Mathematische Logik
SS 2009

Prof. Dr. Erich Grädel

Mathematische Grundlagen der Informatik
RWTH Aachen

Inhaltsverzeichnis

3 Modallogik, temporale Logiken und monadische Logik

Modale und temporale Logiken sind geeignete logische Systeme, um Aussagen über Transitionssysteme zu formalisieren. Sie bieten ein gutes Gleichgewicht zwischen vernünftiger Ausdrucksstärke und günstigen algorithmischen Eigenschaften. Dies macht sie für Anwendungen in der Informatik sehr interessant.

3.1 Syntax und Semantik der Modallogik

Modallogiken formalisieren Aussagen über Transitionssysteme von einer internen, lokalen Perspektive her. Die Modallogik erweitert die Aussagenlogik um einstellige Modaloperatoren, mit welchen man aus einer Formel ψ neue Formeln der Form $\langle a \rangle \psi$ bzw. $[a]\psi$ bildet, für alle a aus einer Menge von *Aktionen*.

Definition 3.1. Die Menge ML der modallogischen Formeln (mit Aktionen aus A und atomaren Eigenschaften P_i für $i \in I$) ist induktiv definiert wie folgt:

- Alle aussagenlogischen Formeln mit Aussagenvariablen P_i gehören zu ML.
- Wenn $\psi, \varphi \in$ ML, dann auch $\neg\psi$, $(\psi \vee \varphi)$, $(\psi \wedge \varphi)$ und $(\psi \to \varphi)$.
- Wenn $\psi \in$ ML und $a \in A$, dann gehören auch $\langle a \rangle \psi$ und $[a]\psi$ zu ML.

Notation. Wenn nur eine Aktion a vorhanden ist, also $|A| = 1$, dann schreiben wir $\Diamond\psi$ (sprich „Diamond ψ“ oder „möglicherweise ψ“) und $\Box\psi$ (sprich „Box ψ “ oder „notwendigerweise ψ“) anstelle von $\langle a \rangle \psi$ und $[a]\psi$.

Definition 3.2. Ein *Transitionssystem* oder eine *Kripkestruktur* mit Aktionen aus A und atomaren Eigenschaften $\{P_i : i \in I\}$ ist eine Struktur

$$\mathcal{K} = (V, (E_a)_{a\in A}, (P_i)_{i\in I})$$

mit Universum V (dessen Elemente Zustände oder Welten genannt werden), zweistelligen Relationen $E_a \subseteq V \times V$ $(a \in A)$ (welche Transitionen zwischen Zuständen beschreiben) und einstelligen Relationen (Eigenschaften der Zustände) $P_i \subseteq V$ $(i \in I)$. Statt $(u, v) \in E_a$ schreiben wir oft auch $u \xrightarrow{a} v$.

Man kann sich ein Transitionssystem als einen Graphen mit beschrifteten Knoten und Kanten vorstellen. Die Elemente des Universums sind Knoten, die einstelligen Relationen entsprechen den Beschriftungen der Knoten und die zweistelligen Relationen den beschrifteten Kanten.

Definition 3.3. Sei $\mathcal{K} = (V, (E_a)_{a\in A}, (P_i)_{i\in I})$ ein Transitionssystem, $\psi \in$ ML eine Formel und v ein Zustand von $\mathcal{K}$. Die *Modellbeziehung* $\mathcal{K}, v \models \psi$ (d.h. ψ gilt im Zustand v von $\mathcal{K}$) ist induktiv wie folgt definiert:

(1) $\mathcal{K}, v \models P_i \iff v \in P_i$.
(2) Die Bedeutungen von $\neg\psi$, $(\psi \wedge \varphi)$, $(\psi \vee \varphi)$ und $(\psi \to \varphi)$ sind wie üblich.
(3) $\mathcal{K}, v \models \langle a\rangle\psi$, wenn ein w existiert mit $(v, w) \in E_a$ und $\mathcal{K}, w \models \psi$.
(4) $\mathcal{K}, v \models [a]\psi$, wenn für alle w mit $(v, w) \in E_a$ gilt, dass $\mathcal{K}, w \models \psi$.

Wie schon eingangs erwähnt, haben wir hier im Gegensatz zu FO eine lokale Sichtweise der Modellbeziehung. Von einem bestimmten Zustand v ausgehend, wird eine Formel ψ an diesem v evaluiert. Die Modaloperatoren $\langle a\rangle$ und $[a]$ können als eingeschränkte Varianten der Quantoren $\exists$ und $\forall$ (Quantifizierung entlang von Transitionen) gesehen werden.

Wir können auch jeder Formel ψ und jedem Transitionssystem $\mathcal{K}$ die Extension

$$[\![\psi]\!]^{\mathcal{K}} := \{v : \mathcal{K}, v \models \psi\}$$

zuordnen, also die Menge der Zustände v, an denen ψ in $\mathcal{K}$ gilt. Die Modellbeziehung ist dann durch folgende Regeln gegeben (welche natürlich zu den in Definition 3.3 gegeben Regeln äquivalent sind):

(1) $[\![P_i]\!]^{\mathcal{K}} = P_i$.
(2) $[\![\neg\psi]\!]^{\mathcal{K}} := V \setminus [\![\psi]\!]^{\mathcal{K}}$
$[\![\psi \wedge \varphi]\!]^{\mathcal{K}} := [\![\psi]\!]^{\mathcal{K}} \cap [\![\varphi]\!]^{\mathcal{K}}$
$[\![\psi \vee \varphi]\!]^{\mathcal{K}} := [\![\psi]\!]^{\mathcal{K}} \cup [\![\varphi]\!]^{\mathcal{K}}$
$[\![\psi \to \varphi]\!]^{\mathcal{K}} := (V \setminus [\![\psi]\!]^{\mathcal{K}}) \cup [\![\varphi]\!]^{\mathcal{K}}$.
(3) $[\![\langle a\rangle\psi]\!]^{\mathcal{K}} := \{v : vE_a \cap [\![\psi]\!]^{\mathcal{K}} \neq \emptyset\}$.
(4) $[\![[a]\psi]\!]^{\mathcal{K}} := \{v : vE_a \subseteq [\![\psi]\!]^{\mathcal{K}}\}$.

Dabei ist $vE_a := \{w : (v, w) \in E_a\}$ die Menge aller a-Nachfolger von v.

Einbettung der Modallogik in die Prädikatenlogik. Formal ist die Modallogik eine Erweiterung der Aussagenlogik. Oft ist es aber weitaus zweckmäßiger, ML in die Prädikatenlogik zu übersetzen und sie damit als Fragment von FO aufzufassen. Dies liegt schon deshalb nahe, weil die Modallogik über Transitionssysteme, also Strukturen, spricht.

Die folgende Übersetzung zeigt, dass man dabei mit FO-Formeln auskommt, die nur zwei Variablen x und y (diese allerdings mehrfach quantifiziert) verwenden.

Definition 3.4. FO^2, das *Zwei-Variablen-Fragment* von FO, ist die Menge aller relationalen FO-Formeln, welche nur zwei Variablen x und y enthalten.

Beispiel 3.5. Wir wollen ausdrücken, dass es (in einem gegebenen Transitionssystem) vom aktuellen Zustand x aus einen a-Pfad der Länge 5 zu einem Zustand gibt, der in der Menge P liegt. In ML wird dies durch die Formel $\langle a\rangle\langle a\rangle\langle a\rangle\langle a\rangle\langle a\rangle P$ formalisiert. Die naheliegendste Weise, dieselbe Aussage in FO auszudrücken, führt zu der Formel

$$\psi(x) := \exists y_1 \cdots \exists y_5 (E_a x y_1 \wedge \bigwedge_{i=1}^{4} E_a y_i y_{i+1} \wedge P y_5),$$

welche sechs Variablen verwendet. Wir können aber denselben Sachverhalt auch mit nur zwei Variablen ausdrücken durch die Formel

$$\psi'(x) := \exists y(Exy \wedge \exists x(Eyx \wedge \exists y(Exy \wedge \exists x(Eyx \wedge \exists y(Exy \wedge Py))))).$$

Satz 3.6. Zu jeder Formel $\psi \in \mathrm{ML}$ gibt es eine Formel $\psi^*(x)$ in FO^2, so dass für alle Transitionssysteme $\mathcal{K}$ und alle Zustände v von $\mathcal{K}$ gilt:

$$\mathcal{K}, v \models \psi \iff \mathcal{K} \models \psi^*(v).$$

Beweis. Wir geben eine Tabelle an, nach der jede Formel $\psi \in \mathrm{ML}$ induktiv in eine Formel $\psi^*(x) \in \mathrm{FO}^2$ übersetzt werden kann. Mit $\psi^*(y)$ sei hier die Formel bezeichnet, die man aus $\psi^*(x)$ erhält indem man alle (freien und gebundenen) Vorkommen von x durch y ersetzt, und umgekehrt:

$$\begin{aligned}
P_i &\mapsto P_i x \\
\neg\psi &\mapsto \neg\psi^*(x) \\
(\psi \circ \varphi) &\mapsto (\psi^*(x) \circ \varphi^*(x)) \text{ für } \circ \in \{\wedge, \vee, \rightarrow\} \\
\langle a\rangle\psi &\mapsto \exists y(E_a xy \wedge \psi^*(y)) \\
[a]\psi &\mapsto \forall y(E_a xy \rightarrow \psi^*(y))
\end{aligned}$$

Q.E.D.

Erfüllbarkeit, Gültigkeit, Äquivalenz. Analog zu Aussagenlogik und Prädikatenlogik definieren wir: Eine Formel $\psi \in \mathrm{ML}$ ist *erfüllbar*, wenn ein Transitionssystem $\mathcal{K}$ und ein Zustand v von $\mathcal{K}$ existiert, so dass $\mathcal{K}, v \models \psi$. Sie ist *gültig*, wenn $\mathcal{K}, v \models \psi$ für alle $\mathcal{K}$ und alle v. Zwei Formeln ψ, φ sind *äquivalent*, kurz $\psi \equiv \varphi$, wenn $[\![\psi]\!]^{\mathcal{K}} = [\![\varphi]\!]^{\mathcal{K}}$ für alle zu ψ und φ passenden Transitionssysteme $\mathcal{K}$.

Beispiel 3.7. Für alle Formeln $\psi \in \mathrm{ML}$ und alle Aktionen a gilt:

(1) $\langle a\rangle\psi \rightarrow [a]\psi$ ist erfüllbar, aber nicht gültig.
(2) $[a](\psi \rightarrow \varphi) \rightarrow ([a]\psi \rightarrow [a]\varphi)$ ist gültig.
(3) $[a]\psi \equiv \neg\langle a\rangle\neg\psi$ (Dualität von $\langle a\rangle$ und $[a]$).

NEGATIONSNORMALFORM. Wie für Aussagenlogik und Prädikatenlogik gibt es auch für die Modallogik Normalformen. Nützlich ist insbesondere die Negationsnormalform. Jede Formel $\psi \in \mathrm{ML}$ ist äquivalent zu einer Formel, in der die Negation nur auf atomare Eigenschaften P_i angewandt wird. Dies folgt unmittelbar aus den de Morganschen Gesetzen und der Dualität von $\langle a \rangle$ und $[a]$.

Übung 3.1. Gilt für ML das Analogon des Satzes über die Pränex-Normalform?

3.2 Bisimulation

Einer der wichtigsten Begriffe bei der Analyse von Modallogiken ist die Bisimulation. Mit ihr wollen wir die Ununterscheidbarkeit von Kripkestrukturen bezüglich Formeln aus ML untersuchen.

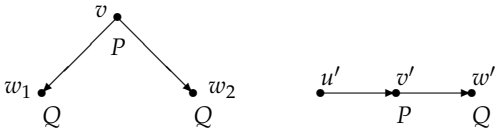
Definition 3.8. Eine *Bisimulation* zwischen zwei Transitionssystemen $\mathcal{K} = (V, (E_a)_{a\in A}, (P_i)_{i\in I})$ und $\mathcal{K}' = (V', (E'_a)_{a\in A}, (P'_i)_{i\in I})$ ist eine Relation $Z \subseteq V \times V'$, so dass für alle $(v, v') \in Z$ gilt:

(1) $v \in P_i \iff v' \in P'_i$ für alle $i \in I$.

(2) *Hin:* Für alle $a \in A$, $w \in V$ mit $v \xrightarrow{a} w$ existiert ein $w' \in V'$ mit $v' \xrightarrow{a} w'$ und es ist $(w, w') \in Z$.

Her: Für alle $a \in A$, $w' \in V'$ mit $v' \xrightarrow{a} w'$ existiert ein $w \in V$ mit $v \xrightarrow{a} w$ und es ist $(w, w') \in Z$.

Beispiel 3.9. $Z = \{(v, v'), (w_1, w'), (w_2, w')\}$ ist eine Bisimulation zwischen den beiden folgenden Transitionssystemen.



Definition 3.10. Seien $\mathcal{K}, \mathcal{K}'$ Kripkestrukturen und $u \in V$, $u' \in V'$. $(\mathcal{K}, u)$ und $(\mathcal{K}', u')$ sind *bisimilar* (kurz: $\mathcal{K}, u \sim \mathcal{K}', u'$), wenn eine Bisimulation Z zwischen $\mathcal{K}$ und $\mathcal{K}'$ existiert, so dass $(u, u') \in Z$.

Das Bisimulationsspiel. Die Bisimililarität zweier Transitionssysteme kann auch auf spieltheoretische Weise durch ein Bisimulationsspiel beschrieben werden. Das Spiel wird von zwei Spielern auf zwei Kripkestrukturen $\mathcal{K}$ und $\mathcal{K}'$, auf denen sich je ein Spielstein befindet, gespielt. In der Anfangsposition liegen die Steine auf u bzw. u'. Die Spieler ziehen nun abwechselnd nach folgenden Regeln:

Spieler I bewegt den Stein in $\mathcal{K}$ oder $\mathcal{K}'$ entlang einer Transition zu einem neuem Zustand: von v entlang $v \xrightarrow{a} w$ zu w oder von v' entlang $v' \xrightarrow{a} w'$ zu w'. Spielerin II antwortet mit einer entsprechenden Bewegung in der anderen Struktur: $v' \xrightarrow{a} w'$ oder $v \xrightarrow{a} w$. Wenn ein Spieler nicht ziehen kann, verliert er. D.h. Spieler I verliert, wenn er zu einem Knoten kommt, von dem keine Transitionen mehr wegführen und Spielerin II verliert, wenn sie nicht mehr mit der entsprechenden Aktion antworten kann. Am Anfang und nach jedem Zug wird überprüft, ob für die aktuelle Position v, v' gilt: $v \in P_i \iff v' \in P'_i$ für alle $i \in I$. Wenn nicht, dann hat I gewonnen, ansonsten geht das Spiel weiter. II gewinnt, wenn sie nie verliert.

Uns interessieren nicht primär einzelne Partien, sondern ob einer der Spieler eine Gewinnstrategie hat. Wir sagen, II gewinnt das Bisimulationsspiel auf $(\mathcal{K}, \mathcal{K}')$ von (u, u') aus, wenn es eine Strategie für II gibt, mit der sie nie verliert, was auch immer I zieht. Eine derartige Strategie entspricht genau einer Bisimulation. Also gilt:

Lemma 3.11. II gewinnt genau dann das Bisimulationsspiel auf $\mathcal{K}, \mathcal{K}'$ von (u, u'), wenn $\mathcal{K}, u \sim \mathcal{K}', u'$.

Wir können die Analyse noch etwas verfeinern, wenn wir die Anzahl der Züge in einem Bisimulationsspiel in Betracht ziehen. Wir sagen, II gewinnt das n-Züge-Bisimulationsspiel, wenn sie eine Strategie hat, um n Züge lang zu spielen ohne zu verlieren. Analog dazu betrachten wir den Begriff der *n-Bisimilarität*, kurz $\sim_n$. Es seien $\mathcal{K}$ und $\mathcal{K}'$ zwei Kripkestrukturen mit Zuständen v bzw. v'.

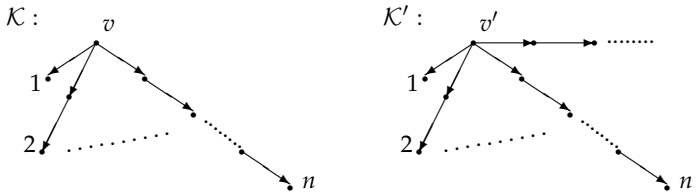
- $\mathcal{K}, v \sim_0 \mathcal{K}', v'$ gdw. für alle $i \in I$ gilt: $v \in P_i \iff v' \in P'_i$.
- $\mathcal{K}, v \sim_{n+1} \mathcal{K}', v'$ genau dann, wenn:
  - $\mathcal{K}, v \sim_n \mathcal{K}', v'$

– für alle w mit $v \xrightarrow{a} w$ existiert ein w' mit $v' \xrightarrow{a} w'$ und $\mathcal{K}, w \sim_n \mathcal{K}', w'$
– für alle w' mit $v' \xrightarrow{a} w'$ existiert ein w mit $v \xrightarrow{a} w$ und $\mathcal{K}, w \sim_n \mathcal{K}', w'$.

Es gilt für alle $n \in \mathbb{N}$, dass II genau dann das n-Züge-Bisimulationsspiel von (v, v') aus gewinnt, wenn $\mathcal{K}, v \sim_n \mathcal{K}', v'$ gilt.

Satz 3.12. Für alle Kripkestrukturen $\mathcal{K}, \mathcal{K}'$ mit Zuständen v bzw. v' gilt: Wenn $\mathcal{K}, v \sim \mathcal{K}, v$, dann ist $\mathcal{K}, v \sim_n \mathcal{K}', v'$ für alle n. Die Umkehrung gilt jedoch nicht: es gibt $\mathcal{K}, v$ und $\mathcal{K}, v'$, so dass $\mathcal{K}, v \sim_n \mathcal{K}', v'$ aber $\mathcal{K}, v \not\sim \mathcal{K}', v'$.

Beweis. Die erste Behauptung folgt unmittelbar aus den Definitionen. Für die zweite Behauptung betrachten wir folgende Kripkestrukturen:



$\mathcal{K}$ besitzt von v aus für jedes $n \in \mathbb{N}$ einen Pfad der Länge n. $\mathcal{K}'$ setzt sich aus $\mathcal{K}$ und einem unendlichen Pfad, der von v' ausgeht, zusammen. Es ist $\mathcal{K}, v \sim_n \mathcal{K}', v'$ für alle $n \in \mathbb{N}$, aber $\mathcal{K}, v \nsim \mathcal{K}', v'$. Spielt nämlich I entlang des unendlichen Pfades von $\mathcal{K}'$, dann muss II einen endlichen Pfad in $\mathcal{K}$ auswählen und auf diesem ziehen. Ist eine bestimmte Anzahl n von Zügen vor dem Spiel festgelegt worden, so kann II immer einen Pfad finden, der länger ist als n und somit n-Züge spielen ohne zu verlieren. Ist keine feste Zugzahl ausgemacht worden, so verliert II nach endlich vielen Zügen. Q.E.D.

Bisimulationsinvarianz von modallogischen Formeln. Die grundlegende Bedeutung von Bisimulationen kommt daher, dass modallogische Formeln bisimilare Zustände nicht unterscheiden können. Eine

verfeinerte Analyse zieht auch die Modaltiefe, d.h. die maximale Schachtelungstiefe von Modaloperatoren in einer Formel, in Betracht.

Definition 3.13. Die *Modaltiefe* einer Formel $\psi \in \mathrm{ML}$ ist induktiv definiert durch:

(1) $\mathrm{md}(\psi) = 0$ für aussagenlogische Formeln ψ,
(2) $\mathrm{md}(\neg\psi) = \mathrm{md}(\psi)$,
(3) $\mathrm{md}(\psi \circ \varphi) = \max(\mathrm{md}(\psi), \mathrm{md}(\varphi))$ für $\circ \in \{\wedge, \vee, \rightarrow\}$,
(4) $\mathrm{md}(\langle a \rangle \psi) = \mathrm{md}([a]\psi) = \mathrm{md}(\psi) + 1$.

Definition 3.14. Seien $\mathcal{K}$ und $\mathcal{K}'$ zwei Kripkestrukturen und $v \in \mathcal{K}$, $v' \in \mathcal{K}'$.

(1) $\mathcal{K}, v \equiv_{\mathrm{ML}} \mathcal{K}', v'$, wenn für alle $\psi \in \mathrm{ML}$ gilt:
$\mathcal{K}, v \models \psi \iff \mathcal{K}', v' \models \psi$.
(2) $\mathcal{K}, v \equiv^n_{\mathrm{ML}} \mathcal{K}', v'$, wenn für alle $\psi \in \mathrm{ML}$ mit $\mathrm{md}(\psi) \leq n$ gilt:
$\mathcal{K}, v \models \psi \iff \mathcal{K}', v' \models \psi$.

Satz 3.15. Für Kripkestrukturen $\mathcal{K}$, $\mathcal{K}'$ und $u \in \mathcal{K}$, $u' \in \mathcal{K}'$ gilt:

(1) Aus $\mathcal{K}, u \sim \mathcal{K}', u'$ folgt $\mathcal{K}, u \equiv_{\mathrm{ML}} \mathcal{K}', u'$;
(2) Aus $\mathcal{K}, u \sim_n \mathcal{K}', u'$ folgt $\mathcal{K}, u \equiv^n_{\mathrm{ML}} \mathcal{K}', u'$.

Beweis. Wir beweisen nur die erste Aussage, der Beweis der zweiten ist analog (per Induktion nach n).

Sei Z eine Bisimulation zwischen $\mathcal{K}$ und $\mathcal{K}'$. Wir behaupten, dass für alle $\psi \in \mathrm{ML}$ gilt:

$$\mathcal{K}, v \models \psi \iff \mathcal{K}', v' \models \psi \text{ für alle } (v, v') \in Z.$$

Wir beweisen dies per Induktion über den Formelaufbau von ψ. Für $\psi = P_i$ ist die Behauptung nach Definition einer Bisimulation erfüllt. Für die Fälle $\psi = \neg\varphi$, $\psi = (\varphi \vee \vartheta)$ und $\psi = (\varphi \wedge \vartheta)$ ist der Induktionsschritt offensichtlich: Wenn die Teilformeln von ψ auf $(\mathcal{K}, v)$ und $(\mathcal{K}, v')$ denselben Wahrheitswert haben, dann auch ψ selbst. Sei $\psi = \langle a \rangle \varphi$. Aus $\mathcal{K}, v \models \langle a \rangle \varphi$ folgt $\mathcal{K}, w \models \varphi$ für ein $w \in \mathcal{K}$ mit $v \xrightarrow{a} w$. Nach der Hin-Eigenschaft von Z existiert ein $w' \in \mathcal{K}'$ mit $v' \xrightarrow{a} w'$ und $(w, w') \in Z$. Nach Induktionsvoraussetzung gilt $\mathcal{K}', w' \models \varphi$, also $\mathcal{K}', v' \models \langle a \rangle \varphi$. Die Umkehrung folgt analog mit der Her-Eigenschaft.

$\psi = [a]\varphi$ brauchen wir wegen der Dualität $\langle a\rangle\varphi \equiv \neg[a]\neg\varphi$ nicht zu betrachten. Q.E.D.

Die Aussage (1) nennt man die *Bisimulationsinvarianz der Modallogik*:

Wenn $\mathcal{K}, v \models \psi$ und $\mathcal{K}, v \sim \mathcal{K}', v'$, dann auch $\mathcal{K}', v' \models \psi$.

Die Umkehrung von (1) gilt im Allgemeinen *nicht*. Um dies einzusehen, betrachten wir wieder die Kripkestrukturen $\mathcal{K}, \mathcal{K}'$ aus dem Beweis von Satz 3.12. Da $\mathcal{K}, v \sim_n \mathcal{K}', v'$ gilt $\mathcal{K}, v \equiv^n_{\mathrm{ML}} \mathcal{K}', v'$ für alle $n \in \mathbb{N}$ und daher $\mathcal{K}, v \equiv_{\mathrm{ML}} \mathcal{K}', v'$, obwohl $\mathcal{K}, v \not\sim \mathcal{K}', v'$. Es gibt jedoch wichtige Spezialfälle, in denen die Umkehrung doch gilt.

Definition 3.16. Ein Transitionssystem ist *endlich verzweigt*, wenn für alle Zustände v und alle Aktionen a die Menge $vE_a := \{w : (v, w) \in E_a\}$ der a-Nachfolger von v endlich ist. Insbesondere ist natürlich jedes endliche Transitionssystem endlich verzweigt.

Satz 3.17. Seien $\mathcal{K}, \mathcal{K}'$ endlich verzweigte Transitionssysteme. Dann gilt $\mathcal{K}, u' \sim \mathcal{K}', u'$ genau dann, wenn $\mathcal{K}, u \equiv_{\mathrm{ML}} \mathcal{K}', u'$ gilt.

Beweis. Sei $\mathcal{K}, u \equiv_{\mathrm{ML}} \mathcal{K}', u'$. Wir setzen $Z := \{(v, v') : \mathcal{K}, v \equiv_{\mathrm{ML}} \mathcal{K}', v'\}$. Dabei folgt sofort aus der Voraussetzung $\mathcal{K}, u \equiv_{\mathrm{ML}} \mathcal{K}', u'$, dass $(u, u') \in Z$. Wir zeigen, dass Z eine Bisimulation zwischen $\mathcal{K}$ und $\mathcal{K}'$ ist. Dann ist $\mathcal{K}, u \sim \mathcal{K}', u'$.

- Wenn $(v, v') \in Z$, dann gilt $v \in P_i \iff v' \in P'_i$, denn sonst wäre $\mathcal{K}, v \models P_i$ und $\mathcal{K}', v' \models \neg P_i$ (oder umgekehrt).
- *Hin:* Sei $(v, v') \in Z$, d.h. $\mathcal{K}, v \equiv_{\mathrm{ML}} \mathcal{K}', v'$, und $v \xrightarrow{a} w$. Wir setzen

  $$v'E_a := \{z' : v' \xrightarrow{a} z'\} \text{ und}$$
  $$X_w := \{z' \in v'E_a : \mathcal{K}, w \not\equiv_{\mathrm{ML}} \mathcal{K}', z'\}.$$

  Es reicht zu zeigen, dass ein $w' \in v'E_a \setminus X_w$ existiert, denn dann ist $(w, w') \in Z$ und die Hin-Eigenschaft erfüllt. Dazu wählen wir für jedes $z' \in X_w$ eine Formel $\varphi_{z'} \in \mathrm{ML}$, so dass $\mathcal{K}, w \models \varphi_{z'}$ aber $\mathcal{K}', z' \models \neg\varphi_{z'}$ und setzen $\varphi := \bigwedge\{\varphi_{z'} : z' \in X_w\}$. Da $\mathcal{K}'$ endlich verzweigt ist, gibt es nur endlich viele $z' \in X_w$, es ist also $\varphi \in \mathrm{ML}$. Es gilt $\mathcal{K}, w \models \varphi$, also $\mathcal{K}, v \models \langle a\rangle\varphi$. Da $\mathcal{K}, v \equiv_{\mathrm{ML}} \mathcal{K}', v'$, ist

auch $\mathcal{K}', v' \models \langle a \rangle \varphi$, d.h. es existiert ein $w' \in v' E_a$ mit $\mathcal{K}', w' \models \varphi$. Dann kann aber w' nicht Element von X_w sein, denn dann wäre $\mathcal{K}', w' \models \neg \varphi_{w'}$ und daher $\mathcal{K}', w' \models \neg \varphi$.

- Der Beweis der Her-Eigenschaft verläuft analog mit vertauschten Rollen von $\mathcal{K}, v$ und $\mathcal{K}', v'$. Q.E.D.

3.3 Abwicklungen und Baummodell-Eigenschaft

Eine Menge von Formeln (irgendeiner Logik, etwa der Modallogik oder der Prädikatenlogik), welche auf Transitionssystemen interpretiert wird, hat die *Baummodell-Eigenschaft* (BME), wenn jede erfüllbare Formel in Φ ein Modell hat, welches ein Baum ist.

Definition 3.18. Ein Transitionssystem $\mathcal{K} = (V, (E_a)_{a \in A}, (P_i)_{i \in I})$ mit einem ausgezeichneten Knoten w ist ein *Baum*, wenn

(1) $E_a \cap E_b = \emptyset$ für alle Aktionen $a \neq b$,
(2) für $E = \bigcup_{a \in A} E_a$ der Graph (V, E) ein (gerichteter) Baum mit Wurzel w im Sinn der Graphentheorie ist (siehe auch Kapitel 1.4, Lemma von König).

Wir werden zeigen, dass die Modallogik die Baumodell-Eigenschaft besitzt. Dazu betrachten wir *Abwicklungen* von Transitionssystemen. Die Abwicklung von $\mathcal{K}$ vom Zustand v aus besteht aus allen Pfaden in $\mathcal{K}$, die bei v beginnen. Dabei wird jeder Pfad als ein separates Objekt angesehen, d.h. selbst wenn sich zwei Pfade überschneiden, wird jeder zu einem neuen Zustand in der abgewickelten Struktur $\mathcal{T}$, und jeder Zustand aus $\mathcal{K}$, der auf einem Pfad von v aus erreicht wird, wird neu zu der Abwicklung hinzugefügt, unabhängig davon, ob er schon einmal erreicht wurde. Schleifen in $\mathcal{K}$ entsprechen also unendlichen Wegen in der Abwicklung. Formal werden Abwicklungen wie folgt definiert.

Definition 3.19. Sei $\mathcal{K} = (V^{\mathcal{K}}, (E_a^{\mathcal{K}})_{a \in A}, (P_i^{\mathcal{K}})_{i \in I})$ eine Kripkestruktur und $v \in V^{\mathcal{K}}$. Die *Abwicklung von $\mathcal{K}$ von v aus* ist die Kripkestruktur $\mathcal{T}_{\mathcal{K},v} = (V^{\mathcal{T}}, (E_a^{\mathcal{T}})_{a \in A}, (P_i^{\mathcal{T}})_{i \in I})$ mit

$$V^{\mathcal{T}} = \{\bar{v} = v_0 a_0 v_1 a_1 v_2 \dots v_{m-1} a_{m-1} v_m : m \in \mathbb{N}, \\ v_0 = v,\ v_i \in V^{\mathcal{K}},\ a_i \in A,\ (v_i, v_{i+1}) \in E_{a_i}^{\mathcal{K}} \text{ für alle } i < m\},$$

$$E_a^{\mathcal{T}} = \{(\bar{v}, \bar{w}) \in V^{\mathcal{T}} \times V^{\mathcal{T}} : \bar{w} = \bar{v}aw \text{ für ein } w \in V^{\mathcal{K}}\} \text{ und}$$
$$P_i^{\mathcal{T}} = \{\bar{v} = v_0 a_0 \dots v_m \in V^{\mathcal{T}} : v_m \in P_i^{\mathcal{K}}\}$$

Mit $\mathrm{End}(\bar{v})$ bezeichnen wir den letzten Knoten auf dem Pfad $\bar{v}$. Damit ist $\bar{v} \in P_i^{\mathcal{T}} \iff \mathrm{End}(\bar{v}) \in P_i^{\mathcal{K}}$.

Lemma 3.20. Es gilt $\mathcal{K}, v \sim \mathcal{T}_{K,v}, v$.

Beweis. $Z := \{(w, \bar{w}) \in V^{\mathcal{K}} \times V^{\mathcal{T}} : \mathrm{End}(\bar{w}) = w\}$ ist eine Bisimulation von $\mathcal{K}$ nach $\mathcal{T}_{\mathcal{K},v}$ mit $(v, v) \in Z$. Q.E.D.

Satz 3.21. ML hat die Baummodell-Eigenschaft.

Beweis. Sei ψ eine beliebige erfüllbare Formel aus ML. Es gibt also ein Modell $\mathcal{K}, v \models \psi$. Sei $\mathcal{T} := \mathcal{T}_{\mathcal{K},v}$ die Abwicklung von $\mathcal{K}, v$. Da $\mathcal{K}, v \sim \mathcal{T}, v$ gilt nach der Bisimulationsinvarianz der Modallogik auch $\mathcal{T}, v \models \psi$. Also hat ψ ein Baummodell. Q.E.D.

Dasselbe Argument zeigt, dass *jede* Klasse von bisimulationsinvarianten Formeln die Baummodell-Eigenschaft besitzt.

3.4 Temporale Logiken

ML ist keine besonders ausdrucksstarke Logik. Eine wesentliche Schwäche ist, dass der Wahrheitswert einer an einem Zustand v ausgewerteten Formel nur von einer beschränkten Umgebung von v abhängen kann. Zu den wichtigsten Aussagen in einer Reihe von Anwendungen (insbesondere in der Verifikation) gehören *Erreichbarkeitsaussagen* (ein „guter" Zustand wird auf jeden Fall irgendwann erreicht) oder *Sicherheitsbedingungen* (kein schlechter Zustand ist erreichbar). Erreichbarkeit ist aber nicht in ML formalisierbar, da jede ML-Formel ψ vom Zustand, an dem sie ausgewertet wird, höchstens $\mathrm{md}(\psi)$ viele Schritte weit in das Transitionssystem „hineinsehen" kann. Wir werden später sehen, dass Erreichbarkeitsaussagen in Transitionssystemen auch in FO nicht formalisierbar sind.

Es gibt verschiedene Möglichkeiten, solche Mängel von Logiken zu beseitigen, indem Rekursionsmechanismen hinzugefügt werden. Die

eleganteste Lösung sind sogenannte *Fixpunktlogiken*, welche so definiert sind, dass kleinste und größte Fixpunkte von definierbaren monotonen Operationen wieder definierbar sind. Fixpunktlogiken sind allerdings relativ kompliziert und sprengen den Rahmen dieser Vorlesung. Wir behandeln stattdessen die temporalen Logiken LTL („linear time temporal logic“) und CTL („computation tree logic“ oder auch „branching time temporal logic“), welche die Modallogik ML erweitern und in der (Hardware-)Verifikation sehr populär sind.

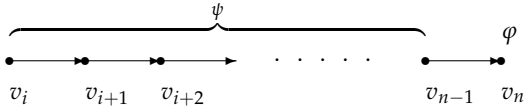
Syntax und Semantik von LTL

Die temporale Logik LTL wird auf endlichen oder unendlichen Wörtern oder Pfaden ausgewertet, also auf Folgen $v_0 v_1 \dots v_{n-1}$ bzw. $v_0 v_1 \dots$ mit atomaren Aussagen P_i. Die Idee von LTL ist, aussagenlogische Formeln über den atomaren Aussagen P_i durch temporale Operatoren (wie „next“, „until“, „eventually“ und „globally“) zu erweitern.

Definition 3.22 (Syntax von LTL)**.** Die Formeln von LTL sind induktiv wie folgt definiert:

- Alle aussagenlogischen Formeln über $\{P_i : i \in I\}$ gehören zu LTL.
- LTL ist abgeschlossen unter den Booleschen Operatoren $\wedge$, $\vee$, $\rightarrow$ und $\neg$.
- Wenn ψ, $\varphi \in$ LTL, dann sind auch die Ausdrücke $\mathrm{X}\,\psi$ und $(\psi\,\mathrm{U}\,\varphi)$ Formeln von LTL.

Die Intuition bei der Modellbeziehung ist folgende: Wie die Modallogik ML wird auch LTL an einzelnen Punkten ausgewertet. Ob eine Formel ψ an einem Punkt v_i gilt, kurz $\mathcal{W}, v_i \models \psi$, hängt von dem Teilwort $v_i v_{i+1} \dots$ ab, welches bei v_i beginnt. Der Ausdruck $\mathrm{X}\,\psi$ („next ψ“) bedeutet, dass am unmittelbar folgenden Element v_{i+1} die Formel ψ gilt, und der Ausdruck $\psi\,\mathrm{U}\,\varphi$ („ψ until φ“) besagt, dass an irgendeinem „späteren“ Element v_n φ gilt und davor immer ψ wahr ist:



Definition 3.23 (Semantik von LTL)**.** Sei $\mathcal{W}$ eine endliche oder unendliche Folge von Elementen $v_0 \dots v_{n-1}$ oder $v_0 v_1 \dots$ und atomaren Relationen P_i für $i \in I$. Die Bedeutung der Formeln P_i und der aussagenlogischen Junktoren ist auf die übliche Weise definiert. Außerdem gilt:

- $\mathcal{W}, v_i \models \mathrm{X}\,\psi$ genau dann, wenn v_i nicht das letzte Element von $\mathcal{W}$ ist und $\mathcal{W}, v_{i+1} \models \psi$;
- $\mathcal{W}, v_i \models (\psi\,\mathrm{U}\,\varphi)$, wenn ein $n \geq i$ existiert, so dass $\mathcal{W}, v_n \models \varphi$ und $\mathcal{W}, v_m \models \psi$ für alle m mit $i \leq m < n$.

Notation. Zwei wichtige Abkürzungen sind:

$\mathrm{F}\,\psi := (1\,\mathrm{U}\,\psi)$ (irgendwann wird ψ gelten)

$\mathrm{G}\,\psi := \neg\,\mathrm{F}\,\neg\psi$ (immer wird ψ gelten)

Beispiel 3.24.

- In LTL kann man ausdrücken, dass in einem unendlichen Wort $\mathcal{W}$ eine Formel φ an unendlichen vielen Positionen gilt. In der Tat besagt $\mathrm{G\,F}\,\varphi$, dass für jedes i ein $j \geq i$ existiert, so dass $\mathcal{W}, v_j \models \varphi$, und dies ist genau dann der Fall, wenn φ an unendlich vielen Positionen v_j gilt.
- Entsprechend gilt die Formel $\mathrm{F\,G}\,\neg\varphi$ ausgewertet über einem unendlichen Wort $\mathcal{W}$ genau dann, wenn φ nur an endlich vielen Positionen gilt.
- Die Formel $\mathrm{G}(\varphi \to (\varphi\,\mathrm{U}\,\psi))$ besagt, dass zu jeder Position an der φ gilt eine spätere Position existiert an der ψ gilt, und dass zwischen beiden Positionen immer φ gilt.

Wir haben gesehen, dass die Modallogik ML in die Prädikatenlogik FO eingebettet werden kann. Gilt dies auch für LTL? Dies hängt davon ab, wie wir Wörter bzw. Pfade als Strukturen formalisieren; anders ausgedrückt, ob wir FO-Formeln betrachten, welche die Ordnungsrelation auf den Elementen benutzen, oder ob nur die Nachfolgerrelation zur Verfügung steht. Stellen wir (endliche oder unendliche) Wörter als Strukturen der Form

$$\mathcal{W} = (V, <, (P_i)_{i \in I})$$

mit Universum $V = \omega$ (die Menge der natürlichen Zahlen) oder $V = \{0, \ldots, n-1\}$ sowie der üblichen linearen Ordnung $<$ auf V und mit einstelligen Relationen $P_i \subseteq V$ dar, so lässt sich LTL in FO einbetten.

Satz 3.25. Zu jeder LTL-Formel ψ existiert eine FO-Formel $\psi^*(x)$ der Signatur $\{<\} \cup \{P_i : i \in I\}$, so dass für alle $\mathcal{W}, v$ gilt:

$$\mathcal{W}, v \models \psi \quad \iff \quad \mathcal{W} \models \psi^*(v).$$

Beweis. Der Beweis ist analog zum Beweis der Einbettung von ML in FO, mit folgenden Änderungen: Formeln der Form $\psi = \mathrm{X}\,\varphi$ werden übersetzt in

$$\psi^*(x) := \exists y(x < y \wedge \neg\exists z(x < z \wedge z < y) \wedge \varphi^*(y)),$$

und Formeln der Form $\psi = (\varphi\,\mathrm{U}\,\vartheta)$ werden übersetzt in

$$\psi^*(x) := \exists y(x < y \wedge \vartheta^*(y) \wedge \forall z((x \leq z \wedge z < y) \rightarrow \varphi^*(z))).$$

Q.E.D.

Wenn aber auf dem Universum V statt der Ordnungsrelation $<$ nur die Nachfolgerrelation $E = \{(v_i, v_j) \in V \times V : j = i + 1\}$ zur Verfügung steht, dann kann man mit FO-Formeln nicht alle LTL-Eigenschaften ausdrücken. Wir werden mit den im nächsten Kapitel entwickelten Methoden beweisen können, dass bereits Formeln der Form $\mathrm{G\,F}\,P$ keine äquivalente FO-Formel ohne Ordnungsrelation zulassen.

Temporale Logiken auf Transitionssystemen. In vielen Anwendungen wird LTL (und andere temporale Logiken) zur Verifikation von Eigenschaften von Transitionssystemen verwendet. Wir betrachten dabei Transitionssysteme mit nur einer Transitionsrelation, d.h. Strukturen der Form $\mathcal{K} = (V, E, (P_i)_{i \in I})$ und setzen der Einfachheit halber voraus, dass E nicht terminiert, d.h. zu jedem $u \in V$ existiert ein v, so dass $(u, v) \in E$. Für eine LTL-Formel ψ sagen wir, dass ψ am Zustand v von $\mathcal{K}$ gilt, kurz $\mathcal{K}, v \models \psi$, wenn ψ auf *allen* unendlichen Pfaden durch $\mathcal{K}$, welche bei v beginnen, gilt.

Syntax und Semantik von CTL

Eine andere Möglichkeit, Aussagen über das mögliche Verhalten eines Transitionssystem zu machen, führt auf die „branching time logic“ CTL. Die Idee von CTL ist, ML um Pfadquantoren und temporale Operatoren auf Pfaden zu erweitern.

Definition 3.26 (Syntax von CTL)**.** Die Formeln von CTL sind induktiv definiert wie folgt.

- Alle aussagenlogischen Formeln über $\{P_i : i \in I\}$ gehören zu CTL.
- CTL ist abgeschlossen unter den Booleschen Operatoren $\wedge$, $\vee$, $\rightarrow$ und $\neg$.
- Wenn ψ, $\varphi \in$ CTL, dann sind auch die Ausdrücke $\mathrm{E}\,\mathrm{X}\,\psi$, $\mathrm{A}\,\mathrm{X}\,\psi$, $\mathrm{E}(\psi\,\mathrm{U}\,\varphi)$ und $\mathrm{A}(\psi\,\mathrm{U}\,\varphi)$ Formeln von CTL.

Die Intuition bei der Modellbeziehung ist folgende: Sei $\mathcal{K}$ ein Transitionssystem und v ein Zustand von $\mathcal{K}$. Dann quantifizieren E und A über unendliche Pfade $v = v_0v_1v_2\ldots$ in $\mathcal{K}$, welche bei v beginnen und auf denen die temporalen Operatoren dann ausgewertet werden.

Definition 3.27 (Semantik von CTL)**.** Sei $\mathcal{K} = (V, E, (P_i)_{i\in I})$ eine Kripkestruktur und $v \in V$. Dann gilt:

- $\mathrm{E}\,\mathrm{X}\,\psi :\equiv \Diamond\psi$;
- $\mathrm{A}\,\mathrm{X}\,\psi :\equiv \Box\psi$;
- Es gilt $\mathcal{K}, v \models \mathrm{E}(\psi\,\mathrm{U}\,\varphi)$, wenn ein Pfad $v_0v_1v_2\ldots$ mit $v = v_0$ und ein $n \geq 0$ existiert, so dass $\mathcal{K}, v_n \models \varphi$ und $\mathcal{K}, v_m \models \psi$ für alle m mit $0 \leq m < n$.
- Es gilt $\mathcal{K}, v \models \mathrm{A}(\psi\,\mathrm{U}\,\varphi)$, wenn für alle unendlichen Pfade $v_0v_1v_2\ldots$ mit $v_0 = v$ ein $n \geq 0$ existiert, so dass $\mathcal{K}, v_n \models \varphi$ und $\mathcal{K}, v_m \models \psi$ für alle m mit $0 \leq m < n$.

Analog zu LTL definieren wir die folgenden abkürzenden Schreibweisen:

$\mathrm{E}\,\mathrm{F}\,\psi :\equiv \mathrm{E}(1\,\mathrm{U}\,\psi)$ (ex. ein Pfad, auf dem irgendwann ψ gilt)

$\mathrm{A}\,\mathrm{F}\,\psi :\equiv \mathrm{A}(1\,\mathrm{U}\,\psi)$ (auf allen Pfaden gilt irgendwann ψ)

$\mathrm{E}\,\mathrm{G}\,\psi :\equiv \neg\,\mathrm{A}\,\mathrm{F}\,\neg\psi$ (ex. ein Pfad, auf dem immer ψ gilt)

$$\mathrm{A}\,\mathrm{G}\,\psi :\equiv \neg\,\mathrm{E}\,\mathrm{F}\,\neg\psi \qquad \text{(auf allen Pfaden gilt immer } \psi\text{)}$$

Beispiel 3.28.

- In CTL ist Erreichbarkeit definierbar: $\mathrm{E}\,\mathrm{F}\,\psi$ bedeutet, dass ein Zustand erreicht werden kann, an dem ψ gilt.
- $\mathrm{A}\,\mathrm{G}\,\neg(P \wedge Q)$ drückt aus, dass sich P und Q in allen erreichbaren Zuständen ausschließen.
- $\mathrm{A}\,\mathrm{G}\,\mathrm{A}\,\mathrm{F}\,\psi$ besagt, dass ψ unendlich oft auf allen Pfaden gilt.

Diese Beispiele zeigen, dass viele für die Verifikation wichtige Aussagen in CTL formalisierbar sind. Dies allein macht aber noch nicht die Bedeutung von CTL aus. Wichtig ist, dass CTL andererseits günstige modelltheoretische und algorithmische Eigenschaften besitzt. Zunächst ist CTL (wie ML) invariant unter Bisimulation.

Übung 3.2. Zeigen Sie, per Induktion über den Aufbau von CTL-Formeln, dass für alle $\psi \in$ CTL gilt: Wenn $\mathcal{K}, v \models \psi$ und $\mathcal{K}, v \sim \mathcal{K}', v'$, dann auch $\mathcal{K}', v' \models \psi$. Es folgt, dass CTL die Baummodell-Eigenschaft hat.

CTL-Formeln können effizient ausgewertet werden (in linearer Zeit sowohl bezüglich der Länge der Formel wie der Größe des Transitionssystems).

Satz 3.29. Es gibt einen Algorithmus, welcher zu einem gegebenen endlichen Transitionssystem $\mathcal{K}$ und einer Formel $\psi \in$ CTL in Zeit $O(\|\mathcal{K}\| \cdot |\psi|)$ die Extension $[\![\psi]\!]^{\mathcal{K}}$ berechnet.

Der Beweis beruht auf darauf, dass Formeln der Form $\mathrm{E}(\psi\,\mathrm{U}\,\varphi)$ und $\mathrm{A}(\psi\,\mathrm{U}\,\varphi)$ mit Hilfe von graphentheoretischen Algorithmen mit linearer Laufzeit ausgewertet werden können. Weitere wichtige Eigenschaften von CTL sind:

- CTL hat die Endliche-Modell-Eigenschaft.
- das Erfüllbarkeitsproblem für CTL ist entscheidbar (in exponentieller Zeit).

Dies kann hier nicht bewiesen werden. Für die Behandlung von CTL und anderen modalen und temporalen Logiken sind insbesondere automatentheoretische Methoden wichtig.

Im Gegensatz zu ML kann CTL *nicht* in FO eingebettet werden (da z.B. Erreichbarkeit nicht FO-definierbar ist).

3.5 Monadische Logik

Eine hinreichende Erweiterung von FO ist MSO, die *monadische Logik zweiter Stufe*, welche FO um Quantoren über einstellige Relationssymbole (d.h. Mengenvariablen) erweitert. Aus einer Formel ψ können neue Formeln der Form $\exists X\psi$ bzw. $\forall X\psi$ gebildet werden, mit der Bedeutung „es gibt eine Teilmenge X des Universums, so dass ψ“ bzw. „für alle Teilmengen X des Universums gilt ψ“. So drückt z.B. die Formel

$$\forall X((Xs \wedge \forall y \forall z(Xy \wedge Eyz \rightarrow Xz)) \rightarrow Xt)$$

aus, dass im Graphen (V, E) ein Pfad von s nach t existiert.

Übung 3.3. Zeigen Sie, dass jede CTL-Formel in eine äquivalente Formel in MSO übersetzt werden kann.