Fortgeschrittene Themen der Wissensrepräsentation

Gabriele Kern-Isberner
LS 1 – Information Engineering

TU Dortmund
Sommersemester 2015

Kapitel 2 – Erweiterungen und Alternativen zu AGM

Kapitel 2

Erweiterungen und Alternativen zu AGM

Kapitel 2

2. Erweiterungen und Alternativen zu AGM

2.1 Update und andere Revisionsformen

Revision und Update 1/2

Die Revision im engeren Sinne behandelt lediglich den Prozess der Änderung von Wissen über eine bestimmte Situation, nicht jedoch die Modifikationen des Wissens, die durch sich ändernde Situationen notwendig werden (in diesem Fall spricht man von Update):

- Revision verarbeitet Wissensänderungen, die durch neue Informationen über schon vorher betrachtete Situationen verursacht werden (statische Weltsicht).
- Update verarbeitet Wissensänderungen, die sich durch neue Informationen über eine sich (möglicherweise) entwickelnde Welt ergeben (dynamische Weltsicht).

Revision – Beispiel

Big Boss hatte gewürfelt, wer von den drei Profikillern Peter, Paul und Mary einen Auftrag ausführen sollte: Bei einer geraden Augenzahl sollte es ein Mann sein, bei einer ungeraden eine Frau.

Das weiß auch Inspector Somerset, der den Mord aufzuklären hat. Er hält also jeden der drei für einen möglichen Täter.

Nun erfährt er, dass Peter am Mordtage wegen Trunkenheit am Steuer festgenommen wurde und sich zur Tatzeit auf einem Revier befand.

Inspektor Somerset revidiert nun sein Wissen und streicht Peter von seiner Verdächtigenliste, glaubt also nun, dass nur noch entweder Paul oder Mary als Täter in Frage kommen. (Man kann nun überlegen, welche Täterwahrscheinlichkeit man jedem der beiden zuweist ...)

Statisches Szenario des bereits verübten Mordes.

Revision und Update – Beispiel 1

Agent A glaubt, dass Agent B gestern den ganzen Tag zu Hause war und vermutet, dass er auch heute zu Hause ist.

Revision: Agent C erzählt Agent A, dass B gestern bei ihm zu Besuch war. Daraufhin revidiert A sein Wissen, er glaubt nun, dass B gestern nicht zu Hause war, sondern bei Agent C.

Update: Agent C erzählt Agent A, dass B heute morgen mit ihm gefrühstückt hat. Nun muss Agent A nicht notwendig sein bisheriges Wissen über den gestrigen Tag ändern, da sich die neue Information nicht direkt auf die gestrige Welt bezieht. Er könnte jedoch nun auch in Erwägung ziehen, dass B auch schon gestern bei Agent C war, sein Wissen über B's gestrigen Aufenthalt wird aber nicht notwendigerweise falsch.

Revision und Update 2/2

Unterschied zwischen Revision und Update auf Modellebene:

- Revision $K * A$: Die neuen Modelle sind Modelle von A, die möglichst nah an den Modellen von K liegen.
- Update $K \diamond A$: Alle Modelle von K sind mögliche (Ausgangs-)Weltzustände, die minimale Änderung aller dieser Welten muss bei einem Update berücksichtigt werden.

Revision und Update – Beispiel 2

$O_1, \ldots, O_5$ Register mit Inhalten o_i, $1 \leq i \leq 5$.

Datenübertragung über gestörten Kanal mit zwei möglichen Ergebnissen ω_1, ω_2 (bestimmt priori-Wissen $\phi \equiv \mathit{Core}(K)$):

$$\phi = \omega_1 \vee \omega_2, \quad \omega_1 = o_1 \overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5},\ \omega_2 = \overline{o_1}\,\overline{o_2} o_3 o_4 o_5$$

Zusätzliche Information: Alle Registerinhalte haben den gleichen Wert, d.h.

$$A = \nu_1 \vee \nu_2, \quad \nu_1 = o_1 o_2 o_3 o_4 o_5,\ \nu_2 = \overline{o_1}\,\overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5}$$

Revision und Update – Beispiel 2 (Forts.)

Wir verwenden als Distanzmaß die Metrik von Dalal:

$$\begin{array}{ll} d(\omega_1, \nu_1) = 4 & d(\omega_2, \nu_1) = 2 \\ d(\omega_1, \nu_2) = 1 & d(\omega_2, \nu_2) = 3, \end{array}$$

also $d(K, \nu_1) = 2, d(K, \nu_2) = 1.$

Revision $Mod(K * A) = \{\nu_2\}$ liefert in diesem Szenario ein intuitiv richtiges Ergebnis.

Revision und Update – Beispiel 2 (Forts.)

Im folgenden Kontext wäre ein Update sinnvoll:

Objekte $O_1, \ldots O_5$ befinden sich in einem Raum

o_i O_i *liegt auf dem Tisch*, $1 \leq i \leq 5$;

priori-Wissen:

$$\phi = o_1\overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5} \vee \overline{o_1}\,\overline{o_2}o_3o_4o_5$$

Anweisung an Roboter:

> *Sorge dafür, dass entweder alle Objekte auf dem Tisch liegen oder keines auf dem Tisch liegt!*

$\rightarrow$ neue Information: $A = o_1o_2o_3o_4o_5 \vee \overline{o_1}\,\overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5}$

Revision und Update – Beispiel 2 (Forts.)

$$Mod(\phi)\ : \omega_1 = o_1\overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5},\ \omega_2 = \overline{o_1}\,\overline{o_2}o_3o_4o_5$$

$$Mod(A) : \nu_1 = o_1o_2o_3o_4o_5\ \ ,\ \nu_2 = \overline{o_1}\,\overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5}$$

Wünschenswertes Ergebnis eines Update wäre hier:

$$Mod(\phi \diamond A) = Mod(A).$$

Allerdings – Revision liefert $K * A = Cn(\overline{o_1}\,\overline{o_2}\,\overline{o_3}\,\overline{o_4}\,\overline{o_5})$ (s.o.).

Revision und Update – Beispiel 2 (Forts.)

Wir verwenden den pma-Operator:

$$Mod(\phi \diamond_{pma} A) \quad = \quad \{\omega' \in Mod(A) \mid \exists \omega \in Mod(\phi) \text{ so, dass } \mathit{Diff}(\omega, \omega') \text{ minimal}^1 \text{ ist in } \mathit{Diff}(\omega, A)\}$$

$\mathit{Diff}(\omega_1, A)$	$\mathit{Diff}(\omega_2, A)$
$\mathit{Diff}(\omega_1, \nu_1) = \{O_2, O_3, O_4, O_5\}$	$\mathit{Diff}(\omega_2, \nu_1) = \{O_1, O_2\}$
$\mathit{Diff}(\omega_1, \nu_2) = \{O_1\}$	$\mathit{Diff}(\omega_2, \nu_2) = \{O_3, O_4, O_5\}$

Folglich $Mod(\phi \diamond_{pma} A) = \{\nu_1, \nu_2\} = Mod(A)$, wie gewünscht.

[1] bezgl. Mengeninklusion

Update – Ansatz

Es ist

$$Mod(\phi \diamond_{pma} A) = \bigcup_{\omega \in Mod(\phi)} \{\omega' \in Mod(A) \mid \textit{Diff}(\omega, \omega') \text{ minimal in } \textit{Diff}(\omega, A)\}$$

Dies führt auf den Ansatz:

$$Mod(\phi \diamond A) = \bigcup_{\omega \in Mod(\phi)} \min(Mod(A), \leq_\omega)$$

wobei $\leq_\omega$ eine geeignete, von ω abhängige Relation auf Ω ist.

Treue Zuordnungen

Jedem $\omega \in \Omega$ ordne man eine partielle Präordnung $\leq_\omega$ auf Ω zu; diese Zuordnung heißt treu, wenn gilt:

$$\forall\, \omega' \in \Omega,\ \text{wenn } \omega \neq \omega',\ \text{dann } \omega <_\omega \omega'$$

Beispiel pma: Definiere

$$\omega_1 \leq_\omega \omega_2 \text{ gdw. } \mathit{Diff}(\omega, \omega_1) \subseteq \mathit{Diff}(\omega, \omega_2)$$

Es ist $\mathit{Diff}(\omega, \omega') = \emptyset$ genau dann, wenn $\omega = \omega'$, also $\omega <_\omega \omega'$ für alle $\omega' \neq \omega$. Damit ist $\omega \mapsto \leq_\omega$ eine treue Zuordnung. ♣

Charakterisierungstheorem für Update

Theorem 1

Ein Update-Operator $\diamond$ erfüllt genau dann die (folgenden) Postulate (U1) - (U8), wenn es eine treue Zuordnung

$$\omega \mapsto \leq_\omega$$

gibt mit

$$Mod(\phi \diamond A) = \bigcup_{\omega \in Mod(\phi)} \min(Mod(A), \leq_\omega)$$

Postulate für Update

(U1) $\phi \diamond A \models A$.

(U2) Wenn $\phi \models A$ dann $\phi \diamond A \equiv \phi$.

(U3) Wenn ϕ, A erfüllbar, dann auch $\phi \diamond A$ erfüllbar.

(U4) Wenn $\phi_1 \equiv \phi_2$ und $A_1 \equiv A_2$, dann auch $\phi \diamond A_1 \equiv \phi \diamond A_2$.

(U5) $(\phi \diamond A) \wedge B \models \phi \diamond (A \wedge B)$.

(U6) Wenn $\phi \diamond A_1 \models A_2$ und $\phi \diamond A_2 \models A_1$, dann $\phi \diamond A_1 \equiv \phi \diamond A_2$.

(U7) Wenn ϕ vollständig[2] ist, dann
$(\phi \diamond A_1) \wedge (\phi \diamond A_2) \models \phi \diamond (A_1 \vee A_2)$.

(U8) Disjunktionsregel: $(\phi_1 \vee \phi_2) \diamond A \equiv (\phi_1 \diamond A) \vee (\phi_2 \diamond A)$.

[2] D.h., für jede Formel $A \in \mathcal{L}$ gilt entweder $\phi \models A$ oder $\phi \models \neg A$

Folgerungen 1/2

Aus Theorem 1 folgt:

Proposition 1

Erfüllt ein Update-Operator $\diamond$ die Postulate (U1) - (U8), so gilt:

$$\phi \diamond A = \bigvee_{\omega \in Mod(\phi)} \omega \diamond A.$$

Folgerungen 2/2

Proposition

Ist $\diamond$ ein Update-Operator, der (U2), (U4) und (U8) erfüllt, so gilt:

$$\phi \wedge A \models \phi \diamond A$$

Proposition

Ist $\diamond$ ein Update-Operator, der (U4) und (U8) erfüllt, so gilt die folgende Monotonie-Eigenschaft:

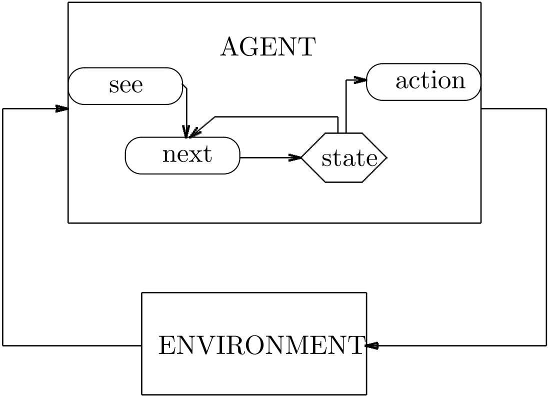
Gilt $\phi \models \psi$, dann $\phi \diamond A \models \psi \diamond A$.

Anwendungsgebiete für Update

Typische Einsatzgebiete für Update sind:

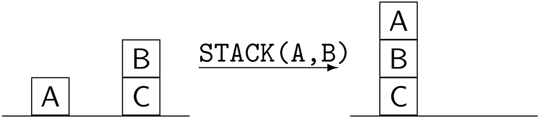
- Änderungen in (relationalen) Datenbanken
- Änderungen in Agentensystemen (Aktionslogik)

Anwendung: Agenten und Update 1/3



Ein einfacher Update-Algorithmus war der `STRIPS`-Algorithmus (s. DVEW) . . .

Anwendung: Agenten und Update 2/3



Vorbedingung:

`ONTABLE(A)`
`CLEAR(A)`
`CLEAR(B)`

Datenbasis vorher:			Datenbasis nachher:
`ONTABLE(A)` `CLEAR(B)`	*D*-dist	*A*-list	`ON(A,B)`

`CLEAR(A)`
`ON(B,C)`
`ONTABLE(C)`
	$\longrightarrow$ unverändert		`CLEAR(A)`
			`ON(B,C)`
			`ONTABLE(C)`

Anwendung: Agenten und Update 2/2

Sei a eine Aktion mit Ausführungsaxiom α und Effektaxiom β; der Agent befinde sich im inneren Zustand ϕ; $\alpha, \beta, \phi \in \mathcal{L}$.

Sei $\diamond$ ein Update-Operator für $\mathcal{L}$.

Ein neuer interner Zustand nach Ausführen der Aktion a im Zustand ϕ könnte dann bestimmt werden wie folgt:

$$\phi_{neu} = \begin{cases} \phi \diamond \beta, & \text{wenn } \phi \models \alpha \\ \phi \quad, & \text{sonst} \end{cases}$$

Revision vs. Update – Postulate

Postulate Revision

(RF 1) $\phi \circ A \models A$.

(RF 2) Wenn $\phi \wedge A$ erfüllbar ist, dann ist $\phi \circ A \equiv \phi \wedge A$.

(RF 3) Wenn A erfüllbar ist, dann ist auch $\phi \circ A$ erfüllbar.

(RF 4) $\phi_1 \equiv \phi_2,\ A_1 \equiv A_2 \Rightarrow \phi_1 \circ A_1 \equiv \phi_2 \circ A_2$.

(RF 5) $(\phi \circ A) \wedge B$ impliziert $\phi \circ (A \wedge B)$.

(RF 6) $(\phi \circ A) \wedge B$ erfüllbar: $\phi \circ (A \wedge B) \models (\phi \circ A) \wedge B$.

Postulate Update

(U1) $\phi \diamond A \models A$.

(U2) Wenn $\phi \models A$ dann $\phi \diamond A \equiv \phi$.

(U3) Wenn ϕ, A erfüllbar, dann auch $\phi \diamond A$ erfüllbar.

(U4) $\phi_1 \equiv \phi_2,\ A_1 \equiv A_2 \Rightarrow \phi \diamond A_1 \equiv \phi \diamond A_2$.

(U5) $(\phi \diamond A) \wedge B \models \phi \diamond (A \wedge B)$.

(U6) $\phi \diamond A_1 \models A_2,\ \phi \diamond A_2 \models A_1 \Rightarrow \phi \diamond A_1 \equiv \phi \diamond A_2$.

(U7) ϕ vollständig $\Rightarrow (\phi \diamond A_1) \wedge (\phi \diamond A_2) \models \phi \diamond (A_1 \vee A_2)$.

(U8) Disjunktionsregel: $(\phi_1 \vee \phi_2) \diamond A \equiv (\phi_1 \diamond A) \vee (\phi_2 \diamond A)$.

Revision vs. Update – Modelle

Revision:

$$Mod(K * A) = \min(Mod(A), \leq_K),$$

$\leq_K$ totale K-persistente Präordnung auf Ω.

Update:

$$Mod(\phi \diamond A) = \bigcup_{\omega \in Mod(\phi)} \min(Mod(A), \leq_\omega),$$

$\omega \mapsto \leq_\omega$ treue Zuordnung, $\leq_\omega$ partielle Präordnung auf Ω.

Andere Revisionsformen

- Ausradierung (Erasure) ist die Umkehroperation zu Update (ähnlich wie Kontraktion zu Revision);
- Focusing behandelt die Anwendung von Allgemeinwissen auf spezielle Situationen, unterscheidet also zwischen generischem Wissen und situativem bzw. evidentiellem Wissen.

  Beispiel: Im Schwanensee-Beispiel sind die Fakten A (Der Vogel ist ein Schwan) und B (Der See liegt in Schweden) evidentielles Wissen, während die Implikationen $B \Rightarrow C$ (Schweden ist ein Teil von Europa) und $A \wedge C \Rightarrow D$ (Europäische Schwäne sind weiß) generisches Wissen darstellen. ♣

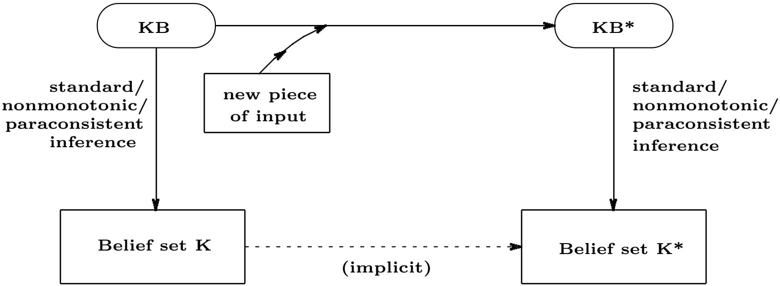
  In einer aussagen- oder prädikatenlogischen Umgebung ist diese Unterscheidung kaum möglich.

Kapitel 2

2. Erweiterungen und Alternativen zu AGM

2.2 Basisrevision

Basisrevision II



Basisrevision

Wissensbasis vs. Wissensmenge

- explizit repräsentiertes vs. implizit abgeleitetes Wissen;
- Wissen der Wissensbasis wird als (ausdrucks)stärker angesehen als implizites Wissen;
- Syntax ist wichtig! – D.h. semantisch äquivalente Wissensbasen können unter Revision bzw. Kontraktion unterschiedlich behandelt werden.

Basisrevision – A, B-Beispiel

Die folgenden Wissensbasen sind semantisch äquivalent:

$$H_1 = \{A, B\} \qquad H_2 = \{A \wedge B\} \qquad H_3 = \{B, B \Rightarrow A\}$$

Mögliche sinnvolle Kontraktionen:

$$H_1 - A = \{B\} \qquad H_2 - A = \emptyset \qquad \begin{aligned} H_3 - A &= \{B\} \\ H_3 - A &= \{B \Rightarrow A\} \end{aligned}$$

Änderungen von Wissensbasen

Auch im Fall der Wissensbasen ist es sinnvoll, zwischen Expansion, Revision und Kontraktion zu unterscheiden (mit den aus AGM bekannten Beschreibungen). Allerdings muss nun beachtet werden, dass deduktiver Abschluss (im Ergebnis) gerade nicht erwünscht ist.

Das vereinfacht zunächst die Expansion noch einmal:

Definition 2 (Basis-Expansion)

Sei $H \subseteq \mathcal{L}$ eine Wissensbasis, sei $A \in \mathcal{L}$. Die Basisexpansion von H um A ist definiert durch

$$H +_B A = H \cup \{A\}.$$

Levi-Identität für Basisrevision

Auch die Levi-Identität ist sinnvoll für Basisrevisionen:

Ist $\div$ ein Kontraktionsoperator für Wissensbasen, so wird ein Revisionsoperator $*_B$ für Wissensbasen H durch die folgenden Gleichheit gewonnen:

$$H *_B A = (H \div \neg A) +_B A.$$

Zentral ist daher die Suche nach Kontraktionsoperatoren $\div$ für Wissensbasen.

Obwohl als Ergebnis wieder eine Wissensbasis zurückgegeben wird, erwartet man, dass logische Konsequenzen berücksichtigt werden.

Basisrevision – Schwanensee

A Schwan, B Schweden, C Europa, D weiß

Basis: $H = \{A, B, B \Rightarrow C, A \wedge C \Rightarrow D\}$

Mögliche Kontraktionen $H \div D$:

$$\begin{aligned} H_1^- &= \{B, B \Rightarrow C, A \wedge C \Rightarrow D\} \\ H_2^- &= \{A, B \Rightarrow C, A \wedge C \Rightarrow D\} \\ H_3^- &= \{A, B, A \wedge C \Rightarrow D\} \\ H_4^- &= \{A, B, B \Rightarrow C\} \end{aligned}$$

Basiskontraktion – Methodik und Qualität?

Es stellt sich wieder die Frage, welche Basiskontraktionen gut (im Sinne von "logisch sinnvoll") sind.

Einiges können wir aus dem AGM-Rahmenwerk übernehmen, allerdings müssen wir nun explizit darauf achten, dass wir logische Konsequenzen nicht übersehen. Andererseits ist die endliche Repräsentation der Basis von Vorteil, wenn wir Teilmengen der Wissensbasis betrachten wollen.

Eine Methodik, die wir sofort auch für diesen Fall von Wissensrevision übernehmen können, ist die partial meet-Kontraktion.

Maximale Kontraktionsmengen

Definition 3 (Maximale Kontraktionsmengen)

Eine Menge M heißt maximale Kontraktionsmenge von H bzgl. A, wenn gilt:

- $M \subseteq H$;
- $A \notin Cn(M)$;
- für jedes M' mit $M \subseteq M' \subseteq H$, $M \neq M'$, ist $A \in Cn(M')$.

Die Menge aller maximalen Kontraktionsmengen von H bzgl. A wird wieder mit $H \perp A$ bezeichnet.

Beachten Sie: Maximale Kontraktionsmengen sind jetzt in der Regel nicht mehr deduktiv abgeschlossen (wie im AGM-Fall).

Partial meet für Basiskontraktion 1/2

Eine Selektionsfunktion S für H wird realisiert durch Auswahl einer Teilmenge der "besten" Mengen aus $H \perp A$ (für alle $A \in \mathcal{L}$):

$$S(H \perp A) \subseteq H \perp A$$

mit

- $S(H \perp A) \neq \emptyset$, wenn $H \perp A \neq \emptyset$, und
- $S(H \perp A) = H$, wenn $H \perp A = \emptyset$ (gdw. A Tautologie).

Partial meet für Basiskontraktion 2/2

Definition 4 (Partial meet-Basiskontraktion)

Sei $H \subseteq \mathcal{L}$ und S eine Selektionsfunktion für H. Die durch S erzeugte partial meet Basiskontraktion $\div_S$ auf H ist gegeben durch

$$H \div_S A = \bigcap S(H \perp A)$$

für alle $A \in \mathcal{L}$.

Partial meet Basisrevision - Schwanensee

A Schwan, B Schweden, C Europa, D weiß

Basis: $H = \{A, B, B \Rightarrow C, A \wedge C \Rightarrow D\}$

Maximale Kontraktionsmengen $H \perp D = \{H_1^-, \ldots, H_4^-\}$ mit

$$\begin{aligned} H_1^- &= \{B, B \Rightarrow C, A \wedge C \Rightarrow D\} \\ H_2^- &= \{A, B \Rightarrow C, A \wedge C \Rightarrow D\} \\ H_3^- &= \{A, B, A \wedge C \Rightarrow D\} \\ H_4^- &= \{A, B, B \Rightarrow C\} \end{aligned}$$

$S(H \perp D) = \{H_1^-, H_4^-\} \Rightarrow$

$$H \div_S D = H_1^- \cap H_4^- = \{B, B \Rightarrow C\}.$$

Hansson-Postulate für Basiskontraktion 1/2

S.O. Hansson schlägt die folgenden Postulate als grundlegend für Basiskontraktionen vor [Hansson 1999][3]:

Sei $H \subseteq \mathcal{L}$ (endlich), sei $A \in \mathcal{L}$.

- **(BC 1)** Inclusion: $H \div A \subseteq H$.
- **(BC 2)** Success: $A \notin Cn(H \div A)$, wenn A keine Tautologie ist.
- **(BC 3)** Failure: Ist A eine Tautologie, so ist $H \div A = H$.
- **(BC 4)** Vacuity: Ist $A \notin Cn(H)$, so ist $H \div A = H$.

[3] S.O. Hansson, A Textbook of Belief Dynamics. Kluwer Academic Publishers, 1999

Hansson-Postulate für Basiskontraktion 2/2

(BC 5) Relevance: Wenn $B \in H$, aber $B \notin H \div A$, dann gibt es H' mit $H \div A \subseteq H' \subseteq H$ und $A \notin Cn(H')$, aber $A \in Cn(H' \cup \{B\})$.

(BC 6) Core Retainment: Wenn $B \in H$, aber $B \notin H \div A$, dann gibt es H' mit $H' \subseteq H$ und $A \notin Cn(H')$, aber $A \in Cn(H' \cup \{B\})$.

(BC 7) Closure: Wenn H deduktiv abgeschlossen ist, so ist $H \div A$ deduktiv abgeschlossen.

(BC 8) Relative Closure: $H \cap Cn(H \div A) \subseteq H \div A$.

(BC 9) Extensionality: Sind A, B semantisch äquivalent, d.h., $A \equiv B$, so ist $H \div A = H \div B$.

(BC 10) Uniformity: Wenn für alle $H' \subseteq H$ gilt:

$$A \in Cn(H') \quad \text{gdw.} \quad B \in Cn(H'),$$

dann ist $H \div A = H \div B$.

BC-Postulate – Zusammenhänge

Diese Postulate sind nicht unabhängig voneinander, sondern lassen sich auf wenige Grundpostulate reduzieren:

Proposition 2

Sei $\div$ eine Basiskontraktion auf H. Dann gelten die folgenden Implikationen:

- *Relevance impliziert Relative Closure und Core Retainment.*
- *Inclusion und Core Retainment implizieren Failure und Vacuity.*
- *Inclusion und Relative Closure implizieren Closure.*
- *Uniformity impliziert Extensionality.*

Gemeinsam mit Success bleiben damit die folgenden 4 Grundpostulate:

Success, Inclusion, Relevance und Uniformity.

BC-Postulate für partial meet-Basiskontraktionen

Theorem 5

Sei $H \subseteq \mathcal{L}$ eine Wissensbasis. Ein Kontraktionsoperator $\div$ ist genau dann eine partial meet-Basiskontraktion, wenn er Success, Inclusion, Relevance und Uniformity erfüllt.

Basisrevision – A, B-Beispiel (Forts.)

Die folgenden Wissensbasen sind semantisch äquivalent:

$$H_1 = \{A, B\} \qquad H_2 = \{A \wedge B\} \qquad H_3 = \{B, B \Rightarrow A\}$$

Mögliche partial meet Kontraktionen:

$$H_1 \dotdiv A \;=\; \{B\} \qquad H_2 \dotdiv A \;=\; \emptyset \qquad \begin{array}{rcl} H_3 \dotdiv A & = & \{B\} \\ H_3 \dotdiv A & = & \{B \Rightarrow A\} \end{array}$$

$H_1 \dotdiv A = \emptyset$ ist nicht kompatibel mit Relevance;

aber $H_3 \dotdiv A = \emptyset$ ist kompatibel mit Relevance.

Verbindungen zu AGM

Grundsätzlich sind alle Methodiken und Postulate für Basiskontraktionen auch für Wissensmengen anwendbar. Für deduktiv abgeschlossene Mengen gelten aber stärkere Implikationen, d.h., gewisse feine Unterschiede, die für den Fall der Basisrevision wichtig sind, sind im AGM-Rahmen unsichtbar.

Umgekehrt vermisst man das wichtige Recovery-Postulat aus AGM für den Fall der Basiskontraktion:

Recovery $H \subseteq Cn((H \div A) \cup \{A\})$.

Verbindungen zu AGM (Forts.)

Die folgende Proposition zeigt z.B., warum Uniformity für AGM keine Rolle spielt, und wie das Recovery-Postulat aus AGM bei der Basiskontraktion gewonnen werden kann:

Proposition 3

Sei $\div$ ein Kontraktionsoperator für eine deduktiv abgeschlossene Menge K.

- *Wenn $\div$ Extensionality und Vacuity erfüllt, so erfüllt er auch Uniformity.*
- *Wenn $\div$ Core Retainment erfüllt, so erfüllt er auch Recovery.*
- *Wenn $\div$ Relevance erfüllt, so erfüllt er auch Recovery.*
- *Wenn $\div$ Closure, Inclusion, Vacuity, und Recovery erfüllt, so erfüllt er auch Relevance.*

Verbindungen zu AGM (Forts.)

In der Sprache der Basiskontraktion erhalten wir nun die bekannte Charakterisierung der partial meet-Kontraktionen für Wissensmengen:

Theorem 6

Ist H eine Wissensmenge (also deduktiv abgeschlossen), so ist der Kontraktionsoperator $\dot{-}$ genau dann eine partial meet-Kontraktion für H, wenn er die Postulate Closure, Inclusion, Vacuity, Success, Extensionality und Recovery erfüllt.

Kernel-Kontraktion 1/3

Im Bereich der Basiskontraktion ist noch eine andere Methodik zur Kontraktion sehr wichtig und auch naheliegend, die für AGM zu keinen neuen Ergebnissen führt: Die sog. Kernel-Kontraktion.

Im Unterschied zur partial meet-Kontraktion wählt man hier keine Mengen aus, sondern Formeln:

Definition 7 (Kernel-Menge, A-Kernels)

Sei $H \subseteq \mathcal{L}$ eine Wissensbasis und $A \in \mathcal{L}$. Die Kernel-Menge von H bzgl. A, bezeichnet mit $H \perp\!\!\!\perp A$, ist die Menge aller Mengen X so, dass gilt:

- $X \subseteq H$,
- $X \models A$, und
- wenn $Y \subset X$, $Y \neq X$, dann gilt $Y \not\models A$.

Die Elemente von $H \perp\!\!\!\perp A$ heißen A-Kernels.

Kernel-Kontraktion 2/3

$H \perp\!\!\!\perp A$ enthält also die minimalen Teilmengen von H, die A implizieren. D.h., wenn man auch nur eine Formel aus einer Menge aus $H \perp\!\!\!\perp A$ entfernt, so impliziert die verbleibenden Menge A nicht mehr, kann also als Basis für eine Kontraktion benutzt werden.

Definition 8 (Incision-Funktion)

Eine Incision-Funktion für H ist eine Funktion σ auf den Mengen $H \perp\!\!\!\perp A$ $(A \in \mathcal{L})$ mit den folgenden Eigenschaften:

- $\sigma(H \perp\!\!\!\perp A) \subseteq \bigcup(H \perp\!\!\!\perp A)$;
- für $X \in H \perp\!\!\!\perp A, X \neq \emptyset$ ist $X \cap \sigma(H \perp\!\!\!\perp A) \neq \emptyset$.

Incision-Funktionen wählen also aus jedem nichtleeren A-Kernel mindestens eine Formel aus.

Kernel-Kontraktion 3/3

Definition 9 (Kernel-Kontraktion)

Sei σ eine Incision-Funktion für H. Die durch σ definierte Kernel-Kontraktion $\div_\sigma$ für H ist definiert durch

$$H \div_\sigma A = H \backslash \sigma(H \perp\!\!\!\perp A).$$

Ein Basiskontraktionsoperator $\div$ für H heißt Kernel-Kontraktion wenn es eine Incision-Funktion σ für H gibt so, dass $H \div A = H \div_\sigma A$ für alle $A \in \mathcal{L}$ gilt.

Kernel-Kontraktion – Sonderfälle

Wir betrachten die üblichen Sonderfälle:

- $A \equiv \top$: In diesem Fall ist $H \perp\!\!\!\perp A = \{\emptyset\}$ und damit $\sigma(H \perp\!\!\!\perp A) = \emptyset$, und folglich ist $H \div_\sigma A = H$. (Failure)
- $A \notin Cn(H)$: In diesem Fall ist $H \perp\!\!\!\perp A = \emptyset$ und damit ebenfalls $\sigma(H \perp\!\!\!\perp A) = \emptyset$, und folglich ist $H \div_\sigma A = H$. (Vacuity)

Kernel-Kontraktion – Charakterisierung

Theorem 10 (Charakterisierung Kernel-Kontraktion)

Ein Basiskontraktionsoperator $\div$ für H ist genau dann eine Kernel-Kontraktion, wenn er Success, Inclusion, Core-Retainment und Uniformity erfüllt.

Proposition 4

Jede partial meet Kontraktion ist auch eine Kernelkontraktion.

Hier gilt nicht die Umkehrung, d.h., Kernelkontraktionen sind echt allgemeiner als partial meet Kontraktionen.

Kernel-Kontraktion – Probleme

Allerdings sind Kernel-Kontraktionen für den allgemeinen Gebrauch zu einfach und können zu unintuitiven Resultaten führen:

Beispiel: Sei $H = \{A, A \vee B, A \Leftrightarrow B\}$. Dann ist

$$H \perp\!\!\!\perp AB = \{\{A, A \Leftrightarrow B\}, \{A \vee B, A \Leftrightarrow B\}\}.$$

Sei σ definiert durch

$$\sigma(H \perp\!\!\!\perp AB) = \{A \vee B, A \Leftrightarrow B\}.$$

Dann ist $H \div_\sigma AB = \{A\}$. Es erscheint jedoch seltsam, dass auch $A \vee B$ aufgegeben wird. ♣

Glatte Kernel-Kontraktionen

Bei Incision sollten auch Konsequenzen beachtet werden:

Definition 11 (Glatte Kernel-Kontraktionen)

Eine Incision-Funktion σ für H heißt glatt, wenn für alle Teilmengen $H' \subseteq H$ das Folgende gilt:

Gilt $H' \models B$ und $B \in \sigma(H \perp\!\!\!\perp A)$, dann ist $H' \cap \sigma(H \perp\!\!\!\perp A) \neq \emptyset$.

Eine Kernel-Kontraktion heißt glatt, wenn sie durch eine glatte Incision-Funktion definiert werden kann.

Im obigen Beispiel gilt für $H' = \{A\}$:
$H' \models A \vee B$ und $A \vee B \in \sigma(H \perp\!\!\!\perp AB)$, aber $H' \cap \sigma(H \perp\!\!\!\perp AB) = \emptyset$.
Die Kernel-Kontraktion ist also nicht glatt.

Glatte Kernel-Kontraktionen – Charakterisierung

Theorem 12 (Charakterisierung glatte Kernel-Kontraktion)

Ein Basiskontraktionsoperator $\div$ für H ist genau dann eine glatte Kernel-Kontraktion, wenn er Success, Inclusion, Core-Retainment, Uniformity und Relative Closure erfüllt.

Glatte Kernel-Kontraktionen und pm-Kontraktionen

Proposition 5

Jede partial meet Basiskontraktion ist auch eine glatte Kernel-Kontraktion.

Die Umkehrung dieser Proposition gilt jedoch nicht ...

Glatte Kernel-Kontraktionen $\neq$ pm-Kontraktionen!

Es gibt glatte Kernel-Kontraktionen, die keine partial meet Kontraktionen sind:

Beispiel: Sei $H = \{A, B, C\}$. Dann ist

$$H \perp\!\!\!\perp (A(B \vee C)) = \{\{A, B\}, \{A, C\}\}.$$

Die Incision-Funktion σ sei definiert durch $\sigma(H \perp\!\!\!\perp (A(B \vee C))) = \{A, C\}$. Dann ist

$$H \div_\sigma (A(B \vee C)) = \{B\}.$$

Dies ist aber keine partial meet Kontraktion, denn:
Es ist $H \perp (A(B \vee C)) = \{\{A\}, \{B, C\}\}$. Jede Selektionsfunktion S muss eine nichtleere Teilmenge aus $H \perp (A(B \vee C))$ wählen, ist also eine der Mengen $\{\{A\}\}, \{\{B, C\}\}$ oder $\{\{A\}, \{B, C\}\}$.

Damit ist $H \div_S (A(B \vee C))$ eine der Mengen $\{A\}, \{B, C\}$ oder $\emptyset$, jedoch niemals $\{B\} = H \div_\sigma (A(B \vee C))$. ♣

Glatte Kernel-Kontraktionen – Charakterisierung

Theorem 13

Sei H eine Wissensmenge, d.h., $H = Cn(H)$. Ein Basiskontraktionsoperator $\div$ für H ist genau dann eine glatte Kernel-Kontraktion, wenn er eine partial meet Kontraktion ist.

Im AGM-Rahmen fallen also Kernel-Kontraktion und partial meet Kontraktion zusammen, für Wissensbasen ist jedoch die Kernel-Kontraktion eine echte Erweiterung.