Non-classical logics

Lecture 20:

- Dynamic logic (Part 2)
- First-order modal logic

Viorica Sofronie-Stokkermans

sofronie@uni-koblenz.de

Until now

The idea of dynamic logic

- Annotated programs use formulas within programs
- Dynamic Logic uses programs within formulas
- Instead of "assert F" after program segment α, write: $[\alpha]F$

$\mapsto$ multi-modal logic

Propositional Dynamic Logic

Syntax

Prog set of programs

$\text{Prog}_0 \subseteq \text{Prog}$: set of atomic programs

Π: set of propositional variables

The set of formulae $\text{Fma}^{PDL}_{\text{Prog},\Pi}$ of (regular) propositional dynamic logic and the set of programs P_0 are defined by simultaneous induction as follows:

PDL: Syntax

Formulae:

$$
\begin{array}{lcll}
F, G, H & ::= & \bot & \text{(falsum)} \\
 & \mid & \top & \text{(verum)} \\
 & \mid & p & p \in \Pi_0 \ \text{(atomic formula)} \\
 & \mid & \neg F & \text{(negation)} \\
 & \mid & (F \wedge G) & \text{(conjunction)} \\
 & \mid & (F \vee G) & \text{(disjunction)} \\
 & \mid & (F \rightarrow G) & \text{(implication)} \\
 & \mid & (F \leftrightarrow G) & \text{(equivalence)} \\
 & \mid & [\alpha]F & \text{if } \alpha \in \text{Prog} \\
 & \mid & \langle\alpha\rangle\, F & \text{if } \alpha \in \text{Prog}
\end{array}
$$

Programs:

$$
\begin{array}{lcll}
\alpha, \beta, \gamma & ::= & \alpha_0 & \alpha_0 \in \text{Prog}_0 \ \text{(atomic program)} \\
 & \mid & F? & F \ \text{formula (test)} \\
 & \mid & \alpha; \beta & \text{(sequential composition)} \\
 & \mid & \alpha \cup \beta & \text{(non-deterministic choice)} \\
 & \mid & \alpha^* & \text{(non-deterministic repetition)}
\end{array}
$$

Semantics

A PDL structure $\mathcal{K} = (S, R(), I)$ is a multimodal Kripke structure with an accessibility relation for each atomic program. That is it consists of:

- a non-empty set S of states
- an interpretation $R() : \mathrm{Prog}_0 \to \mathcal{P}(S \times S)$ of atomic programs that assigns a transition relation $R(\alpha) \subseteq S \times S$ to each atomic program α
- an interpretation $I : \Pi \times S \to \{0, 1\}$

The interpretation of PDL relative to a PDL structure $\mathcal{K} = (S, R(), I)$ is defined by extending $R()$ to Prog and extensing I to $\mathrm{Fma}^{PDL}_{\mathrm{Prop}_0}$ by the following simultaneously inductive definition:

Interpretation of formulae/programs

$$
\begin{array}{lll}
val_{\mathcal{K}}(p, s) & = & I(p, s) \\
val_{\mathcal{K}}(\neg F, s) & = & \neg_{\text{Bool}} val_{\mathcal{K}}(F, s) \\
val_{\mathcal{K}}(F \wedge G, s) & = & val_{\mathcal{K}}(F, s) \wedge_{\text{Bool}} val_{\mathcal{K}}(G, s) \\
val_{\mathcal{K}}(F \vee G, s) & = & val_{\mathcal{K}}(F, s) \vee_{\text{Bool}} val_{\mathcal{K}}(G, s) \\
val_{\mathcal{K}}([\alpha]F, s) = 1 & \mathit{iff} & \text{ for all } t \in S \text{ with } (s, t) \in R(\alpha), val_{\mathcal{K}}(F, t) = 1 \\
val_{\mathcal{K}}(\langle\alpha\rangle\, F, s) = 1 & \mathit{iff} & \text{ for some } t \in S \text{ with } (s, t) \in R(\alpha), val_{\mathcal{K}}(F, t) = 1 \\
R([F?]) & = & \{(s, s) \mid val_{\mathcal{K}}(F, s) = 1\} \\
 & & (F? \text{ has the same meaning as: if } F \text{ then skip else do not terminate}) \\
R(\alpha \cup \beta) & = & R(\alpha) \cup R(\beta) \\
R(\alpha; \beta) & = & \{(s, t) \mid \text{ there exists } u \in S \ s.t.(s, u) \in R(\alpha) \text{ and } (u, t) \in R(\beta)\} \\
R(\alpha^*) & = & \{(s, t) \mid \text{ there exists } n \geq 0 \text{ and there exist } u_0, \ldots, u_n \in S \text{ with} \\
 & & s = u_0, y = u_n, (u_0, u_1), \ldots, (u_{n-1}, u_n) \in R(\alpha)\}
\end{array}
$$

Interpretation of formulae/programs

- $(\mathcal{K}, s)$ satisfies F (notation $(\mathcal{K}, s) \models F$) iff $val_{\mathcal{K}}(F, s) = 1$.
- F is valid in $\mathcal{K}$ (notation $\mathcal{K} \models F$) iff $(\mathcal{K}, s) \models F$ for all $s \in S$.
- F is valid (notation $\models F$) iff $\mathcal{K} \models F$ for all PDL-structures $\mathcal{K}$.

Axiom system for PDL

Axioms

$(D1)$	All propositional logic tautologies
$(D2)$	$[\alpha](A \rightarrow B) \rightarrow ([\alpha]A \rightarrow [\alpha]B)$
$(D3)$	$[\alpha](A \wedge B) \leftrightarrow [\alpha]A \wedge [\alpha]B$
$(D4)$	$[\alpha; \beta]A \leftrightarrow [\alpha][\beta]A$
$(D5)$	$[\alpha \cup \beta]A \leftrightarrow [\alpha]A \wedge [\beta]A$
$(D6)$	$[A?]B \leftrightarrow (A \rightarrow B)$
$(D7)$	$[\alpha^*]A \leftrightarrow A \wedge [\alpha][\alpha^*]A,$
$(D8)$	$[\alpha^*](A \rightarrow [\alpha]A) \rightarrow (A \rightarrow [\alpha^*]A]$

Inference rules

$MP \quad \frac{P, \quad P \rightarrow Q}{Q}$

$Gen \quad \frac{F}{[\alpha]F}$

We show that PDL is determined by PDL structures, and has the finite model property.

Soundness and Completeness of PDL

Proof similar to the proof in the case of the modal system K (with small differences)

Theorem. If the formula F is provable in the inference system for PDL then F is valid in all PDL structures.

Proof: Induction of the length of the proof, unsing the following facts:
1. The axioms are valid in every PDL structure. Easy computation.

2. If the premises of an inference rule are valid in a structure $\mathcal{K}$, the conclusion is also valid in $\mathcal{K}$.

(MP) If $\mathcal{K} \models F, \mathcal{K} \models F \rightarrow G$ then $\mathcal{K} \models G$ (follows from the fact that for every state s of $\mathcal{L}$ if $(\mathcal{K}, s) \models F, (\mathcal{K}, s) \models F \rightarrow G$ then $(\mathcal{K}, s) \models G$

(Gen) Assume that $\mathcal{K} \models F$. Then $(\mathcal{K}, s) \models F$ for every state s of $\mathcal{K}$.

Let t be a state of $\mathcal{K}$. $(\mathcal{K}, t) \models [\alpha]F$ if for all t' with $(t, t') \in R(\alpha)$ we have $(\mathcal{K}, t') \models F$. But under the assumption that $\mathcal{K} \models F$ the latter is always the case. This shows that $(\mathcal{K}, t) \models [\alpha]F$ for all t.

Soundness and Completeness of PDL

Theorem. If the formula F is is valid in all PDL structures then F is provable in the inference system for PDL.

Proof

Idea:
Assume that F is not provable in the inference system for PDL.

We show that:

(1) $\neg F$ is consistent with the set L of all theorems of PDL

(2) We can construct a "canonical" PDL structure $\mathcal{K}_L$ and a state w in this PDL structure such that $(\mathcal{K}, w) \models \neg F$.

Contradiction!

Consistent sets of formulae

Let L be a set of PDL formulae which:
(1) contains all propositional tautologies
(2) contains axiom PDL
(3) is closed under modus ponens and generalization
(4) is closed under instantiation

Definition. A subset $F \subseteq L$ is called L-inconsistent iff there exist formulae $A_1, \ldots, A_n \in F$ such that

$$(\neg A_1 \vee \cdots \vee \neg A_n) \in L$$

F is called L-consistent iff it is not L-inconsistent.

Definition. A consistent set F of PDL formulae is called maximal L-consistent if for every formula A wither $A \in F$ or $\neg A \in F$.

Consistent sets of formulae

Let L be as before. In what follows we assume that L is consistent.

Theorem. Let F be a maximal L-consistent set of formulae. Then:

(1) For every formula A, either $A \in F$ or $\neg A \in F$, but not both.

(2) $A \vee B \in F$ iff $A \in F$ or $B \in F$

(3) $A \wedge B \in F$ iff $A \in F$ and $B \in F$

(4) $L \subseteq F$

(5) F is closed under Modus Ponens

Theorem. Every consistent set F of formulae is contained in a maximally consistent set of formulae.

Lemma. If F is not provable in PDL then $\neg F$ is consistent with the set L of all theorems of PDL, so it is contained in a maximally conststent set of formulae $W_{\neg F}$.

Canonical models

Goal: Assume F is not a PDL theorem. Construct a PDL structure $\mathcal{K}$ and a state w of $\mathcal{K}$ such that $(\mathcal{K}, w) \models \neg F$.

States:

State of $\mathcal{K}$: maximal consistent set of formulae.

Intuition: $(\mathcal{K}, W) \models \phi$ iff $\phi \in W$.

Then: $(\mathcal{K}, W_{\neg F}) \models \neg F$

Accessibility relation:

Intuition:
$(\mathcal{K}, W) \models [\alpha]F$ iff for all $W', ((W, W') \in R(\alpha) \rightarrow (\mathcal{K}, W') \models F$
$[\alpha]F \in W$ iff for all $W', ((W, W') \in R(\alpha) \rightarrow F \in W')$

$(W, W') \in R(\alpha)$ iff $W' \supseteq \{F \mid [\alpha]F \in W\}$

Canonical models

Theorem. $\mathcal{K}$ satisfies all PDL structure conditions except $R(\alpha^*) \subseteq (R(\alpha))^*$.

Proof: By direct checking.

Example: $R(\alpha;\beta) \subseteq R(\alpha) \circ R(\beta)$
Assume $(W, W') \in R(\alpha;\beta)$. Then $W' \subseteq \{F \mid [\alpha;\beta]F \in W\}$.

We want to show that there exists W_0 with $(W, W_0) \in R(\alpha)$ and $(W_0, W') \in R(\beta)$.

- $(W, W_0) \in R(\alpha)$ iff $\{A \mid [\alpha]A \in W\} \subseteq W_0$
- $(W_0, W') \in R(\beta)$ iff $\{B \mid [\beta]B \in W_0\} \subseteq W'$ iff $\{\neg[\beta]D \mid D \notin W'\} \subseteq W_0$.

It is sufficient to show that $W_0 = \{B \mid [\alpha]B \in W\} \cup \{\neg[\beta]D \mid D \notin W'\}$ is PDL-consistent.

For this, the PDL-theorem $[\alpha][\beta]A \rightarrow [\alpha][\beta]A$ is used.

Canonical models

Proof: (ctd.) We show that $\{A \mid [\alpha]A \in W\} \cup \{\neg[\beta]B \mid B \not\in W'\}$ is PDL-consistent.

Assume that the set is not PDL consistent. Then there is a theorem

$$\vdash A_1 \wedge \cdots \wedge A_m \wedge \neg[\beta]B_1 \wedge \ldots \neg[\beta]B_n \rightarrow \perp$$

where $[\alpha]A_i \in W$ and $B_j \not\in W'$. Let $B = B_1 \vee \cdots \vee B_n$.

Since $\vdash [\beta]B_1 \vee \cdots \vee [\beta]B_n \rightarrow [\beta]B$ it follows that $\vdash A_1 \wedge \cdots \wedge A_m \rightarrow [\beta]B$ hence:

$$\vdash [\alpha](A_1 \wedge \cdots \wedge A_m) \rightarrow [\alpha][\beta]B$$

and since $\vdash [\alpha]A_1 \wedge \cdots \wedge [\alpha]A_m \rightarrow [\alpha](A_1 \wedge \cdots \wedge A_m)$ we showed that

$$\vdash [\alpha]A_1 \wedge \cdots \wedge [\alpha]A_m \rightarrow [\alpha][\beta]B$$

Using the PDL-theorem $[\alpha][\beta]B \rightarrow [\alpha;\beta]B$ it then follows that

$$\vdash [\alpha]A_1 \wedge \cdots \wedge [\alpha]A_m \rightarrow [\alpha;\beta]B$$

Since $[\alpha]A_i \in W$ and W is maximally consistent it follows that $[\alpha;\beta]B \in W$, hence $B = B_1 \vee \cdots \vee B_n \in W'$. But then (as W' maximally consistent) $B_j \in W'$ for some j which is a contradiction.

Canonical models

Theorem. Assume F is not a PDL theorem. We can construct a PDL structure $\mathcal{K}'$ and a state w of $\mathcal{K}'$ such that $(\mathcal{K}', w) \models \neg F$.

Proof. To obtain a PDL structure that falsifies F we will collapse $\mathcal{K}$ by a suitable Γ that contains F. The closure rules for Γ that will be needed are:

- Γ is closed under subformulae;
- $[B?]D \in \Gamma$ implies $B \in \Gamma$;
- $[\alpha; \beta]B \in \Gamma$ implies $[\alpha][\beta]B \in \Gamma$;
- $[\alpha \cup \beta]B \in \Gamma$ implies $[\alpha]B, [\beta]B \in \Gamma$;
- $[\alpha^*]B \in \Gamma$ implies $[\alpha][\alpha^*]B \in \Gamma$

A set Γ satisfying these conditions will be called closed.

Completeness/Decidability of PDL

Theorem. If Γ is the smallest closed set containing a given formula F, then Γ is finite.

Proof. The point is to show that closing Subformulae(F) under the above rules produces only finitely many new formulae.

Define a formula to be boxed if it is prefixed by a modal connective, i.e. is of the form $[\alpha]B$ for some α and B. Each time we apply a closure rule, new boxed formulae appear on the right side of the rule, and further rules may apply to these new formulae.

But observe that the programs α indexing prefixes $[\alpha]$ on the right side are in all cases shorter in length than those indexing the prefix on the left of the rule in question. Hence we will eventually produce only atomic prefixes on the right, and run out of rules to apply.

Completeness/Decidability of PDL

Having determined that Γ, the smallest closed set containing F, is finite, we identify the states which satisfy the same formulae in Γ:

Fix a model $\mathcal{K} = (S, R, I)$ and a set $\Gamma \subseteq Fma_\Sigma$ that is closed under subformulae, i.e. $B \in \Gamma$ implies Subformulae$(B) \subseteq \Gamma$.

For each $s \in S$, define

$$\Gamma_s = \{B \in \Gamma \mid (\mathcal{K}, s) \models B\}$$

and put $s \sim_\Gamma t$ iff $\Gamma_s = \Gamma_t$,

Then $s \sim_\Gamma t$ iff for all $B \in \Gamma$, $(\mathcal{K}, s) \models B$ iff $(\mathcal{K}, t) \models B$.

Fact: $\sim_\Gamma$ is an equivalence relation on S.

Let $[s] = \{t \mid s \sim_\Gamma t\}$ be the $\sim_\Gamma$-equivalence class of s.

Let $S_\Gamma := \{[s] \mid s \in S\}$ be the set of all such equivalence classes.

Decidability/Completeness

Goal: $(\mathcal{K}, s) \models A \quad \mapsto \quad (\mathcal{K}', s') \models A, \mathcal{K}' = (S', R', I')$.

Step 1: $S' := S_\Gamma$, where $\Gamma = \text{Subformulae}(S)$

Step 2: $I' : (\Pi \cap \Gamma) \times S' \to \{0, 1\}$ def. by $I'(P, [s]) = I(P, s)$

Step 3: $R'(\alpha)$ def. e.g. by: $([s], [t]) \in R'(\alpha)$ iff $\exists s' \in [s], \exists t' \in [t]$: $(s', t') \in R(\alpha)$

Theorem: $\mathcal{K}'$ is a PDL structure and it is a filtration of $\mathcal{K}$, i.e. has the properties:

(F1) If $sR(\alpha)t$ then $[s]R'(\alpha)[t]$

(F2) If $[s]R'(\alpha)[t]$ then for all formulae B,
if $[\alpha]B \in \Gamma$ and $(\mathcal{K}, s) \models [\alpha]B$ then $(\mathcal{K}, t) \models B$

Since $(\mathcal{K}, W_{\neg F}) \models \neg F$ it can easily be seen that $(\mathcal{K}', [W_{\neg F}]) \models \neg F$. $\mapsto$ completeness

Lemma. If Γ is finite, then S_Γ is finite and has at most 2^n elements, where n is the number of elements of Γ. $\mapsto$ decidability

Conclusions

PDL is decidable (it has the finite model propety).

Proof calculi for PDL exist (e.g. sequent calculi, tableau calculi)

For really reasoning about programs, often *first order dynamic logic* is needed (undecidable)

Nevertheless, many systems used for verification use sequent or tableau calculi also for first order dynamic logic.

First-order modal logic

First-order modal logic

We introduce first-order modal logic and consider its relationship to classical first-order logic.

First-order modal logic

Syntax

Semantics

Syntax

Given:

A signature $\Sigma = (\Omega, \Pi)$,

A set X of variables

Syntax

Given:

A signature $\Sigma = (\Omega, \Pi)$,

A set X of variables

Terms are defined as for classical logic

Atomic formulae are defined as for classical logic

General first-order formulae

$$
\begin{array}{rcll}
F, G, H & ::= & \bot & \text{(falsum)} \\
 & \mid & \top & \text{(verum)} \\
 & \mid & A & \text{(atomic formula)} \\
 & \mid & \neg F & \text{(negation)} \\
 & \mid & (F \wedge G) & \text{(conjunction)} \\
 & \mid & (F \vee G) & \text{(disjunction)} \\
 & \mid & (F \rightarrow G) & \text{(implication)} \\
 & \mid & (F \leftrightarrow G) & \text{(equivalence)} \\
 & \mid & \Box F & \\
 & \mid & \Diamond F & \\
 & \mid & \forall x\ F & \text{(universal quantification)} \\
 & \mid & \exists x\ F & \text{(existential quantification)}
\end{array}
$$

Semantics

A Kripke structure $K = (S, R, I)$ consists of Kripke frame $F = (S, R)$ and a mapping I that assigns to each world $s \in S$ a first-order structure

$$I(s) = (U_{I(s)}, \{f_{I(s)}\}_{f \in \Omega}, \{p_{I(s)}\}_{p \in \Pi})$$

such that, for each $s, t \in S$ with sRt, $I(s)$ is a substructure of $I(t)$, i.e.:

- the universe of $I(s)$ is a subset of the universe of $I(t)$ (monotonicity), and
- the structures $I(s)$ and $I(t)$ agree on the interpretation of all function symbols on the (smaller) universe of $I(s)$.

A Kripke structure (S, R, I) is called Kripke structure with constant domain if all the models $\{I(s) \mid s \in S\}$ are required to have the same universe.

Semantics

Interpretation of quantified modal formulae

$(\mathcal{K}, s) \models p(t_1, \dots, t_k)$ iff $I(s) \models p(t_1, \dots, t_k)$ for atomic formulae $p(t_1, \dots, t_k)$

$(\mathcal{K}, s) \models F \wedge G$ iff $(\mathcal{K}, s) \models F$ and $(\mathcal{K}, s) \models G$

$(\mathcal{K}, s) \models F \vee G$ iff $(\mathcal{K}, s) \models F$ or $(\mathcal{K}, s) \models G$

$(\mathcal{K}, s) \models F \rightarrow G$ iff $(\mathcal{K}, s) \not\models F$ or $(\mathcal{K}, s) \models G$

$(\mathcal{K}, s) \models \neg F$ iff $(\mathcal{K}, s) \not\models F$

$(\mathcal{K}, s) \models \Box F$ iff for all t with $R(s, t)$, $(\mathcal{K}, s) \models F$

$(\mathcal{K}, s) \models \Diamond F$ iff there exists t with $R(s, t)$ and $(\mathcal{K}, s) \models F$

$(\mathcal{K}, s) \models \forall x F(x)$ iff for all $d \in U_{I(s)}$, $(\mathcal{K}, s) \models F(d)$

$(\mathcal{K}, s) \models \exists x F(x)$ iff there exists $d \in U_{I(s)}$, $(\mathcal{K}, s) \models F(d)$

Semantics

Interpretation of quantified modal formulae

$(\mathcal{K}, s) \models p(t_1, \dots, t_k)$ iff $I(s) \models p(t_1, \dots, t_k)$ for atomic formulae $p(t_1, \dots, t_k)$

$(\mathcal{K}, s) \models F \wedge G$ iff $(\mathcal{K}, s) \models F$ and $(\mathcal{K}, s) \models G$

$(\mathcal{K}, s) \models F \vee G$ iff $(\mathcal{K}, s) \models F$ or $(\mathcal{K}, s) \models G$

$(\mathcal{K}, s) \models F \rightarrow G$ iff $(\mathcal{K}, s) \not\models F$ or $(\mathcal{K}, s) \models G$

$(\mathcal{K}, s) \models \neg F$ iff $(\mathcal{K}, s) \not\models F$

$(\mathcal{K}, s) \models \Box F$ iff for all t with $R(s, t)$, $(\mathcal{K}, s) \models F$

$(\mathcal{K}, s) \models \Diamond F$ iff there exists t with $R(s, t)$ and $(\mathcal{K}, s) \models F$

$(\mathcal{K}, s) \models \forall x F(x)$ iff for all $d \in U_{I(s)}$, $(\mathcal{K}, s) \models F(d)$

$(\mathcal{K}, s) \models \exists x F(x)$ iff there exists $d \in U_{I(s)}$, $(\mathcal{K}, s) \models F(d)$

Semantics

Goal: formalize the last statements using variable assignments, as for first-order logic

A (variable) assignment, also called a valuation (over a given Σ-algebra $\mathcal{A}$), is a map $\beta : X \to \mathcal{A}$.

Semantics

Goal: formalize the last statements using variable assignments, as for first-order logic

A (variable) assignment, also called a valuation (over a given Σ-algebra $\mathcal{A}$), is a map $\beta : X \to \mathcal{A}$.

Difficulty: the domains of the structures are different from world to world.

Semantics

In varying domain semantics, quantifiers may possibly refer to a different set of objects, depending on the world.

In constant domain semantics, quantifiers refer to the same set of objects (same universe U) in all worlds.

Variable assignment: $\beta : X \rightarrow U$.

Evaluation of quantified formulae as in classical first-order logic.

Proof calculi

Tableau calculi

Translation to first-order logic and resolution

Proof calculi

Tableau calculi

need to take into account γ and δ rules
(as for classical first-order logic)

Translation to first-order logic and resolution

out of the scope of this lecture

First-order Dynamic Logic

First-order dynamic logic (DL) extends PDL (although DL had been developed first) to a first-order logic and gives concrete atomic programs with specific effects, as opposed to abstract atomic programs with unknown effects as in PDL.

First-order Dynamic Logic

Definition. Let X be a set of variables. The set Fma^{DL} of formulas of dynamic logic and the set of programs Prog are defined by simultaneous induction as:

- $\bot, \top \in \text{Fma}^{DL}$ (propositional constants)
- All instances of formulas of first-order logic are in Fma^{DL}
- If $F, G \in \text{Fma}^{DL}$ then $\neg F, (F \wedge G), (F \vee G) \in \text{Fma}^{DL}$
- If $F \in \text{Fma}^{DL}$ and $x \in X$ is a variable then $\forall x F, \exists x F \in \text{Fma}^{DL}$
- If $F \in \text{Fma}^{DL}$, $\alpha \in \text{Prog}$ then $[\alpha]F, \langle\alpha\rangle\, F \in \text{Fma}^{DL}$
- $(x := t) \in \text{Prog}$ are atomic programs for variables $x \in X$ and terms t.
- If $F \in \text{Fma}^{DL}$ then $F? \in \text{Prog}$.
- If $\alpha, \beta \in \text{Prog}$ then $\alpha; \beta, \alpha \cup \beta, \alpha^* \in \text{Prog}$.

First-order Dynamic Logic

The semantics of DL is extended from that of PDL in the obvious way where the set of states is chosen to be $W := D^X$, i.e., the set of assignments $s : X \to D$ of elements of the domain D (of the first-order structure) to the variables X.

Predicate symbols, function symbols, and terms are interpreted as usual in first-order logic.

Because there are no other atomic programs, we only need to specify the accessibility relation belonging to an assignment $x := t$:

$$R(x := t) = \{(s, s') \mid s'(x) = s(t) \text{ and } s'(z) = s(z) \text{ for all } z \neq x\}$$

First-order Dynamic Logic

DL can inherit the axioms of first-order logic and of PDL. One typical axiom that DL needs in addition is an axiom that relates assignment to substitution:

$$(D15) \qquad \langle x := t \rangle\, F \leftrightarrow F_x^t$$

It says that formula F is true after assigning t to x if and only if F is true after substituting the new value t for x.

First-order Dynamic Logic

DL cannot be decidable because it includes first-order logic, where validity is only semidecidable.

But DL does not have a sound and complete effective calculus. Note here that DL formulas can state the halting problem for Turing machines. Nevertheless, there are proofs showing that DL has a relatively complete or arithmetically complete proof calculus.

Temporal logic

Motivation

The purpose of temporal logic (TL) is:

- reasoning about time (in philosophy), and
- reasoning about the behaviour of systems evolving over time (in computer science).

How to define a TL?

To define a temporal logic (TL), we need to specify:

- the language for talking about time or temporal systems;
- our model of time.

Motivation

What model of time should we use?

What is the structure of time?

Motivation

What model of time should we use?

What is the structure of time?

A very liberal definition:

A flow of time is a pair $(T, <)$, where T is a non-empty set of time points, and $<$ is an irreflexive and transitive binary relation on T.

Depending on the intended application, we often require additional properties. One of the most fundamental decisions is whether or not time should be linear.

$(T, <)$ is linear if, for all $x, y \in T$ with $x \neq y$, we have $x < y$ or $y < x$.

Models of time

Important additional properties for linear flows of time:

Boundedness: We have four options by combining:

- *Bounded to the past:* there exists an $x \in T$ such that $x \leq y$ for all $y \in T$ (genesis).
- *Bounded to the future:* there exists a an $x \in T$ such that $y \leq x$ for all $y \in T$ (doomsday).

Discreteness: Existence of direct predecessors and successors:

- If $x \in T$ is not genesis, then there exists a $y \in T$ such that $y < x$ and $y < z < x$ holds for no $z \in T$.
- If $x \in T$ is not doomsday, then there exists a $y \in T$ such that $x < y$ and $x < z < y$ holds for no $z \in T$.

It can be seen that one does not follow from the other.

Models of time

Important additional properties for linear flows of time:

Density: For all $x, y \in T$ with $x < y$, there is a $z \in T$ such that $x < z < y$.

Dedekind completeness: Any non-empty subset $S \subseteq T$ that has an upper bound has a least upper bound:

Definitions:

Upper bound for S: $x \in T$ with $y \leq x$ for all $y \in S$;

Least upper bound for S: upper bound x for S such that there is no $x' \in T$ with $x' < x$ and x' upper bound for S.

Models of time

The following are among the most natural linear flows of time:

- **The natural numbers $\mathbb{N}$ with the usual order $<$.**

  Linear, discrete, bounded to the past, not bounded to the future.

Note that other flows of time have these properties as well:

$T := \mathbb{N} \times \{0\} \cup \mathbb{Z} \times \{1\}$, where:

$(x, a) < (y, b)$ if (i) $a < b$ or (ii) $a = b$ and $x < y$.

NOTE: above example not Dedekind complete.

Models of time

The following are among the most natural linear flows of time:

- **The rational numbers $\mathbb{Q}$.**

  A natural dense flow of time, though with gaps (e.g. π).

  The unique countable linear dense flow of time without endpoints (up to isomorphism).

- **The real numbers $\mathbb{R}$.**

  Up to isomorphism, the unique dense, Dedekind-complete flow of time without end points that is separable:

  There exists a countable subset $D \subseteq T$ such that, for all $x, y \in T$ with $x < y$, there is a $z \in D$ with $t < z < u$.

Models of time

The alternative to linear time is branching time.

Time can be:

- **Branching to the future** reflecting that there are many possible futures;
- **Branching to the past** reflecting that many different histories are considered possible (due to incomplete knowledge).

Branching to the future and linear to the past is the most popular option

for each $x \in T$, the set $\{y \in T | y < x\}$ is linearly ordered by $<$.

We can identify additional properties similar to the linear case. Usually, branching time is assumed to be discrete and has a genesis.