\lfloor M \rfloor$
abbreviation *S4-sem* :: *bool*
where *S4-sem* ≡ *reflexive* ∧ *transitive*
abbreviation *S4-ax* :: *bool*
where *S4-ax* ≡ $\lfloor M \rfloor \wedge \lfloor IV \rfloor$
abbreviation *S5-sem* :: *bool*
where *S5-sem* ≡ *reflexive* ∧ *euclidean*
abbreviation *S5-ax* :: *bool*
where *S5-ax* ≡ $\lfloor M \rfloor \wedge \lfloor V \rfloor$

end

theory *Temp*
imports *Main*

begin

4 Preliminaris

We present a semantic embedding of temporal logic (Temp) in classical higher-order logic (HOL). Quantifiers are provided for Boolean, first-, second- and higher-order variables (for all types).

We begin by introducing type i for the set of possible (future) worlds and type μ for the set of individuals. Formulae in quantified temporal logic (QML) are functions from the set of possible worlds to Booleans. For convenience, their type is written as σ.

typedecl *ii* — type for possible (future) worlds
typedecl $\mu\mu$ — type for individuals
type-synonym $\sigma\sigma$ = (*ii*⇒*bool*)

5 Embedding of Base Logic $\boldsymbol{K}$

In Kripke semantics, a modal formula is interpreted over an arbitrary accessibility relation, a binary relation between possible worlds.

consts $r :: ii \Rightarrow ii \Rightarrow bool$ (**infixr** r *70*) — accessibility relation r

The set of classical connectives and quantifiers is *lifted* to the modal level by passing an additional parameter w, representing the current world, to the connectives' subformulae or binders' scope. This parameter is only used actively in the definition of both modalities $\{\Box, \diamond\}$, where it is applied to the accessibility relation r.

Temporal connectives are typeset in cursive font.[2] Abbreviations are used in place of definitions to avoid explicit mention of the embeddings' definitions when invoking automated tools via *Sledgehammer*.

abbreviation *ttrue* $:: \sigma\sigma$ $(*\top)$
  where $*\top \equiv \lambda w.\ True$
abbreviation *mfalse* $:: \sigma\sigma$ $(*\bot)$
  where $*\bot \equiv \lambda w.\ False$
abbreviation *mnot* $:: \sigma\sigma \Rightarrow \sigma\sigma$ $(*\neg\text{-}[52]53)$
  where $*\neg\varphi \equiv \lambda w.\ \neg\varphi(w)$
abbreviation *mnegpred* $:: (\mu\mu \Rightarrow \sigma\sigma) \Rightarrow (\mu\mu \Rightarrow \sigma\sigma)$ $(*^{\neg}\text{-}[52]53)$
  where $*^{\neg}\Phi \equiv \lambda x.\lambda w.\ \neg\Phi(x)(w)$
abbreviation *mand* $:: \sigma\sigma \Rightarrow \sigma\sigma \Rightarrow \sigma\sigma$ (**infixr**$*\wedge$*51*)
  where $\varphi * \wedge \psi \equiv \lambda w.\ \varphi(w) \wedge \psi(w)$
abbreviation *mor* $:: \sigma\sigma \Rightarrow \sigma\sigma \Rightarrow \sigma\sigma$ (**infixr**$*\vee$*50*)
  where $\varphi * \vee \psi \equiv \lambda w.\ \varphi(w) \vee \psi(w)$
abbreviation *mimp* $:: \sigma\sigma \Rightarrow \sigma\sigma \Rightarrow \sigma\sigma$ (**infixr**$*\rightarrow$*49*)
  where $\varphi * \rightarrow \psi \equiv \lambda w.\ \varphi(w) \longrightarrow \psi(w)$
abbreviation *mequ* $:: \sigma\sigma \Rightarrow \sigma\sigma \Rightarrow \sigma\sigma$ (**infixr**$*\leftrightarrow$*48*)
  where $\varphi * \leftrightarrow \psi \equiv \lambda w.\ \varphi(w) \longleftrightarrow \psi(w)$
abbreviation *mforall* $:: ('a \Rightarrow \sigma\sigma) \Rightarrow \sigma\sigma$ $(*\forall)$
  where $*\forall\,\Phi \equiv \lambda w.\forall\,x.\ \Phi(x)(w)$
abbreviation *mforallB* $:: ('a \Rightarrow \sigma\sigma) \Rightarrow \sigma\sigma$ (**binder**$*\forall\,[8]9$)
  where $*\forall\,x.\ \varphi(x) \equiv *\forall\,\varphi$
abbreviation *mexists* $:: ('a \Rightarrow \sigma\sigma) \Rightarrow \sigma\sigma$ $(*\exists)$
  where $*\exists\,\Phi \equiv \lambda w.\exists\,x.\ \Phi(x)(w)$
abbreviation *mexistsB* $:: ('a \Rightarrow \sigma\sigma) \Rightarrow \sigma\sigma$ (**binder**$*\exists\,[8]9$)
  where $*\exists\,x.\ \varphi(x) \equiv *\exists\,\varphi$
abbreviation *meq* $:: \mu\mu \Rightarrow \mu\mu \Rightarrow \sigma\sigma$ (**infixr**$*{=}$*52*) — Equality
  where $x*{=}y \equiv \lambda w.\ x = y$
abbreviation *meqL* $:: \mu\mu \Rightarrow \mu\mu \Rightarrow \sigma\sigma$ (**infixr**$*{=}^{L}$*52*) — Leibniz Equality
  where $x*{=}^{L}y \equiv *\forall\,\varphi.\ \varphi(x) * \rightarrow \varphi(y)$
abbreviation P $:: \sigma\sigma \Rightarrow \sigma\sigma$ $(P\text{-}[52]53)$
  where $P\ \varphi \equiv \lambda w.\ (\exists\,v.\ v\ r\ w \wedge \varphi(v))$
abbreviation F $:: \sigma\sigma \Rightarrow \sigma\sigma$ $(F\text{-}[52]53)$

[2] In Isabelle/jEdit, bold characters can be entered by typing `\emph` before entering the actual character.

where $F\ \varphi \equiv \lambda w.\ (\exists\, v.\ w\ r\ v \wedge \varphi(v))$
abbreviation H $\ ::\ \sigma\sigma \Rightarrow \sigma\sigma\ (H\text{-}[52]53)$
where $H\ \varphi \equiv \lambda w. \forall\, v.\ v\ r\ w \longrightarrow \varphi(v)$
abbreviation G $\ ::\ \sigma\sigma \Rightarrow \sigma\sigma\ (G\text{-}[52]53)$
where $G\ \varphi \equiv \lambda w. \forall\, v.\ w\ r\ v \longrightarrow \varphi(v)$

Finally, a formula is valid if and only if it is satisfied in all worlds.

abbreviation *valid* $::\ \sigma\sigma \Rightarrow bool\ (*\lfloor\text{-}\rfloor*[8]109)$
where $*\lfloor p \rfloor* \equiv \forall\, w.\ p\ w$

abbreviation *reflexive*
where *reflexive* $\equiv (\forall\, x.\ x\ r\ x)$
abbreviation *symmetric*
where *symmetric* $\equiv (\forall\, x\ y.\ x\ r\ y \longrightarrow y\ r\ x)$
abbreviation *serial* :: *bool*
where *serial* $\equiv (\forall\, x.\ \exists\, y.\ x\ r\ y)$
abbreviation *serial2* :: *bool*
where *serial2* $\equiv (\forall\, x.\ \exists\, y.\ y\ r\ x)$
abbreviation *transitive* :: *bool*
where *transitive* $\equiv (\forall\, x\ y\ z.\ ((x\ r\ y) \wedge (y\ r\ z) \longrightarrow (x\ r\ z)))$
abbreviation *euclidean* :: *bool*
where *euclidean* $\equiv (\forall\, x\ y\ z.\ ((x\ r\ y) \wedge (x\ r\ z) \longrightarrow (y\ r\ z)))$
abbreviation *total* :: *bool*
where *total* $\equiv \forall\, x\ y.\ x\ r\ y \vee y\ r\ x$

abbreviation *Kt-sem* :: *bool*

where *Kt-sem* $\equiv$ *transitive* $\wedge$ *serial* $\wedge$ *serial2* $\wedge$ *total*

end
theory *LundfallErfurtEx09*
imports *QMLS5U Main Temp*
begin

6 Exercise 2

theorem M:
shows $\lfloor \Box p \rightarrow p \rfloor$
by *simp*

theorem V:
shows $\lfloor \forall\, \varphi.\ \Diamond\varphi \rightarrow \Box\Diamond\varphi \rfloor$
by *simp*

consts $Pp\ ::\ (\mu \Rightarrow \sigma) \Rightarrow \sigma$

definition $God :: \mu \Rightarrow \sigma$ **where** $God = (\lambda x.\ \forall(\lambda\Phi.\ Pp\ \Phi \rightarrow \Phi\ x))$

definition $ess :: (\mu \Rightarrow \sigma) \Rightarrow \mu \Rightarrow \sigma$ (**infixr** *ess 85*) **where**
$\Phi\ ess\ x = \Phi\ x \wedge (\forall \Psi.\ \Psi\ x \rightarrow \Box(\forall y.\ \Phi\ y \rightarrow \Psi\ y))$

definition $NE :: \mu \Rightarrow \sigma$ **where** $NE = (\lambda x.\ \forall \Phi.\ \Phi\ ess\ x \rightarrow \Box(\exists\ \Phi))$

axiomatization where

$A1a$: $\lfloor \forall \Phi.\ Pp(\neg\Phi) \rightarrow \neg(Pp\ \Phi) \rfloor$ **and**
$A1b$: $\lfloor \forall \Phi.\ \neg(Pp\ \Phi) \rightarrow Pp\ (\neg\Phi) \rfloor$ **and**

$A2$: $\lfloor \forall \Phi.\ \forall \Psi.\ (Pp\ \Phi \wedge \Box\ (\forall x.\ \Phi\ x \rightarrow \Psi\ x)) \rightarrow Pp\ \Psi \rfloor$
axiomatization where $A3$: $\lfloor Pp\ God \rfloor$
axiomatization where $A4$: $\lfloor \forall \Phi.\ Pp\ \Phi \rightarrow \Box(Pp\ \Phi) \rfloor$
axiomatization where $A5$: $\lfloor Pp\ NE \rfloor$

theorem *god*:
shows $\lfloor \Box(\exists\ God) \rfloor$
by (*metis A1a A1b A2 A3 A4 A5 God-def NE-def ess-def*)

6.1 d

With this formalization, there is no difference between a proposition being globally valid and it being necessarily true.

7 Exercise 3

7.1 (a)

theorem *KforG*:
assumes $*\lfloor\ G\ (\Psi *\!\!\rightarrow \Phi)\ \rfloor*$
shows $*\lfloor\ G\ \Psi *\!\!\rightarrow G\ \Phi\ \rfloor*$
by (*simp add*: *assms*)

theorem *KforH*:
assumes $*\lfloor\ H(\Psi *\!\!\rightarrow \Phi)\ \rfloor*$
shows $*\lfloor\ H\ \Psi *\!\!\rightarrow H\ \Phi\ \rfloor*$
by (*simp add*: *assms*)

theorem *SymI*:
shows $*\lfloor\ \Psi *\!\!\rightarrow\ G\ (P\ \Psi)\ \rfloor*$

by *auto*

theorem *SymII*:
shows $*\lfloor\ \Psi *\!\rightarrow\ \ H\ (F\ \Psi)\ \rfloor*$
by *auto*

theorem *TRAN*:
assumes *Kt-sem*
shows $*\lfloor G\ \Psi *\!\rightarrow\ G(G\ \Psi)\rfloor*$
using *assms* **by** *blast*

theorem *NOEND*:
assumes *Kt-sem*
shows $*\lfloor\ G\ \Psi *\!\rightarrow F\ \Psi\rfloor*$
using *assms* **by** *blast*

theorem *NOBEG*:
assumes *Kt-sem*
shows $*\lfloor\ H\ \Psi *\!\rightarrow P\ \Psi\rfloor*$
using *assms* **by** *blast*

theorem *LIN*:
assumes *Kt-sem*
shows $*\lfloor\ (P\ (F\ \Psi) *\!\vee F\ (P\ \Psi)) *\!\rightarrow (P\ \Psi) *\!\vee \Psi *\!\vee (F\ \Psi)\ \rfloor*$
using *assms* **by** *blast*

7.2 (c)

consts *dead* :: $\mu\mu \Rightarrow \sigma\sigma$
theorem *deadness*:
assumes *Kt-sem* $\wedge *\lfloor\ *\forall\ \ entity.\ \ (dead(entity) *\!\rightarrow\ G\ dead(entity)) *\!\wedge\ F\ dead(entity) *\!\wedge\ (F *\!\neg\ dead(entity) *\!\vee\ P *\!\neg\ dead(entity) *\!\vee *\!\neg\ dead(entity))\ \rfloor*$
shows $*\lfloor\ *\forall\ \ entity.\ P\ (H *\!\neg\ dead(entity))\ \rfloor*$
by (*metis assms*)

end