Kapitel L:V

V. Erweiterungen und Anwendungen zur Logik

- Produktionsregelsysteme
- Inferenz für Produktionsregelsysteme
- Produktionsregelsysteme mit Negation
- Regeln mit Konfidenzen

- Nicht-monotones Schließen

- Logik und abstrakte Algebren

- Verifikation
- Verifikation mit dem Hoare-Kalkül
- Hoare-Regeln und partielle Korrektheit
- Terminierung

Nicht-monotones Schließen

Problemlösen mittels Logik

1. Ausgangspunkt.
   Ein System der realen Welt und eine Frage dazu.

2. Modellbildung.
   Abstraktion des Systems ($\rightsquigarrow$ Modell) und der Frage.

3. Formalisierung.
   Beschreibung von Modell und Frage als Formel α bzw. β.

4. Schlussfolgern bzw. Inferenz.
   (a) Überprüfung ob $\alpha \models \beta$ gilt.
   (b) Bestimmung möglicher Folgerungen β' aus α.

Bemerkungen:

- Nach dem Schlussfolgerungsprozess weiß man $\alpha \wedge \beta$ bzw. $\alpha \cup \beta$.
- Das klassische Schlussfolgern basiert auf unserem Wissen über die Dinge.
- Beim klassischen Schlussfolgern nimmt unser Wissen immer weiter zu: Je mehr man weiß bzw. je „größer“ α ist, um so mehr kann man folgern. Der Schlussfolgerungsprozess ist monoton.

Nicht-monotones Schließen

Schlussfolgern in der klassischen Logik

(a) Gegeben: $\alpha = \{\gamma,\ \gamma \rightarrow \beta\}$

Schlussregel: Modus Ponens.

$$\gamma \wedge (\gamma \rightarrow \beta) \;\vdash\; \beta$$

Nicht-monotones Schließen

Schlussfolgern in der klassischen Logik

(a) Gegeben: $\alpha = \{\gamma,\ \gamma \rightarrow \beta\}$

Schlussregel: Modus Ponens.

$$\gamma \wedge (\gamma \rightarrow \beta) \ \vdash\ \beta$$

(b) Gegeben: $\alpha = \{\gamma \wedge \neg\delta,\ \gamma \wedge \neg\delta \rightarrow \beta\}$

Schlussregel: Modus Ponens.

$$\gamma \wedge \neg\delta \wedge (\gamma \wedge \neg\delta \rightarrow \beta) \ \vdash\ \beta$$

Nicht-monotones Schließen

Schlussfolgern in der klassischen Logik

(a) Gegeben: $\alpha = \{\gamma,\ \gamma \to \beta\}$

Schlussregel: Modus Ponens.

$$\gamma \wedge (\gamma \to \beta) \ \vdash \ \beta$$

(b) Gegeben: $\alpha = \{\gamma \wedge \neg\delta,\ \gamma \wedge \neg\delta \to \beta\}$

Schlussregel: Modus Ponens.

$$\gamma \wedge \neg\delta \wedge (\gamma \wedge \neg\delta \to \beta) \ \vdash \ \beta$$

(c) Gegeben: $\alpha = \{\gamma \wedge \delta,\ \gamma \wedge \delta \to \beta\}$

Schlussregel: Modus Ponens.

$$\gamma \wedge \delta \wedge (\gamma \wedge \delta \to \beta) \ \vdash \ \beta$$

Bemerkungen:

- Die Mengenschreibweise bei den Formeln steht für eine logische UND-Verknüpfung.
- Schlussfolgern über Nicht-Wissen erfordert besondere Ableitungsprinzipien.
- Die zwei wichtigsten Ansätze hierzu sind Negation-as-Failure und das logische Schließen mittels Defaults (Default-Logik).

Nicht-monotones Schließen

Schlussfolgern über Nicht-Wissen: Negation-as-Failure

Gegeben: $\alpha = \{\gamma,\ \gamma \land \neg\delta \to \beta\}$

Ableitungsprinzip: $\underset{naf}{\vdash} \ \equiv$ Modus Ponens + Negation-as-Failure

$$\gamma \land (\gamma \land \neg\delta \to \beta) \underset{naf}{\vdash} \beta$$

Bemerkungen:

- Lässt sich eine Formel δ nicht aus α ableiten*, so darf *unter Vorbehalt* ihr Gegenteil, $\neg\delta$, konjunktiv zu α hinzugenommen werden. Aus semantischer Sicht ist dies gleichbedeutend damit, dass $\neg\delta$ vorläufig als wahr angenommen wird.

  Idee: Closed-World-Assumption. „Alles, was gilt, ist ableitbar.“ bzw. „Was nicht ableitbar ist, das gilt auch nicht.“
- Durch Negation-as-Failure wird α (unter Vorbehalt, auf Basis aktuellen Wissens) so modifiziert, dass der Modus Ponens bzw. Resolution anwendbar wird (vgl. vorherige Folie zur klassischen Logik).
- Dieses Ableitungsprinzip findet in der Programmiersprache Prolog Verwendung, wobei statt des Modus Ponens das Resolutionsverfahren eingesetzt wird.

($*$) Bei Negation-as-Failure in Prolog wird versucht, δ mittels Backward-Chaining abzuleiten.

Nicht-monotones Schließen

Schlussfolgern über Nicht-Wissen: Default-Logik [Doyle/McDermott 1980]

Gegeben: $\alpha = \{\gamma,\ \gamma \wedge M\delta \rightarrow \beta\}$

Ableitungsprinzip: $\mid_{\overline{default}}$ $\equiv$ Modus Ponens + Default

$$\gamma \wedge (\gamma \wedge M\delta \rightarrow \beta) \mid_{\overline{default}} \beta$$

Schreibweise: $\dfrac{M\delta}{\beta}$

Bemerkungen:

- Lässt sich eine Formel $\neg\delta$ nicht aus α ableiten*, so darf *unter Vorbehalt* ihr Gegenteil, δ, konjunktiv zu α hinzugenommen werden. Aus semantischer Sicht ist dies gleichbedeutend damit, dass δ als wahr angenommen wird.

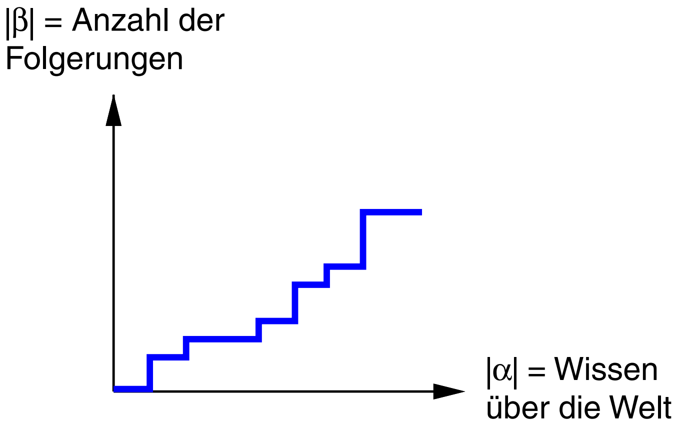
  Idee: „Solange kein Widerspruch bei der Annahme eines Sachverhalts auftritt, gehe von seiner Gültigkeit aus.“ $M\delta \equiv$ „it is consistent to assume δ.“
- Das „M“ (Modality) bei der Default-Logik kennzeichnet einen logischen Fakt, der defaultmäßig als vorhanden angesehen wird bzw. für den der Wert „wahr“ angenommen wird. Dadurch wird α (unter Vorbehalt, auf Basis aktuellen Wissens) so modifiziert, dass der Modus Ponens anwendbar wird (vgl. vorherige Folie zur klassischen Logik).

($*$) Der Ableitungsprozess in der Default-Logik ist üblicherweise datengetrieben, also Forward-Chaining.

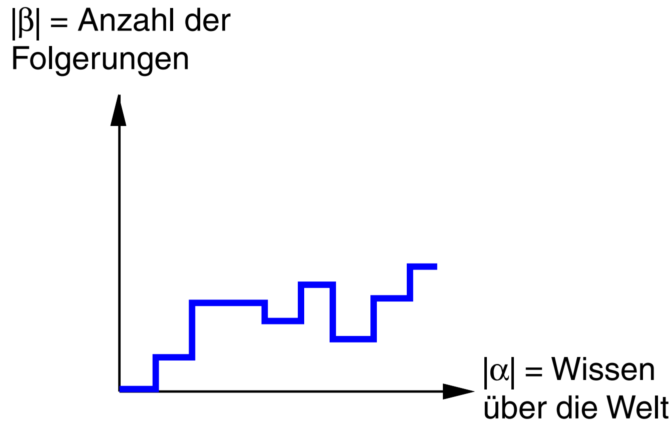
Nicht-monotones Schließen

Schlussfolgern über Nicht-Wissen

Eine Besonderheit bei Schlussfolgern über Nicht-Wissen ist, dass bei Zunahme des Wissens über die Welt das folgerbare Wissen abnehmen kann: α wird größer, die Menge β der Folgerungen aus α jedoch kleiner.



Klassische, monotone Situation



Nicht-monotone Situation

Bemerkungen:

- Wenn man über „Nicht-Wissen“ schlussfolgert nimmt die Menge der ableitbaren Fakten (= Wissen) nicht notwendiger Weise zu. Der Schlussfolgerungsprozess ist nicht-monoton. Üblich in diesem Zusammenhang ist deshalb der Begriff des *nicht-monotonen Schließens*.

Nicht-monotones Schließen

Schlussfolgern über Nicht-Wissen

Beispiel: $\alpha = \{C,\ (C \rightarrow B),\ ((B \wedge \neg D) \rightarrow E)\}$

Ableitungsprinzip: $\mid\!\frac{}{naf} \ \equiv$ Modus Ponens + Negation-as-Failure

Frage: $\beta = \{E\}$. Gilt $\alpha \mid\!\frac{}{naf} \beta$?

Nicht-monotones Schließen

Schlussfolgern über Nicht-Wissen

Beispiel: $\alpha = \{C,\ (C \rightarrow B),\ ((B \wedge \neg D) \rightarrow E)\}$

Ableitungsprinzip: $\underset{naf}{\models}\ \equiv$ Modus Ponens + Negation-as-Failure

Frage: $\beta = \{E\}$. Gilt $\alpha \underset{naf}{\models} \beta$?

Zeitpunkt	t_1	t_2	t_2'
Weltwissen	D ist unbekannt	D ist wahr	$\neg D$ ist wahr
Folgerungen	B, E	B	B, E

Wie operationalisiert man nicht-monotones Schließen?

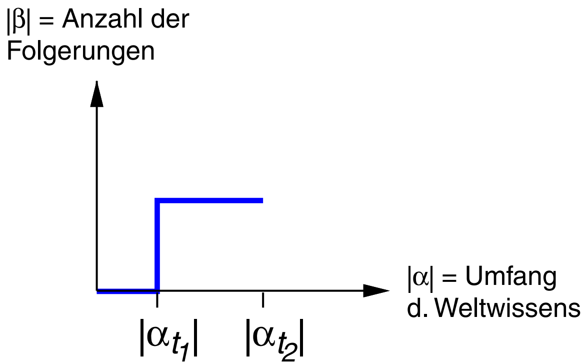
Bemerkungen:

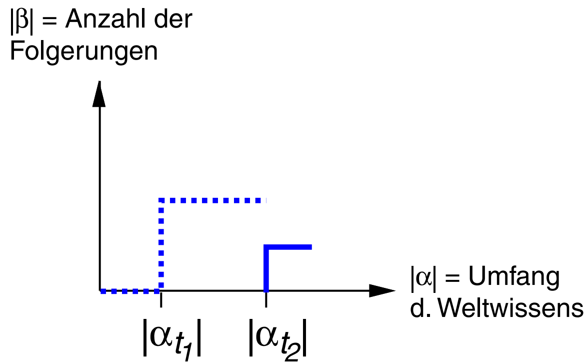
- Kommt man von der Situation zum Zeitpunkt t_1 in die Situation zum Zeitpunkt t_2, so darf unter der Annahme der Closed-World-Assumption E nicht in der Menge der abgeleiteten Fakten sein, denn E kann in dieser Situation nicht gefolgert werden.
- Die Mengenschreibweise bei den Formeln steht für eine logische UND-Verknüpfung.

Nicht-monotones Schließen

Operationalisierung

(a) Jedes Mal, wenn sich das Weltwissen ändert, werden alle Folgerungen (abgeleitete Fakten) gelöscht und der Inferenzprozess von vorne gestartet:

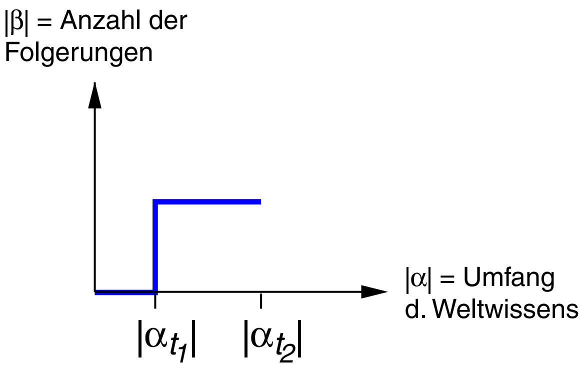


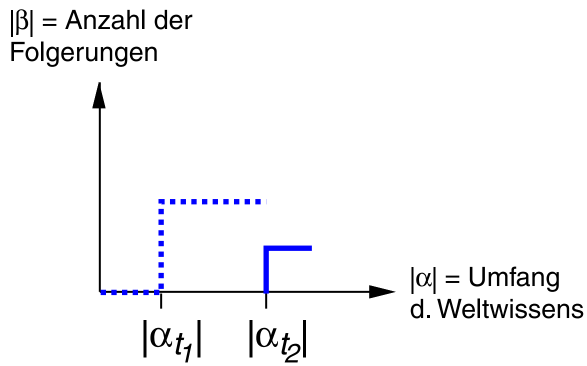


Nicht-monotones Schließen

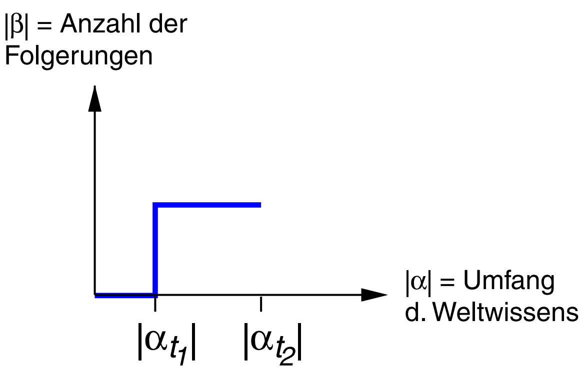
Operationalisierung

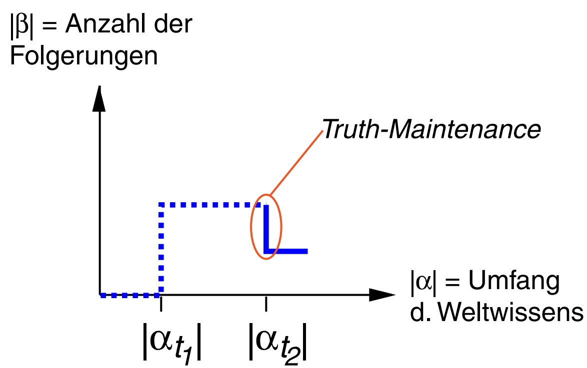
(a) Jedes Mal, wenn sich das Weltwissen ändert, werden alle Folgerungen (abgeleitete Fakten) gelöscht und der Inferenzprozess von vorne gestartet:





(b) Die Menge der Folgerungen (ableitbaren Fakten) wird inkrementell konsistent bzgl. der Schlussregel gehalten:





Bemerkungen:

- Systeme, die dabei helfen, einen nicht-montonen Schlussfolgerungsprozess nachzubilden, heißen *Truth-Maintenance-Systeme* (TMS) oder auch *Reason-Maintenance-Systeme* (RMS). Zwei wichtige Vertreter sind das Justification-based TMS und das Assumption-based TMS.

Nicht-monotones Schließen

Modellierung

Viele Modelle (z. B. in der Diagnose) lassen eine Aufteilung in folgende Mengen zu:

- R, universelle Formeln über die Welt (Regeln).
- P_t, aktuelles Faktenwissen über die Welt zum Zeitpunkt t (Prämissen).
- $\underbrace{F_t}_{\beta}$, Folgerungen aus $\underbrace{R \cup P_t}_{\alpha}$.

Es gelte $P_1 \subset P_2$. Die beiden Konzepte zur Operationalisierung des nicht-monotonen Schließens lassen sich wie folgt illustrieren:

$$
\begin{array}{lllll}
t_1: & R \cup P_1 & \vdash & R \cup P_1 \cup F_1 & \\
 & \quad\quad a & & \quad\quad\quad b & \cup\, P_2 \setminus P_1 \\
t_2: & R \cup P_2 & \vdash & R \cup P_2 \cup F_2 &
\end{array}
$$

Bemerkungen:

- Mit $P_1 \subset P_2$ gilt unmittelbar $F_1 \subseteq F_2$ für die monotone Situation. Für die nicht-monotone Situation lässt sich keine Aussage über die Relation zwischen F_1 und F_2 machen.

