DTU Library

Epistemic Logic: Completeness of Modal Logics

From, Asta Halkjær

Published in:
Archive of Formal Proofs

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

Epistemic Logic: Completeness of Modal Logics

Asta Halkjær From

November 30, 2021

Abstract

This work is a formalization of epistemic logic with countably many agents. It includes proofs of soundness and completeness for the axiom system K. The completeness proof is based on the textbook "Reasoning About Knowledge" by Fagin, Halpern, Moses and Vardi (MIT Press 1995) [2]. The extensions of system K (T, KB, K4, S4, S5) and their completeness proofs are based on the textbook "Modal Logic" by Blackburn, de Rijke and Venema (Cambridge University Press 2001) [1].

Contents

theory *Epistemic-Logic* **imports** *HOL−Library.Countable* **begin**

1 Syntax

```
type-synonym id = string

datatype 'i fm
  = FF (⊥)
  | Pro id
  | Dis ‹'i fm› ‹'i fm› (infixr ∨ 30)
  | Con ‹'i fm› ‹'i fm› (infixr ∧ 35)
  | Imp ‹'i fm› ‹'i fm› (infixr ⟶ 25)
  | K 'i ‹'i fm›

abbreviation TT (⊤) where
  ‹TT ≡ ⊥ ⟶ ⊥›

abbreviation Neg (¬ - [40] 40) where
  ‹Neg p ≡ p ⟶ ⊥›

abbreviation ‹L i p ≡ ¬ K i (¬ p)›
```

2 Semantics

```
datatype ('i, 'w) kripke =
  Kripke (𝒲: ‹'w set›) (π: ‹'w ⇒ id ⇒ bool›) (𝒦: ‹'i ⇒ 'w ⇒ 'w set›)

primrec semantics :: ‹('i, 'w) kripke ⇒ 'w ⇒ 'i fm ⇒ bool›
  (-, - ⊨ - [50, 50] 50) where
  ‹(M, w ⊨ ⊥) = False›
| ‹(M, w ⊨ Pro x) = π M w x›
| ‹(M, w ⊨ (p ∨ q)) = ((M, w ⊨ p) ∨ (M, w ⊨ q))›
| ‹(M, w ⊨ (p ∧ q)) = ((M, w ⊨ p) ∧ (M, w ⊨ q))›
| ‹(M, w ⊨ (p ⟶ q)) = ((M, w ⊨ p) ⟶ (M, w ⊨ q))›
| ‹(M, w ⊨ K i p) = (∀ v ∈ 𝒲 M ∩ 𝒦 M i w. M, v ⊨ p)›
```

3 S5 Axioms

```
definition reflexive :: ‹('i, 'w) kripke ⇒ bool› where
  ‹reflexive M ≡ ∀ i. ∀ w ∈ 𝒲 M. w ∈ 𝒦 M i w›

definition symmetric :: ‹('i, 'w) kripke ⇒ bool› where
  ‹symmetric M ≡ ∀ i. ∀ v ∈ 𝒲 M. ∀ w ∈ 𝒲 M. v ∈ 𝒦 M i w ⟷ w ∈ 𝒦 M i v›

definition transitive :: ‹('i, 'w) kripke ⇒ bool› where
  ‹transitive M ≡ ∀ i. ∀ u ∈ 𝒲 M. ∀ v ∈ 𝒲 M. ∀ w ∈ 𝒲 M.
```

$w \in \mathcal{K}\ M\ i\ v \wedge u \in \mathcal{K}\ M\ i\ w \longrightarrow u \in \mathcal{K}\ M\ i\ v$›

abbreviation *equivalence* :: ‹$('i,\ 'w)$ *kripke* $\Rightarrow$ *bool*› **where**
‹*equivalence* $M \equiv$ *reflexive* $M \wedge$ *symmetric* $M \wedge$ *transitive* M›

lemma *Imp-intro* [*intro*]: ‹$(M,\ w \models p \Longrightarrow M,\ w \models q) \Longrightarrow M,\ w \models (p \longrightarrow q)$›
⟨*proof*⟩

theorem *distribution*: ‹$M,\ w \models (K\ i\ p \wedge K\ i\ (p \longrightarrow q) \longrightarrow K\ i\ q)$›
⟨*proof*⟩

theorem *generalization*:
assumes *valid*: ‹$\forall (M :: ('i,\ 'w)\ kripke)\ w.\ M,\ w \models p$›
shows ‹$(M :: ('i,\ 'w)\ kripke),\ w \models K\ i\ p$›
⟨*proof*⟩

theorem *truth*:
assumes ‹*reflexive* M› ‹$w \in \mathcal{W}\ M$›
shows ‹$M,\ w \models (K\ i\ p \longrightarrow p)$›
⟨*proof*⟩

theorem *pos-introspection*:
assumes ‹*transitive* M› ‹$w \in \mathcal{W}\ M$›
shows ‹$M,\ w \models (K\ i\ p \longrightarrow K\ i\ (K\ i\ p))$›
⟨*proof*⟩

theorem *neg-introspection*:
assumes ‹*symmetric* M› ‹*transitive* M› ‹$w \in \mathcal{W}\ M$›
shows ‹$M,\ w \models (\neg\ K\ i\ p \longrightarrow K\ i\ (\neg\ K\ i\ p))$›
⟨*proof*⟩

4 Normal Modal Logic

primrec *eval* :: ‹$(id \Rightarrow bool) \Rightarrow ('i\ fm \Rightarrow bool) \Rightarrow 'i\ fm \Rightarrow bool$› **where**
‹*eval* - - $\bot$ = *False*›
| ‹*eval* g - $(Pro\ x) = g\ x$›
| ‹*eval* $g\ h\ (p \vee q) = (eval\ g\ h\ p \vee eval\ g\ h\ q)$›
| ‹*eval* $g\ h\ (p \wedge q) = (eval\ g\ h\ p \wedge eval\ g\ h\ q)$›
| ‹*eval* $g\ h\ (p \longrightarrow q) = (eval\ g\ h\ p \longrightarrow eval\ g\ h\ q)$›
| ‹*eval* - $h\ (K\ i\ p) = h\ (K\ i\ p)$›

abbreviation ‹*tautology* $p \equiv \forall\, g\ h.\ eval\ g\ h\ p$›

inductive *AK* :: ‹$('i\ fm \Rightarrow bool) \Rightarrow 'i\ fm \Rightarrow bool$› (- $\vdash$ - [*50*, *50*] *50*)
for A :: ‹$'i\ fm \Rightarrow bool$› **where**
A1: ‹*tautology* $p \Longrightarrow A \vdash p$›
| *A2*: ‹$A \vdash (K\ i\ p \wedge K\ i\ (p \longrightarrow q) \longrightarrow K\ i\ q)$›
| *Ax*: ‹$A\ p \Longrightarrow A \vdash p$›
| *R1*: ‹$A \vdash p \Longrightarrow A \vdash (p \longrightarrow q) \Longrightarrow A \vdash q$›

| *R2*: ‹$A \vdash p \Longrightarrow A \vdash K\ i\ p$›

5 Soundness

lemma *eval-semantics*: ‹$eval\ (pi\ w)\ (\lambda q.\ Kripke\ W\ pi\ r,\ w \models q)\ p = (Kripke\ W\ pi\ r,\ w \models p)$›
⟨*proof*⟩

lemma *tautology*:
assumes ‹$tautology\ p$›
shows ‹$M,\ w \models p$›
⟨*proof*⟩

theorem *soundness*:
fixes M :: ‹$('i,\ 'w)\ kripke$›
assumes ‹$\bigwedge(M :: ('i,\ 'w)\ kripke)\ w\ p.\ A\ p \Longrightarrow P\ M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
shows ‹$A \vdash p \Longrightarrow P\ M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
⟨*proof*⟩

6 Derived rules

lemma *K-A2′*: ‹$A \vdash (K\ i\ (p \longrightarrow q) \longrightarrow K\ i\ p \longrightarrow K\ i\ q)$›
⟨*proof*⟩

lemma *K-map*:
assumes ‹$A \vdash (p \longrightarrow q)$›
shows ‹$A \vdash (K\ i\ p \longrightarrow K\ i\ q)$›
⟨*proof*⟩

lemma *K-LK*: ‹$A \vdash (L\ i\ (\neg\ p) \longrightarrow \neg\ K\ i\ p)$›
⟨*proof*⟩

primrec *imply* :: ‹$'i\ fm\ list \Rightarrow\ 'i\ fm \Rightarrow\ 'i\ fm$› **where**
‹$imply\ []\ q = q$›
| ‹$imply\ (p\ \#\ ps)\ q = (p \longrightarrow imply\ ps\ q)$›

lemma *K-imply-head*: ‹$A \vdash imply\ (p\ \#\ ps)\ p$›
⟨*proof*⟩

lemma *K-imply-Cons*:
assumes ‹$A \vdash imply\ ps\ q$›
shows ‹$A \vdash imply\ (p\ \#\ ps)\ q$›
⟨*proof*⟩

lemma *K-right-mp*:
assumes ‹$A \vdash imply\ ps\ p$› ‹$A \vdash imply\ ps\ (p \longrightarrow q)$›
shows ‹$A \vdash imply\ ps\ q$›

⟨*proof*⟩

lemma *tautology-imply-superset*:
  assumes ‹$set\ ps \subseteq set\ qs$›
  shows ‹$tautology\ (imply\ ps\ r \longrightarrow imply\ qs\ r)$›
⟨*proof*⟩

lemma *K-imply-weaken*:
  assumes ‹$A \vdash imply\ ps\ q$› ‹$set\ ps \subseteq set\ ps'$›
  shows ‹$A \vdash imply\ ps'\ q$›
⟨*proof*⟩

lemma *imply-append*: ‹$imply\ (ps\ @\ ps')\ q = imply\ ps\ (imply\ ps'\ q)$›
  ⟨*proof*⟩

lemma *K-ImpI*:
  assumes ‹$A \vdash imply\ (p\ \#\ G)\ q$›
  shows ‹$A \vdash imply\ G\ (p \longrightarrow q)$›
⟨*proof*⟩

lemma *K-Boole*:
  assumes ‹$A \vdash imply\ ((\neg\ p)\ \#\ G)\ \bot$›
  shows ‹$A \vdash imply\ G\ p$›
⟨*proof*⟩

lemma *K-DisE*:
  assumes ‹$A \vdash imply\ (p\ \#\ G)\ r$› ‹$A \vdash imply\ (q\ \#\ G)\ r$› ‹$A \vdash imply\ G\ (p \lor q)$›
  shows ‹$A \vdash imply\ G\ r$›
⟨*proof*⟩

lemma *K-mp*: ‹$A \vdash imply\ (p\ \#\ (p \longrightarrow q)\ \#\ G)\ q$›
  ⟨*proof*⟩

lemma *K-swap*:
  assumes ‹$A \vdash imply\ (p\ \#\ q\ \#\ G)\ r$›
  shows ‹$A \vdash imply\ (q\ \#\ p\ \#\ G)\ r$›
  ⟨*proof*⟩

lemma *K-DisL*:
  assumes ‹$A \vdash imply\ (p\ \#\ ps)\ q$› ‹$A \vdash imply\ (p'\ \#\ ps)\ q$›
  shows ‹$A \vdash imply\ ((p \lor p')\ \#\ ps)\ q$›
⟨*proof*⟩

lemma *K-distrib-K-imp*:
  assumes ‹$A \vdash K\ i\ (imply\ G\ q)$›
  shows ‹$A \vdash imply\ (map\ (K\ i)\ G)\ (K\ i\ q)$›
⟨*proof*⟩

7 Completeness

7.1 Consistent sets

definition *consistent* :: ‹($'i$ *fm* ⇒ *bool*) ⇒ $'i$ *fm set* ⇒ *bool*› **where**
‹*consistent* A S ≡ $\nexists S'$. *set* $S' \subseteq S \wedge A \vdash$ *imply* S' ⊥›

lemma *inconsistent-subset*:
assumes ‹*consistent* A V› ‹¬ *consistent* A ($\{p\} \cup V$)›
obtains V' **where** ‹*set* $V' \subseteq V$› ‹$A \vdash$ *imply* (p # V') ⊥›
⟨*proof*⟩

lemma *consistent-deriv*:
assumes ‹*consistent* A V› ‹$A \vdash p$›
shows ‹*consistent* A ($\{p\} \cup V$)›
⟨*proof*⟩

lemma *consistent-consequent*:
assumes ‹*consistent* A V› ‹$p \in V$› ‹$A \vdash (p \longrightarrow q)$›
shows ‹*consistent* A ($\{q\} \cup V$)›
⟨*proof*⟩

lemma *consistent-consequent'*:
assumes ‹*consistent* A V› ‹$p \in V$› ‹*tautology* $(p \longrightarrow q)$›
shows ‹*consistent* A ($\{q\} \cup V$)›
⟨*proof*⟩

lemma *consistent-disjuncts*:
assumes ‹*consistent* A V› ‹$(p \vee q) \in V$›
shows ‹*consistent* A ($\{p\} \cup V$) ∨ *consistent* A ($\{q\} \cup V$)›
⟨*proof*⟩

lemma *exists-finite-inconsistent*:
assumes ‹¬ *consistent* A ($\{\neg\, p\} \cup V$)›
obtains W **where** ‹$\{\neg\, p\} \cup W \subseteq \{\neg\, p\} \cup V$› ‹$(\neg\, p) \notin W$› ‹*finite* W› ‹¬ *consistent* A ($\{\neg\, p\} \cup W$)›
⟨*proof*⟩

lemma *inconsistent-imply*:
assumes ‹¬ *consistent* A ($\{\neg\, p\} \cup$ *set* G)›
shows ‹$A \vdash$ *imply* G p›
⟨*proof*⟩

7.2 Maximal consistent sets

definition *maximal* :: ‹($'i$ *fm* ⇒ *bool*) ⇒ $'i$ *fm set* ⇒ *bool*› **where**
‹*maximal* A S ≡ ∀ p. $p \notin S \longrightarrow$ ¬ *consistent* A ($\{p\} \cup S$)›

theorem *deriv-in-maximal*:
assumes ‹*consistent* A V› ‹*maximal* A V› ‹$A \vdash p$›

shows ‹$p \in V$›
⟨*proof*⟩

theorem *exactly-one-in-maximal*:
assumes ‹*consistent* A V› ‹*maximal* A V›
shows ‹$p \in V \longleftrightarrow (\neg\ p) \notin V$›
⟨*proof*⟩

theorem *consequent-in-maximal*:
assumes ‹*consistent* A V› ‹*maximal* A V› ‹$p \in V$› ‹$(p \longrightarrow q) \in V$›
shows ‹$q \in V$›
⟨*proof*⟩

theorem *ax-in-maximal*:
assumes ‹*consistent* A V› ‹*maximal* A V› ‹$A\ p$›
shows ‹$p \in V$›
⟨*proof*⟩

theorem *mcs-properties*:
assumes ‹*consistent* A V› **and** ‹*maximal* A V›
shows ‹$A \vdash p \Longrightarrow p \in V$›
and ‹$p \in V \longleftrightarrow (\neg\ p) \notin V$›
and ‹$p \in V \Longrightarrow (p \longrightarrow q) \in V \Longrightarrow q \in V$›
⟨*proof*⟩

7.3 Lindenbaum extension

instantiation *fm* :: (*countable*) *countable* **begin**
instance ⟨*proof*⟩
end

primrec *extend* :: ‹$('i\ fm \Rightarrow bool) \Rightarrow 'i\ fm\ set \Rightarrow (nat \Rightarrow 'i\ fm) \Rightarrow nat \Rightarrow 'i\ fm\ set$› **where**
‹*extend* A S f $0 = S$› |
‹*extend* A S f $(Suc\ n) =$
$($*if consistent* A $(\{f\ n\} \cup$ *extend* A S f $n)$
then $\{f\ n\} \cup$ *extend* A S f n
else extend A S f $n)$›

definition *Extend* :: ‹$('i\ fm \Rightarrow bool) \Rightarrow 'i\ fm\ set \Rightarrow (nat \Rightarrow 'i\ fm) \Rightarrow 'i\ fm\ set$›
where
‹*Extend* A S $f \equiv \bigcup n.$ *extend* A S f n›

lemma *Extend-subset*: ‹$S \subseteq$ *Extend* A S f›
⟨*proof*⟩

lemma *extend-bound*: ‹$(\bigcup n \le m.$ *extend* A S f $n) =$ *extend* A S f m›
⟨*proof*⟩

lemma *consistent-extend*: ‹*consistent A S* $\Longrightarrow$ *consistent A* (*extend A S f n*)›
⟨*proof*⟩

lemma *UN-finite-bound*:
assumes ‹*finite A*› ‹$A \subseteq (\bigcup n.\ f\ n)$›
shows ‹$\exists\, m :: nat.\ A \subseteq (\bigcup n \leq m.\ f\ n)$›
⟨*proof*⟩

lemma *consistent-Extend*:
assumes ‹*consistent A S*›
shows ‹*consistent A* (*Extend A S f*)›
⟨*proof*⟩

lemma *maximal-Extend*:
assumes ‹*surj f*›
shows ‹*maximal A* (*Extend A S f*)›
⟨*proof*⟩

lemma *maximal-extension*:
fixes V :: ‹('*i* :: *countable*) *fm set*›
assumes ‹*consistent A V*›
obtains W **where** ‹$V \subseteq W$› ‹*consistent A W*› ‹*maximal A W*›
⟨*proof*⟩

7.4 Canonical model

abbreviation *pi* :: ‹'*i fm set* $\Rightarrow$ *id* $\Rightarrow$ *bool*› **where**
‹*pi V x* $\equiv$ *Pro x* $\in$ *V*›

abbreviation *known* :: ‹'*i fm set* $\Rightarrow$ '*i* $\Rightarrow$ '*i fm set*› **where**
‹*known V i* $\equiv$ $\{p.\ K\ i\ p \in V\}$›

abbreviation *reach* :: ‹('*i fm* $\Rightarrow$ *bool*) $\Rightarrow$ '*i* $\Rightarrow$ '*i fm set* $\Rightarrow$ '*i fm set set*› **where**
‹*reach A i V* $\equiv$ $\{W.\ known\ V\ i \subseteq W\}$›

abbreviation *mcss* :: ‹('*i fm* $\Rightarrow$ *bool*) $\Rightarrow$ '*i fm set set*› **where**
‹*mcss A* $\equiv$ $\{W.\ consistent\ A\ W \wedge maximal\ A\ W\}$›

lemma *truth-lemma*:
fixes A **and** p :: ‹('*i* :: *countable*) *fm*›
defines ‹$M \equiv$ *Kripke* (*mcss A*) *pi* (*reach A*)›
assumes ‹*consistent A V*› **and** ‹*maximal A V*›
shows ‹$(p \in V \longleftrightarrow M,\ V \models p) \wedge ((\neg\ p) \in V \longleftrightarrow M,\ V \models \neg\ p)$›
⟨*proof*⟩

lemma *canonical-model*:
assumes ‹*consistent A S*› **and** ‹$p \in S$›
defines ‹$V \equiv$ *Extend A S from-nat*› **and** ‹$M \equiv$ *Kripke* (*mcss A*) *pi* (*reach A*)›
shows ‹$M,\ V \models p$› **and** ‹*consistent A V*› **and** ‹*maximal A V*›

$\langle proof \rangle$

7.5 Completeness

lemma *imply-completeness*:
  assumes *valid*: $\langle \forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
    $(\forall q \in G.\ M, w \models q) \longrightarrow M, w \models p\rangle$
  shows $\langle \exists qs.\ set\ qs \subseteq G \wedge (A \vdash imply\ qs\ p)\rangle$
$\langle proof \rangle$

theorem *completeness*:
  assumes $\langle \forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.\ M, w \models p\rangle$
  shows $\langle A \vdash p\rangle$
  $\langle proof \rangle$

8 System K

abbreviation *SystemK* :: $\langle 'i\ fm \Rightarrow bool\rangle$ ($\vdash_K$ - [50] 50) **where**
  $\langle \vdash_K p \equiv (\lambda\text{-}.\ False) \vdash p\rangle$

lemma *soundness*$_K$: $\langle \vdash_K p \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M, w \models p\rangle$
  $\langle proof \rangle$

abbreviation $\langle valid_K\ p \equiv \forall (M :: (nat, nat\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.\ M, w$
$\models p\rangle$

theorem *main*$_K$: $\langle valid_K\ p \longleftrightarrow \vdash_K p\rangle$
$\langle proof \rangle$

corollary
  assumes $\langle valid_K\ p\rangle$ **and** $\langle w \in \mathcal{W}\ M\rangle$
  shows $\langle M, w \models p\rangle$
$\langle proof \rangle$

9 System T

Also known as System M

inductive *AxT* :: $\langle 'i\ fm \Rightarrow bool\rangle$ **where**
  $\langle AxT\ (K\ i\ p \longrightarrow p)\rangle$

abbreviation *SystemT* :: $\langle 'i\ fm \Rightarrow bool\rangle$ ($\vdash_T$ - [50] 50) **where**
  $\langle \vdash_T p \equiv AxT \vdash p\rangle$

lemma *soundness-AxT*: $\langle AxT\ p \Longrightarrow reflexive\ M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M, w \models p\rangle$
  $\langle proof \rangle$

lemma *soundness*$_T$: $\langle \vdash_T p \Longrightarrow reflexive\ M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M, w \models p\rangle$
  $\langle proof \rangle$

lemma *AxT-reflexive*:
  assumes ‹$\forall p.\ AxT\ p \longrightarrow A\ p$› **and** ‹*consistent* $A\ V$› **and** ‹*maximal* $A\ V$›
  shows ‹$V \in$ *reach* $A\ i\ V$›
⟨*proof*⟩

lemma mcs_T*-reflexive*:
  assumes ‹$\forall p.\ AxT\ p \longrightarrow A\ p$›
  shows ‹*reflexive* (*Kripke* (*mcss* A) *pi* (*reach* A))›
  ⟨*proof*⟩

lemma *imply-completeness-T*:
  assumes *valid*: ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
    reflexive $M \longrightarrow (\forall q \in G.\ M, w \models q) \longrightarrow M, w \models p$›
  shows ‹$\exists qs.$ *set* $qs \subseteq G \land (AxT \vdash$ *imply* $qs\ p)$›
⟨*proof*⟩

lemma $completeness_T$:
  assumes ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$ *reflexive* M $\longrightarrow M, w \models p$›
  shows ‹$\vdash_T p$›
  ⟨*proof*⟩

abbreviation ‹$valid_T\ p \equiv \forall (M :: (nat, nat\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$ *reflexive* $M \longrightarrow M, w \models p$›

theorem $main_T$: ‹$valid_T\ p \longleftrightarrow \vdash_T p$›
  ⟨*proof*⟩

corollary
  assumes ‹*reflexive* M› ‹$w \in \mathcal{W}\ M$›
  shows ‹$valid_T\ p \longrightarrow M, w \models p$›
  ⟨*proof*⟩

10 System KB

inductive *AxB* :: ‹$'i\ fm \Rightarrow bool$› **where**
  ‹$AxB\ (p \longrightarrow K\ i\ (L\ i\ p))$›

abbreviation *SystemKB* :: ‹$'i\ fm \Rightarrow bool$› ($\vdash_{KB}$ - [50] 50) **where**
  ‹$\vdash_{KB} p \equiv AxB \vdash p$›

lemma *soundness-AxB*: ‹$AxB\ p \Longrightarrow$ *symmetric* $M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M, w \models p$›
  ⟨*proof*⟩

lemma $soundness_{KB}$: ‹$\vdash_{KB} p \Longrightarrow$ *symmetric* $M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M, w \models p$›
  ⟨*proof*⟩

lemma *AxB-symmetric'*:

assumes ‹$\forall p.$ *AxB* $p \longrightarrow A\ p$› ‹*consistent* $A\ V$› ‹*maximal* $A\ V$› ‹*consistent* $A\ W$› ‹*maximal* $A\ W$›
and ‹$W \in$ *reach* $A\ i\ V$›
shows ‹$V \in$ *reach* $A\ i\ W$›
⟨*proof*⟩

lemma *AxB-symmetric*:
assumes ‹$\forall p.$ *AxB* $p \longrightarrow A\ p$› ‹*consistent* $A\ V$› ‹*maximal* $A\ V$› ‹*consistent* $A\ W$› ‹*maximal* $A\ W$›
shows ‹$W \in$ *reach* $A\ i\ V \longleftrightarrow V \in$ *reach* $A\ i\ W$›
⟨*proof*⟩

lemma mcs_{KB}*-symmetric*:
assumes ‹$\forall p.$ *AxB* $p \longrightarrow A\ p$›
shows ‹*symmetric* (*Kripke* (*mcss* A) *pi* (*reach* A))›
⟨*proof*⟩

lemma *imply-completeness-KB*:
assumes *valid*: ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
symmetric $M \longrightarrow (\forall q \in G.\ M, w \models q) \longrightarrow M, w \models p$›
shows ‹$\exists qs.$ *set* $qs \subseteq G \wedge$ (*AxB* $\vdash$ *imply* $qs\ p$)›
⟨*proof*⟩

lemma $completeness_{KB}$:
assumes ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$ *symmetric* $M \longrightarrow M, w \models p$›
shows ‹$\vdash_{KB}\ p$›
⟨*proof*⟩

abbreviation ‹$valid_{KB}\ p \equiv \forall (M :: (nat, nat\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$ *symmetric* $M \longrightarrow M, w \models p$›

theorem $main_{KB}$: ‹$valid_{KB}\ p \longleftrightarrow \vdash_{KB}\ p$›
⟨*proof*⟩

corollary
assumes ‹*symmetric* M› ‹$w \in \mathcal{W}\ M$›
shows ‹$valid_{KB}\ p \longrightarrow M, w \models p$›
⟨*proof*⟩

11 System K4

inductive *Ax4* :: ‹$'i\ fm \Rightarrow bool$› **where**
‹*Ax4* ($K\ i\ p \longrightarrow K\ i\ (K\ i\ p)$)›

abbreviation *SystemK4* :: ‹$'i\ fm \Rightarrow bool$› ($\vdash_{K4}$ - [50] 50) **where**
‹$\vdash_{K4}\ p \equiv$ *Ax4* $\vdash p$›

lemma *soundness-Ax4*: ‹*Ax4* $p \Longrightarrow$ *transitive* $M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M, w \models p$›

$\langle proof \rangle$

lemma *soundness*$_{K4}$: ‹$\vdash_{K4}\ p \Longrightarrow transitive\ M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
$\langle proof \rangle$

lemma *Ax4-transitive*:
assumes ‹$\forall p.\ Ax4\ p \longrightarrow A\ p$› ‹$consistent\ A\ V$› ‹$maximal\ A\ V$›
and ‹$W \in reach\ A\ i\ V$› ‹$U \in reach\ A\ i\ W$›
shows ‹$U \in reach\ A\ i\ V$›
$\langle proof \rangle$

lemma *mcs*$_{K4}$*-transitive*:
assumes ‹$\forall p.\ Ax4\ p \longrightarrow A\ p$›
shows ‹$transitive\ (Kripke\ (mcss\ A)\ pi\ (reach\ A))$›
$\langle proof \rangle$

lemma *imply-completeness-K4*:
assumes *valid*: ‹$\forall (M :: ('i :: countable,\ 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
$transitive\ M \longrightarrow (\forall q \in G.\ M,\ w \models q) \longrightarrow M,\ w \models p$›
shows ‹$\exists qs.\ set\ qs \subseteq G \wedge (Ax4 \vdash imply\ qs\ p)$›
$\langle proof \rangle$

lemma *completeness*$_{K4}$:
assumes ‹$\forall (M :: ('i :: countable,\ 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.\ transitive\ M \longrightarrow M,\ w \models p$›
shows ‹$\vdash_{K4}\ p$›
$\langle proof \rangle$

abbreviation ‹$valid_{K4}\ p \equiv \forall (M :: (nat,\ nat\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.\ transitive\ M \longrightarrow M,\ w \models p$›

theorem *main*$_{K4}$: ‹$valid_{K4}\ p \longleftrightarrow \vdash_{K4}\ p$›
$\langle proof \rangle$

corollary
assumes ‹$transitive\ M$› ‹$w \in \mathcal{W}\ M$›
shows ‹$valid_{K4}\ p \longrightarrow M,\ w \models p$›
$\langle proof \rangle$

12 System S4

abbreviation *Or* :: ‹$('a \Rightarrow bool) \Rightarrow ('a \Rightarrow bool) \Rightarrow 'a \Rightarrow bool$› (**infixl** ‹$\oplus$› *65*) **where**
‹$A \oplus A' \equiv \lambda x.\ A\ x \vee A'\ x$›

abbreviation *SystemS4* :: ‹$'i\ fm \Rightarrow bool$› ($\vdash_{S4}$ - [*50*] *50*) **where**
‹$\vdash_{S4}\ p \equiv AxT \oplus Ax4 \vdash p$›

lemma *soundness-AxT4*: ‹$(AxT \oplus Ax4)\ p \Longrightarrow reflexive\ M \wedge transitive\ M \Longrightarrow w$

$\in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
⟨*proof*⟩

lemma $soundness_{S4}$: ‹$\vdash_{S4}\ p \Longrightarrow$ *reflexive* $M \wedge$ *transitive* $M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
⟨*proof*⟩

lemma *imply-completeness-S4*:
assumes *valid*: ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
reflexive $M \longrightarrow$ *transitive* $M \longrightarrow (\forall q \in G.\ M,\ w \models q) \longrightarrow M,\ w \models p$›
shows ‹$\exists qs.\ set\ qs \subseteq G \wedge (AxT \oplus Ax4 \vdash imply\ qs\ p)$›
⟨*proof*⟩

lemma $completeness_{S4}$:
assumes ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
reflexive $M \longrightarrow$ *transitive* $M \longrightarrow M,\ w \models p$›
shows ‹$\vdash_{S4}\ p$›
⟨*proof*⟩

abbreviation ‹$valid_{S4}\ p \equiv \forall (M :: (nat, nat\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
reflexive $M \longrightarrow$ *transitive* $M \longrightarrow M,\ w \models p$›

theorem $main_{S4}$: ‹$valid_{S4}\ p \longleftrightarrow \vdash_{S4}\ p$›
⟨*proof*⟩

corollary
assumes ‹*reflexive* M› ‹*transitive* M› ‹$w \in \mathcal{W}\ M$›
shows ‹$valid_{S4}\ p \longrightarrow M,\ w \models p$›
⟨*proof*⟩

13 System S5

abbreviation *SystemS5* :: ‹$'i\ fm \Rightarrow bool$› ($\vdash_{S5}$ - [*50*] *50*) **where**
‹$\vdash_{S5}\ p \equiv AxT \oplus AxB \oplus Ax4 \vdash p$›

abbreviation *AxTB4* :: ‹$'i\ fm \Rightarrow bool$› **where**
‹$AxTB4 \equiv AxT \oplus AxB \oplus Ax4$›

lemma *soundness-AxTB4*: ‹$AxTB4\ p \Longrightarrow$ *equivalence* $M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
⟨*proof*⟩

lemma $soundness_{S5}$: ‹$\vdash_{S5}\ p \Longrightarrow$ *equivalence* $M \Longrightarrow w \in \mathcal{W}\ M \Longrightarrow M,\ w \models p$›
⟨*proof*⟩

lemma *imply-completeness-S5*:
assumes *valid*: ‹$\forall (M :: ('i :: countable, 'i\ fm\ set)\ kripke).\ \forall w \in \mathcal{W}\ M.$
equivalence $M \longrightarrow (\forall q \in G.\ M,\ w \models q) \longrightarrow M,\ w \models p$›
shows ‹$\exists qs.\ set\ qs \subseteq G \wedge (AxTB4 \vdash imply\ qs\ p)$›

⟨*proof*⟩

lemma *completeness*$_{S5}$:
  assumes ‹$\forall$ (*M* :: (*'i* :: *countable*, *'i fm set*) *kripke*). $\forall$ *w* $\in$ $\mathcal{W}$ *M*. *equivalence M* $\longrightarrow$ *M*, *w* $\models$ *p*›
  shows ‹$\vdash_{S5}$ *p*›
  ⟨*proof*⟩

abbreviation ‹*valid*$_{S5}$ *p* $\equiv$ $\forall$ (*M* :: (*nat*, *nat fm set*) *kripke*). $\forall$ *w* $\in$ $\mathcal{W}$ *M*. *equivalence M* $\longrightarrow$ *M*, *w* $\models$ *p*›

theorem *main*$_{S5}$: ‹*valid*$_{S5}$ *p* $\longleftrightarrow$ $\vdash_{S5}$ *p*›
  ⟨*proof*⟩

corollary
  assumes ‹*equivalence M*› ‹*w* $\in$ $\mathcal{W}$ *M*›
  shows ‹*valid*$_{S5}$ *p* $\longrightarrow$ *M*, *w* $\models$ *p*›
  ⟨*proof*⟩

13.1 Traditional formulation

inductive *SystemS5'* :: ‹*'i fm* $\Rightarrow$ *bool*› ($\vdash_{S5}$'' - [*50*] *50*) **where**
  A1': ‹*tautology p* $\Longrightarrow$ $\vdash_{S5}$' *p*›
| *A2'*: ‹$\vdash_{S5}$' (*K i* (*p* $\longrightarrow$ *q*) $\longrightarrow$ *K i p* $\longrightarrow$ *K i q*)›
| *AT'*: ‹$\vdash_{S5}$' (*K i p* $\longrightarrow$ *p*)›
| *A5'*: ‹$\vdash_{S5}$' ($\neg$ *K i p* $\longrightarrow$ *K i* ($\neg$ *K i p*))›
| *R1'*: ‹$\vdash_{S5}$' *p* $\Longrightarrow$ $\vdash_{S5}$' (*p* $\longrightarrow$ *q*) $\Longrightarrow$ $\vdash_{S5}$' *q*›
| *R2'*: ‹$\vdash_{S5}$' *p* $\Longrightarrow$ $\vdash_{S5}$' *K i p*›

lemma *S5'-trans*: ‹$\vdash_{S5}$' ((*p* $\longrightarrow$ *q*) $\longrightarrow$ (*q* $\longrightarrow$ *r*) $\longrightarrow$ *p* $\longrightarrow$ *r*)›
  ⟨*proof*⟩

lemma *S5'-L*: ‹$\vdash_{S5}$' (*p* $\longrightarrow$ *L i p*)›
⟨*proof*⟩

lemma *S5'-B*: ‹$\vdash_{S5}$' (*p* $\longrightarrow$ *K i* (*L i p*))›
  ⟨*proof*⟩

lemma *S5'-KL*: ‹$\vdash_{S5}$' (*K i p* $\longrightarrow$ *L i p*)›
  ⟨*proof*⟩

lemma *S5'-map-K*:
  assumes ‹$\vdash_{S5}$' (*p* $\longrightarrow$ *q*)›
  shows ‹$\vdash_{S5}$' (*K i p* $\longrightarrow$ *K i q*)›
⟨*proof*⟩

lemma *S5'-map-L*:
  assumes ‹$\vdash_{S5}$' (*p* $\longrightarrow$ *q*)›
  shows ‹$\vdash_{S5}$' (*L i p* $\longrightarrow$ *L i q*)›

$\langle proof \rangle$

lemma *S5′-L-dual*: ‹$\vdash_{S5}{}'$ $(\neg\ L\ i\ (\neg\ p) \longrightarrow K\ i\ p)$›
$\langle proof \rangle$

lemma *S5′-4*: ‹$\vdash_{S5}{}'$ $(K\ i\ p \longrightarrow K\ i\ (K\ i\ p))$›
$\langle proof \rangle$

lemma *S5-S5′*: ‹$\vdash_{S5}\ p \Longrightarrow \vdash_{S5}{}'\ p$›
$\langle proof \rangle$

lemma *S5′-S5*:
  fixes p :: ‹$('i$ :: *countable*) *fm*›
  shows ‹$\vdash_{S5}{}'\ p \Longrightarrow \vdash_{S5}\ p$›
$\langle proof \rangle$

theorem $main_{S5}{}'$: ‹$valid_{S5}\ p \longleftrightarrow \vdash_{S5}{}'\ p$›
  $\langle proof \rangle$

14 Acknowledgements

The formalization is inspired by Berghofer's formalization of Henkin-style completeness.

- Stefan Berghofer: First-Order Logic According to Fitting. https://www.isa-afp.org/entries/FOL-Fitting.shtml

end

References

[1] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2001.

[2] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.