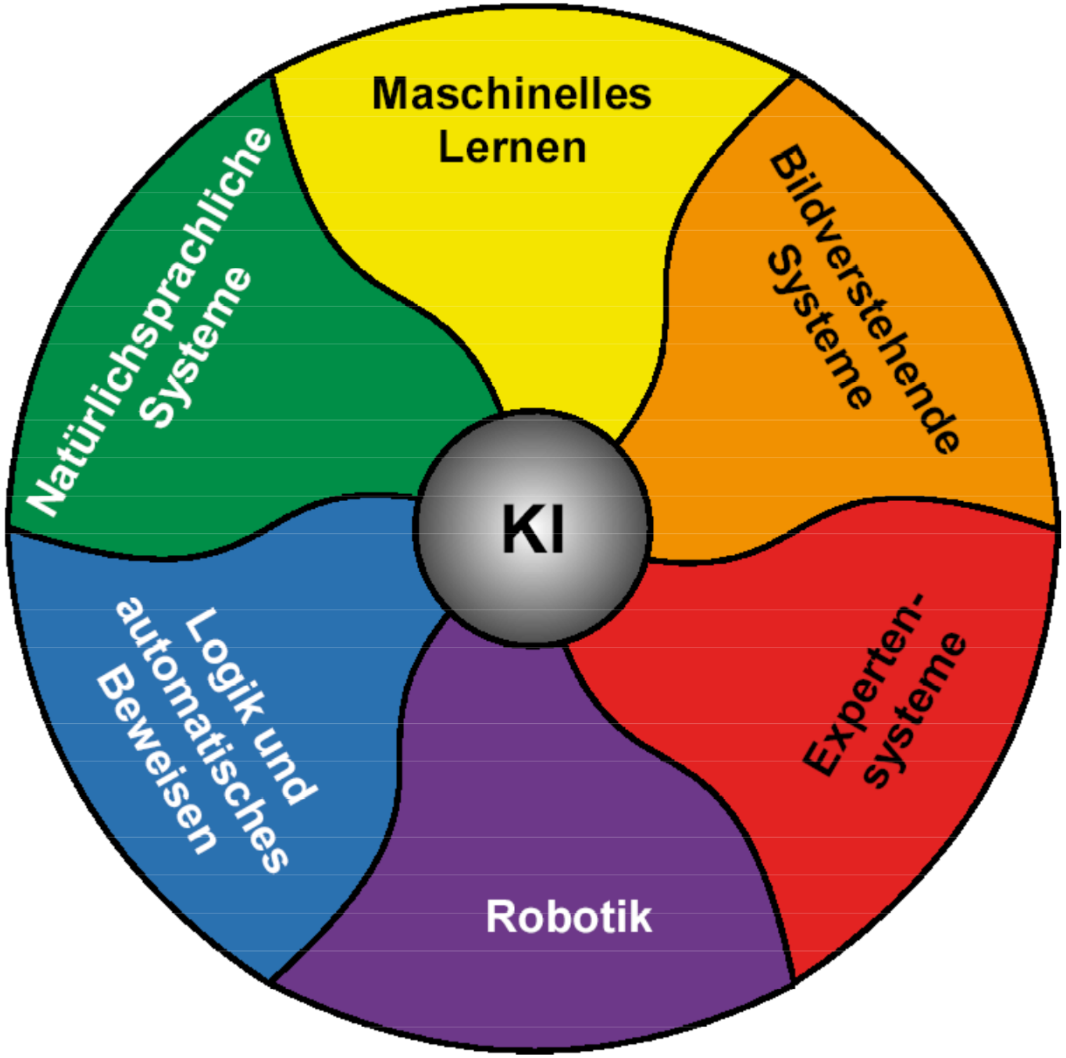
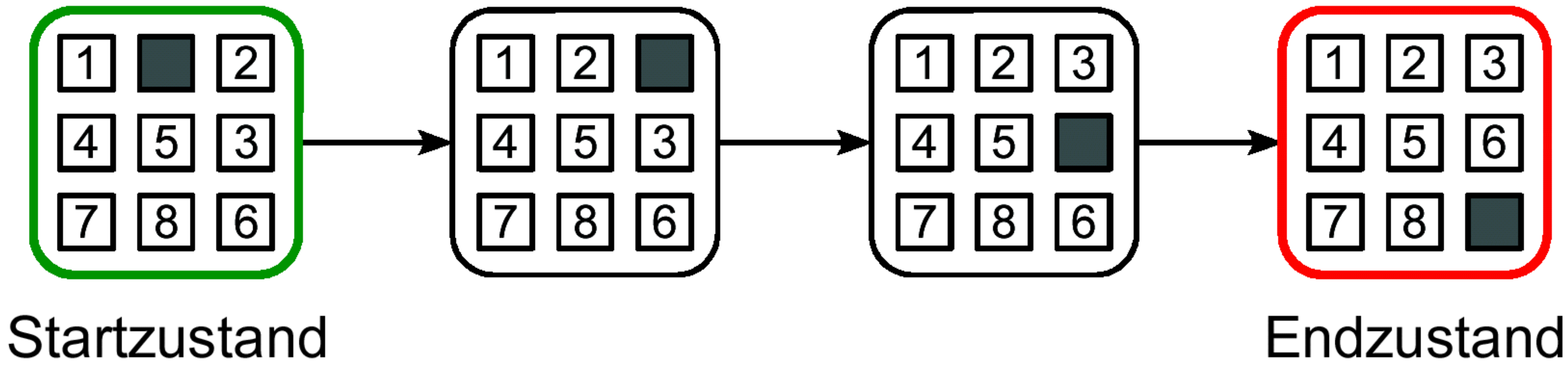
Kapitel 4: Grundlagen intelligenter Systeme

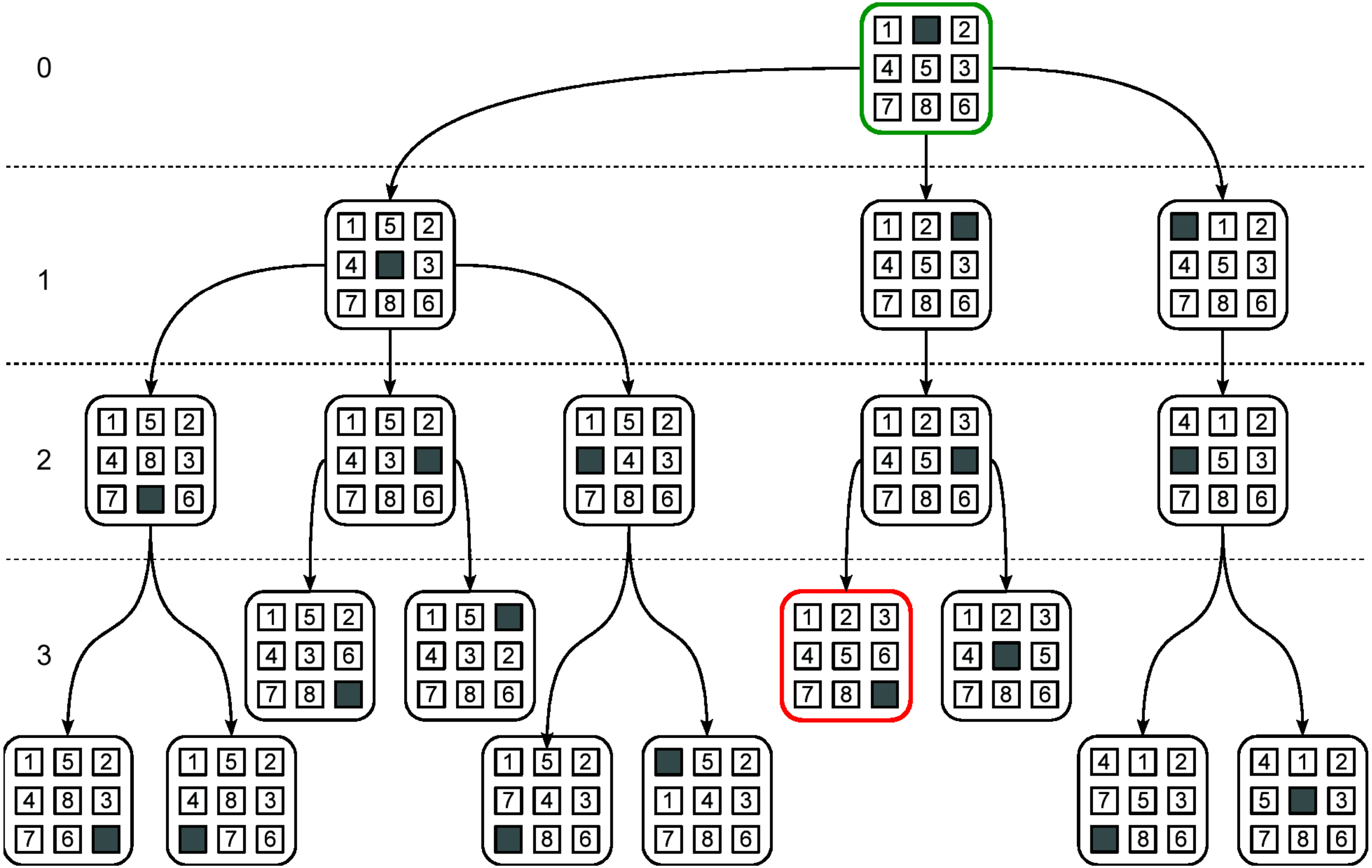
Teilgebiete der „Künstlichen Intelligenz“



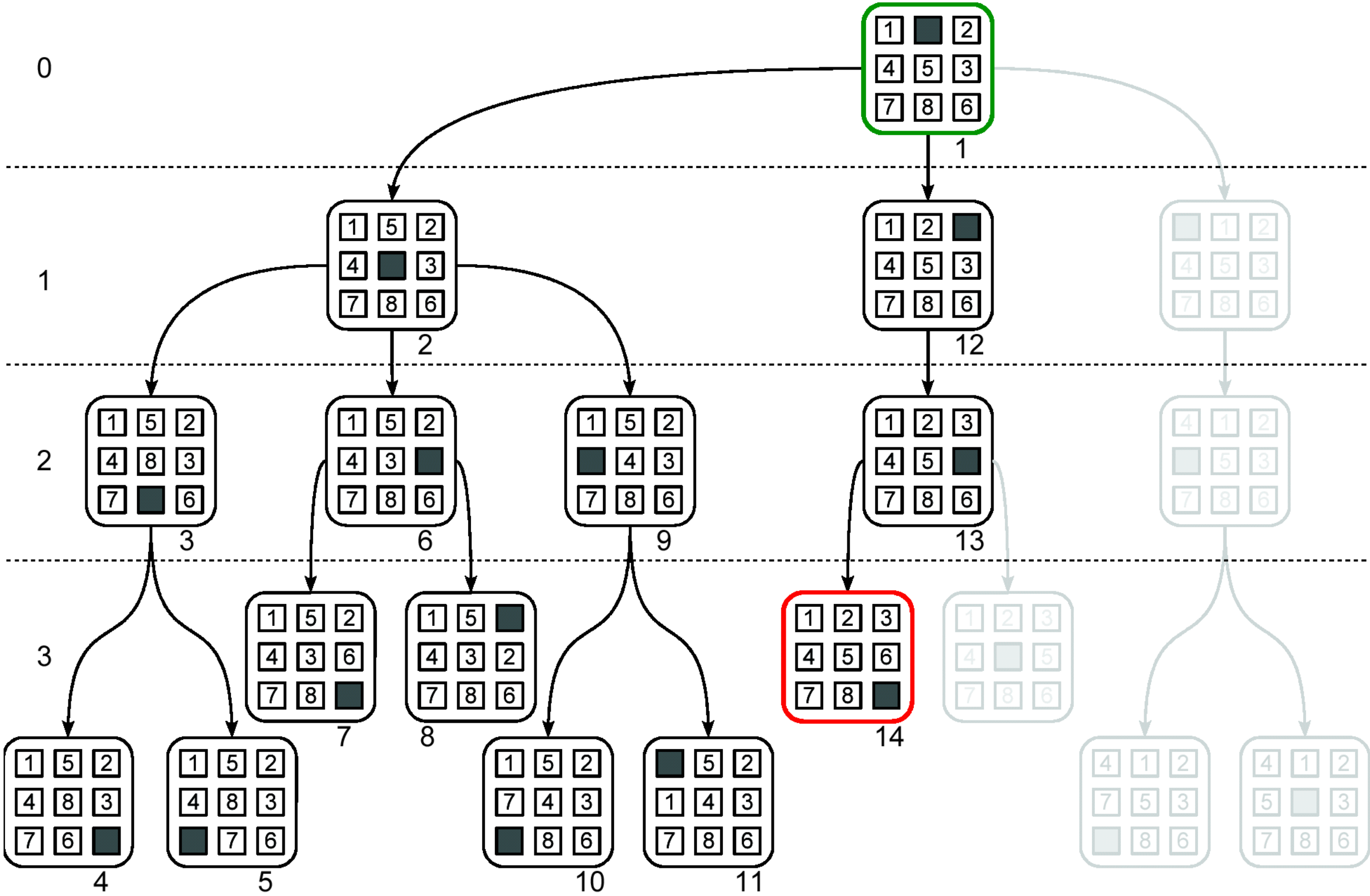
Suchverfahren, Beispiel Schiebepuzzle



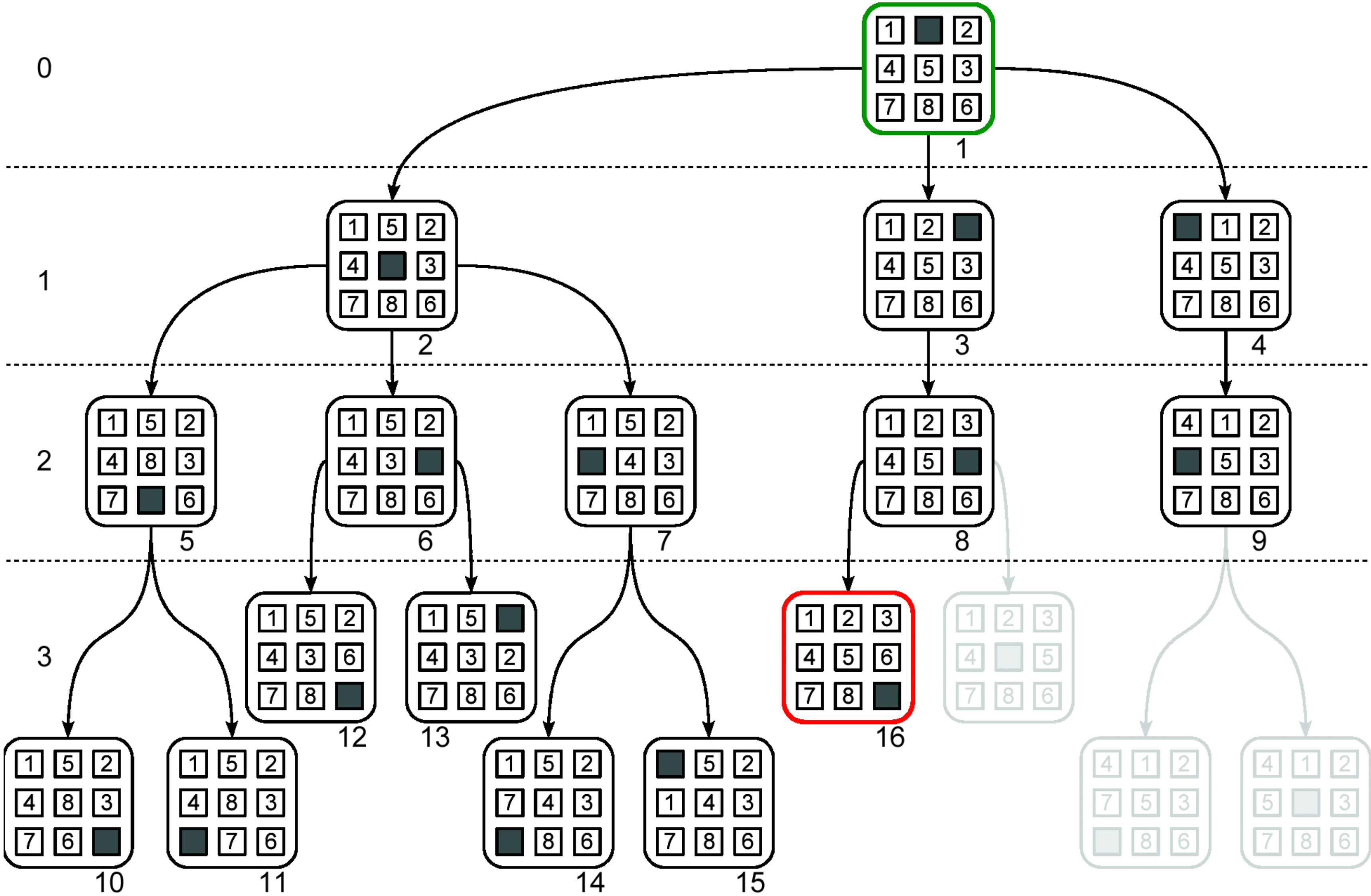
Vollständiger Suchbaum



Tiefensuche



Breitensuche



A und A* - Algorithmus

A – Algorithmus:

$$f(n) = g(n) + h(n)$$

g(n): Anzahl der bisher erfolgten Schritte
h(n): Geschätzte Anzahl der noch erforderlichen Schritte bis zum Zielknoten
f(n): Bewertungsfunktion

Hier: Anzahl der noch falsch platzierten Plättchen

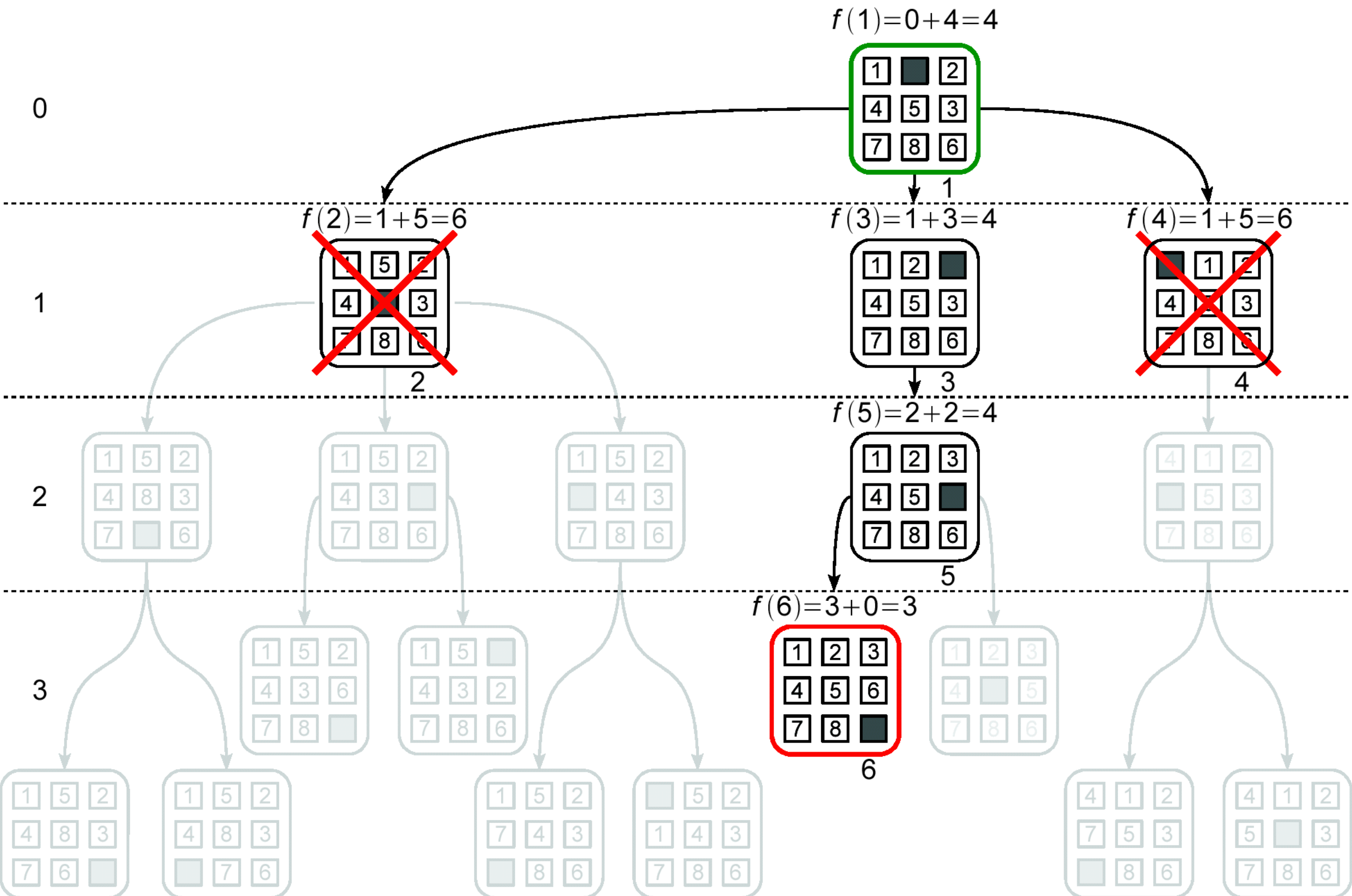
A* – Algorithmus:

$$0 \leq h(n) \leq h^*(n)$$

$h^*(n)$ Exakte Kosten bis zum Zielknoten

Der A*-Algorithmus findet den optimalen (d.h. kürzesten) Pfad

Heuristische Suche



Aufwand der verschiedenen Suchverfahren

	Tiefensuche	Breitensuche	heuristische Suche
Anzahl durchlaufene Knoten	14	16	6

Logik und Theorembeweisen

- ✓ Wichtiges Grundlagengebiet der KI-Forschung
- ✓ Grundlage der meisten regelbasierten KI-Verfahren
- ✓ Algorithmische Darstellung von Intelligenz und Wissen in Maschinen
- ✓ Maschinelle Verarbeitung logischer Schlüsse
- ✓ Wichtigste Teilgebiete: Aussagenlogik, Prädikatenlogik, logisches Schließen (Theorembeweisen)

Aussagenlogik

Verschiedene elementare Verknüpfungen:

UND (Konjunktion) Symbol: $\bullet$

ODER (Disjunktion) Symbol: $+$

NICHT (Negation) Symbol: $\neg$

Zwei Aussagen:

$$A_1 = \text{„Otto wird krank“},\ A_2 = \text{„der Arzt verschreibt Otto eine Medizin.“}$$

UND-Verknüpfung erhält unterschiedlichen Sinn, je nachdem ob man bildet:

$B = A_1 \cdot A_2$ oder $B = A_2 \cdot A_1$

Erfüllbarkeit logischer Aussagen

Erfüllbarkeit diverser Aussagen:

A	B	$A \cdot (\neg B)$	$A \Longrightarrow B \equiv \neg A + B$
1	1	0	1
1	0	1	0
0	1	0	1
0	0	0	1

A • ¬ A ist nicht erfüllbar

A + ¬ A ist eine Tautologie (immer wahr)

Prädikatenlogik

Einfache Prädikate und Argumente:

-Vater (Hans)
- Besitzer (Mann, Auto)
- Verheiratet (x,y)

Beispiele aus dem Bereich der natürlichen Sprache:

1. „Der Mann besitzt ein Auto.“ $\longrightarrow$ Besitzer(Mann, Auto)

2. „Hans und Klara sind verheiratet.“ $\longrightarrow$ Verheiratet(Hans, Klara)

3. „Hans ist mit Klara verheiratet und besitzt ein Auto.“

$$\longrightarrow \underbrace{\text{Verheiratet(Hans, Klara)}}_{\text{Aussage } A} \cdot \underbrace{\text{Besitzer(Hans, Auto)}}_{\text{Aussage } B}.$$

$\longrightarrow$ **A · B entspricht Formel in der Aussagenlogik**

Grundelemente der Prädikatenlogik

- Prädikaten und Funktionen, z. B. $\underline{\text{Verheiratet}}(x, y)$, $\underline{\text{Vater}}(x)$
- Konstanten, z. B. Vater($\underline{\text{Hans}}$)
- Variablen, z. B. Besitzer($\underline{x, y}$)
- Funktionen, z. B. $\underline{\text{f}(}x, y)$
- Negation $\neg$, z. B. $\neg$A
- Disjunktion $+$ (ODER-Verknüpfung)
- Konjunktion $\cdot$ (UND-Verknüpfung)
- Existenz-Quantor $\exists$, z. B. $(\exists x)\text{Vater}(x)$
- All-Quantor $\forall$, z. B. $(\forall x)\text{Vater}(x)$
- Implikation $\Longrightarrow$, z. B. $\text{Mensch}(x) \Longrightarrow \text{Vater}(x)$
- Äquivalenz $\Leftrightarrow$, z. B. $\text{Mensch}(x) \Leftrightarrow \text{Vater}(x)$

Komplexere Beispiele für Prädikatenlogik

Mit den jetzt vorhandenen Grundregeln können komplexere Sachverhalte formuliert werden, z.B.:

1. „In jeder Stadt gibt es einen Bürgermeister."

$$(\forall x)\{\text{Stadt}(x) \Longrightarrow (\exists y)[\text{Mensch}(y) \cdot \text{Bürgermeister}(x, y)]\}$$

2. „Für jede ableitbare Funktion existiert eine ableitbare Umkehrfunktion."

$$(\forall x)\{\text{Funktion}(x)\cdot\text{ableitbar}(x) \Leftrightarrow (\exists y)[\text{Umkehrfunktion}(x, y)\cdot\text{ableitbar}(y)]\}$$

Überprüfen von Theoremen mit Prädikatenlogik

Fakten:

1. „Jeder der lesen kann ist gebildet.": $L \Rightarrow G$
2. „Delphine sind nicht gebildet.": $D \Rightarrow \neg G$
3. „Es gibt intelligente Delphine.": $D \cdot I$

Überprüfen des Theorems:

„Es gibt Intelligente, die nicht lesen können.": $I \cdot (\neg L)$

Wahrheitstabelle

Nr.	L	G	D	I	$L \Longrightarrow G$	$D \Longrightarrow \neg G$	$D \cdot I$	$I \cdot (\neg L)$	$D \cdot L$
0	0	0	0	0	1	1	0	0	0
1	0	0	0	1	1	1	0	1	0
2	0	0	1	0	1	1	0	0	0
4	0	1	0	0	1	1	0	0	0
5	0	1	0	1	1	1	0	1	0
6	0	1	1	0	1	0	0	0	0
7	0	1	1	1	1	0	1	1	0
8	1	0	0	0	0	1	0	0	0
9	1	0	0	1	0	1	0	0	0
10	1	0	1	0	0	1	0	0	1
11	1	0	1	1	0	1	1	0	1
12	1	1	0	0	1	1	0	0	0
13	1	1	0	1	1	1	0	0	0
14	1	1	1	0	1	0	0	0	1
15	1	1	1	1	1	0	1	0	1

Überprüfung einer weiteren Behauptung

Überprüfung der Aussage:

„Delphine können lesen“ ➔ D • L

✓ D • L ist wahr für die Fälle 10, 11, 14 und 15

✓ aber für keinen dieser Fälle sind alle Fakten 1 – 3 auch war

➔ D • L ist nicht erfüllbar

Umformregeln der Prädikatenlogik

1. Doppelte Negation $\neg\neg A \equiv A$

2. Idempotenz $A + A \equiv A$ und $A \cdot A \equiv A$

3. Kommutativität $A \cdot B \equiv B \cdot A$ und $A + B \equiv B + A$

4. Assoziativität $A \cdot (B \cdot C) \equiv (A \cdot B) \cdot C$ und $A + (B + C) \equiv (A + B) + C$

5. Distributvität $A{+}(B{\cdot}C) \equiv (A{+}B){\cdot}(A{+}C)$ und $A \cdot (B + C) \equiv (A \cdot B) + (A \cdot C)$

6. De Morgan $\neg(A \cdot B) \equiv \neg A + \neg B$ und $\neg(A + B) \equiv \neg A \cdot \neg B$

7. Kontrapositiv $A \implies B \equiv \neg B \implies \neg A$

Umformregeln der Prädikatenlogik

8. $A \Longrightarrow B \equiv \neg A + B$

9. $A \Leftrightarrow B \equiv (A \Longrightarrow B) \cdot (B \Longrightarrow A) \equiv (A \cdot B) + (\neg A \cdot \neg B)$

10. $\neg(\forall x)A(x) \equiv (\exists x)(\neg A(x))$

11. $\neg(\exists x)A(x) \equiv (\forall x)(\neg A(x))$

12. $(\forall x)(A(x) \cdot B(x)) \equiv (\forall x)A(x) \cdot (\forall y)B(y)$

13. $(\exists x)(A(x) + B(x)) \equiv (\exists x)A(x) + (\exists y)B(y)$

Umformung auf Standardform

Aufgestelltes Axiom:

$$(\forall x)\{\,\mathrm{A}(x) \Longrightarrow \{\,(\forall y)[\,\mathrm{A}(y) \Longrightarrow \mathrm{A}(f(x,y))\,]\cdot(\,\neg(\forall y)[\mathrm{B}(x,y) \Longrightarrow A(y)]\,)\,\}\,\}.$$

Kann umgeformt werden in eine von den 2 möglichen Standardformen:

Konjunktive Normalform:

$$(\mathrm{A}_1 + \mathrm{A}_2 + \ldots)\cdot(\mathrm{B}_1 + \mathrm{B}_2 + \ldots)\cdot\ldots\cdot(\mathrm{X}_1 + \mathrm{X}_2 + \ldots)\cdot\ldots$$

Disjunktive Normalform:

$$(\mathrm{A}_1 \cdot \mathrm{A}_2 \cdot \ldots) + (\mathrm{B}_1 \cdot \mathrm{B}_2 \cdot \ldots) + \ldots + (\mathrm{X}_1 \cdot \mathrm{X}_2 \cdot \ldots) + \ldots$$

Regeln: Umformung auf Standardform

Regel 1 Eliminierung aller Äquivalenzen $\rightarrow$ Umformregel 9

Regel 2 Eliminierung aller Implikationen $\rightarrow$ Umformregel 8

$$(\forall x)\{\,\neg\mathrm{A}(x) + \{\,(\forall y)[\,\neg\mathrm{A}(y) + \mathrm{A}(f(x,y))\,] \cdot (\,\neg(\forall y)[\,\neg\mathrm{B}(x,y) + \mathrm{A}(y)\,]\,)\,\}\,\}$$

Regel 3 Einziehung der Negation nach innen $\longrightarrow$ Umformregeln 6, 10 und 11

$$(\forall x)\{\,\neg\mathrm{A}(x) + \{\,(\forall y)[\,\neg\mathrm{A}(y) + \mathrm{A}(f(x,y))\,] \cdot (\exists y)[\,\mathrm{B}(x,y) \cdot (\neg\mathrm{A}(y))\,]\,\}\,\}$$

Regel 4 Einführung neuer Variablen für jeden Quantifizierer

$$(\forall x)\{\,\neg\mathrm{A}(x) + \{\,(\forall y)[\,\neg\mathrm{A}(y) + \mathrm{A}(f(x,y))\,] \cdot (\exists w)[\,\mathrm{B}(x,w) \cdot (\neg\mathrm{A}(w))\,]\,\}\,\}$$

Regel 5 Eliminierung aller Existenz-Quantoren

Beispiel $(\forall x)\,\{(\forall y)[\,(\exists z)\mathrm{A}(z)\,]\,\} \equiv (\forall x)\{\,(\forall y)\mathrm{A}(g(x,y))\,\}$

Dabei wurde gesetzt:

$z = g(x,y)$ $\quad$ g: Skolem-Funktion

In diesem Fall beschreibt $g(x,y)$ eine bestimmte Größe z, die eine Funktion von x und y ist.

$$(\forall x)\{\,\neg\mathrm{A}(x) + \{\,(\forall y)[\,\neg\mathrm{A}(y) + \mathrm{A}(f(x,y))\,] \cdot [\,\mathrm{B}(x,g(x)) \cdot (\neg\mathrm{A}(g(x)))\,]\,\}\,\}$$

Regeln: Umformung auf Standardform

Regel 6 Ausklammerung der All-Quantoren und Wegfall dieser Quantoren

$$\{\, \neg \mathrm{A}(x) + \{\, [\, \neg \mathrm{A}(y) + \mathrm{A}(f(x,y))\,] \cdot [\, \mathrm{B}(x, g(x)) \cdot (\neg \mathrm{A}(g(x)))\,]\,\}\,\}$$

Regel 7 Anwendung des Distributivgesetzes zur Transformation in konjunktive Normalform $\rightarrow$ Umformregel 5

$$[\, \neg \mathrm{A}(x) + (\neg \mathrm{A}(y)) + \mathrm{A}(f(x,y))\,] \cdot [\, \neg \mathrm{A}(x) + \mathrm{B}(x, g(x))\,] \cdot [\, \neg \mathrm{A}(x) + (\neg \mathrm{A}(g(x)))\,]$$

Regel 8 Eliminierung der UND-Verknüpfungen durch Auflistung der Klauseln

$\neg \mathrm{A}(x) + (\neg \mathrm{A}(y)) + \mathrm{A}(f(x,y))$ Klausel (1)
$\neg \mathrm{A}(x) + \mathrm{B}(x, g(x))$ Klausel (2)
$\neg \mathrm{A}(x) + (\neg \mathrm{A}(g(x)))$ Klausel (2)

Regel 9 Einführung getrennter Variablen für jede Klausel

$\neg \mathrm{A}(x) + (\neg \mathrm{A}(y)) + \mathrm{A}(f(x,y))$ (1)
$\neg \mathrm{A}(u) + \mathrm{B}(u, g(u))]$ (2)
$\neg \mathrm{A}(v) + (\neg \mathrm{A}(g(v)))$ (3)

Resolutionsverfahren

Grundform:

$$A_1 + A_2 + \ldots + A_n + P$$
$$B_1 + B_2 + \ldots + B_n + \neg P$$

Resolvente:

$$A_1 + A_2 + \ldots + A_n + B_1 + B_2 + \ldots + B_n \equiv R$$

Sonderfälle:

1. A
   $A \Longrightarrow B \equiv \neg A + B$ $\qquad R \equiv B$
2. $A + B$
   $\neg A + B$ $\qquad R \equiv B + B \equiv B.$
3. A
   $\neg A$ $\qquad R \equiv NIL$
4. $A \Longrightarrow B \equiv \neg A + B$
   $B \Longrightarrow C \equiv \neg B + C$ $\qquad R \equiv \neg A + C \equiv A \Longrightarrow C$

Theorembeweisen

Gegeben:

Satz (Set) S = $\{S_1, S_2, \ldots\ldots, S_n\}$ von n existierenden Axiomen (Fakten oder Behauptungen).

T ist ein zu beweisendes Theorem.

Theorembeweis: Bilden des erweiterten Sets $S^* = \{S_1, S_2, \ldots\ldots, S_n, \neg T\}$ und Anwendung des Resolutionsverfahrens auf die Klauseln in S* bis zur Erzeugung der leeren Klausel.

Erklärung: Wenn T war ist und aus S folgt, dann machen alle Belegungen, die S wahr machen, auch T wahr. Keine dieser Konfigurationen macht dann ¬T wahr, daher kann keine dieser Konfigurationen $S^* = S \bullet \neg T$ wahr machen. Daher ist S* unerfüllbar und Resolution muss zu leerer Klausel führen.

Beispiel für Theorembeweis mit Resolution

Aus Behauptungen für intelligente Delphine:

S:	$L \rightarrow G$	(1)
	$D \rightarrow \neg G$	(2)
	$D \cdot I$	(3)
T:	$I \cdot (\neg L)$	(4)

Umformung:

(aus 1)	$\neg L + G$	(5)
(aus 2)	$\neg D + \neg G$	(6)
(aus 3)	D	(7)
(aus 3)	I	(8)
$\neg$T:	$\neg I + L$	(9)

Resolutionsprozess:

(aus 8+9):	$\longrightarrow$ L	(10)
(aus 10+5):	$\longrightarrow$ G	(11)
(aus 11+6):	$\longrightarrow$ $\neg$D	(12)
(aus 12+7):	$\longrightarrow$ NIL	

Wissensrepräsentation

- ✓ notwendig, um das Wissen strukturiert darzustellen und zu formulieren
- ✓ um das Wissen über komplexe Systeme nachvollziehbar darstellen zu können, benötigt man ein bestimmtes Darstellungsschema
- ✓ Wissensrepräsentation notwendig, um umfangreiches Spezialwissen eines Experten schematisch zu extrahieren und strukturiert darzustellen
- ✓ mit Hilfe des Repräsentationsmechanismus kann das Wissen interpretierbar gemacht werden
- ✓ mit einem geeigneten Inferenzmechanismus (gehört nicht zur Wissensrepräsentation!) kann das gespeicherte Wissen verarbeitet werden

- ✓ Populärste Methoden der Wissensrepräsentation:
  - Prädikatenlogik
  - Produktionsregeln
  - Semantische Netze
  - Rahmen

Prädikatenlogik zur Wissensrepräsentation

- ✓ Prädikatenlogik ist die grundlegende Art zur Darstellung von Wissen
- ✓ alle anderen Wissensrepräsentationen bauen praktisch implizit auf der Prädikatenlogik auf
- ✓ Aufteilung des Wissens in Fakten und Regeln möglich
- ✓ durch Umformung in konjunktive Normalform ergibt sich standardisierte Form des Wissens
- ✓ das auf diese Weise dargestellte Wissen kann mit Hilfe des Resolutionsverfahrens abgearbeitet werden
- ✓ das bedeutet, der hier verwendete Inferenzmechanismus ist die Resolution
- ✓ Nachteile:
  - ➢ Formulierung des Wissens aufwändig und unnatürlich, entspricht nicht der Umgangsform
  - ➢ Umformung in Normalform notwendig

Produktionsregeln

- ✓ verwenden Schreibweise der Prädikatenlogik
- ✓ Hauptunterschiede zur klassischer Prädikatenlogik:
  - keine Umformung in konjunktive Normalform, Wenn-Dann-Regelstrukturen bleiben erhalten
  - verwendet als Inferenzmechanismus nicht Resolution, sondern Vorwärts- und Rückwärtsverkettung
  - Inferenz in der Prädikatenlogik bestimmt den Wahrheitswert eines Theorems
  - Inferenz bei Produktionsregeln impliziert aus dem Wenn-Teil einer Regel den Dann-Teil (bei Vorwärtsverkettung)
- ✓ grundlegende Struktur von Produktionsregeln:
  - Fakten (wie in Prädikatenlogik)
  - Wenn-Dann-Regeln (wie in Prädikatenlogik vor der Umformung)

Beispiel für Produktionsregeln

Wissensbasiertes System für Unterstützung beim Autokauf:

Regel 1: $\mathrm{GP} \cdot \mathrm{D} \cdot 3\mathrm{T} + \mathrm{Tu} \cdot 5\mathrm{T} \;\Rightarrow\; \mathrm{Golf}$

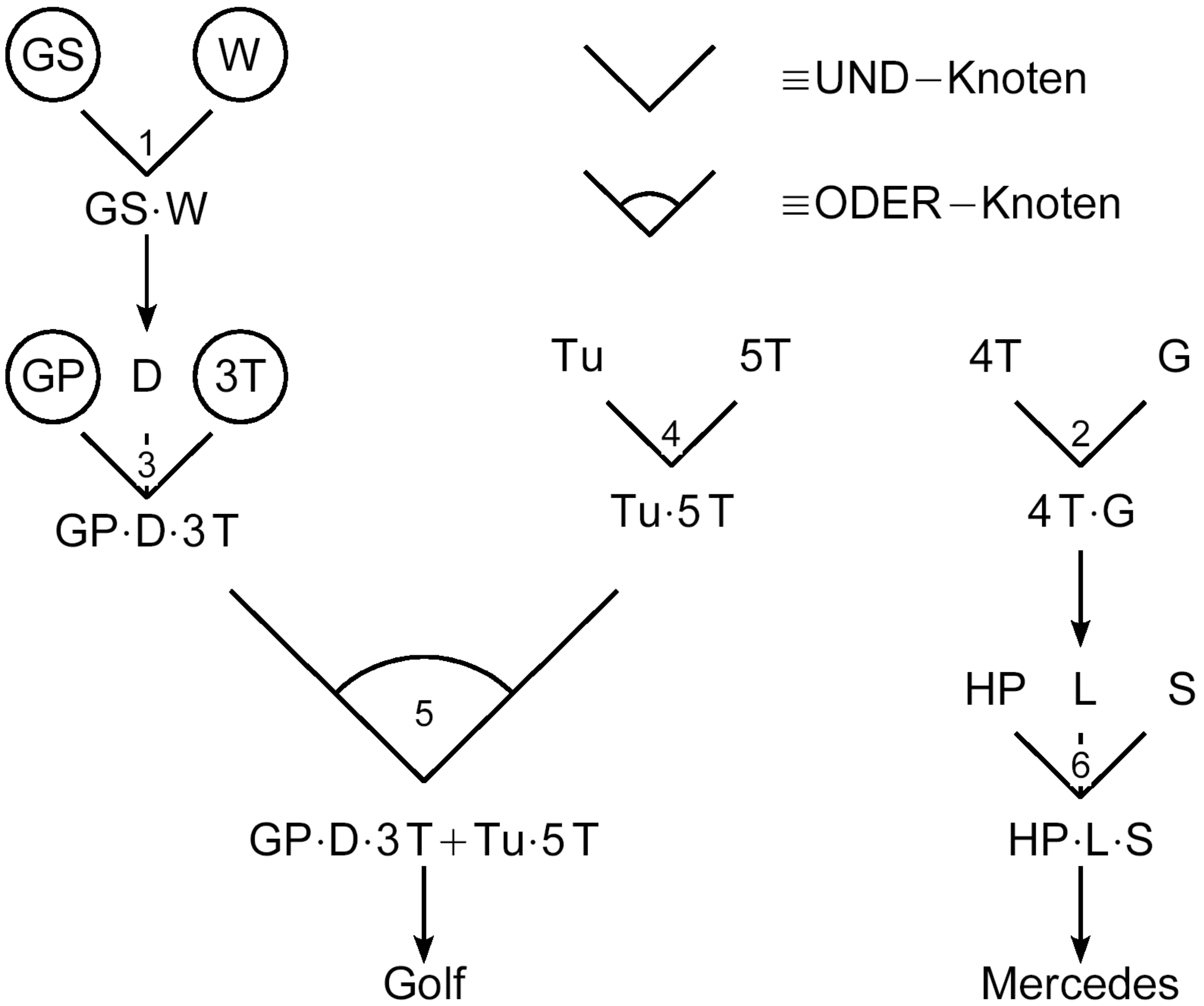
Regel 2: $\mathrm{HP} \cdot \mathrm{L} \cdot \mathrm{S} \;\Rightarrow\; \mathrm{Mercedes}$

Regel 3: $\mathrm{GS} \cdot \mathrm{W} \;\Rightarrow\; \mathrm{D}$

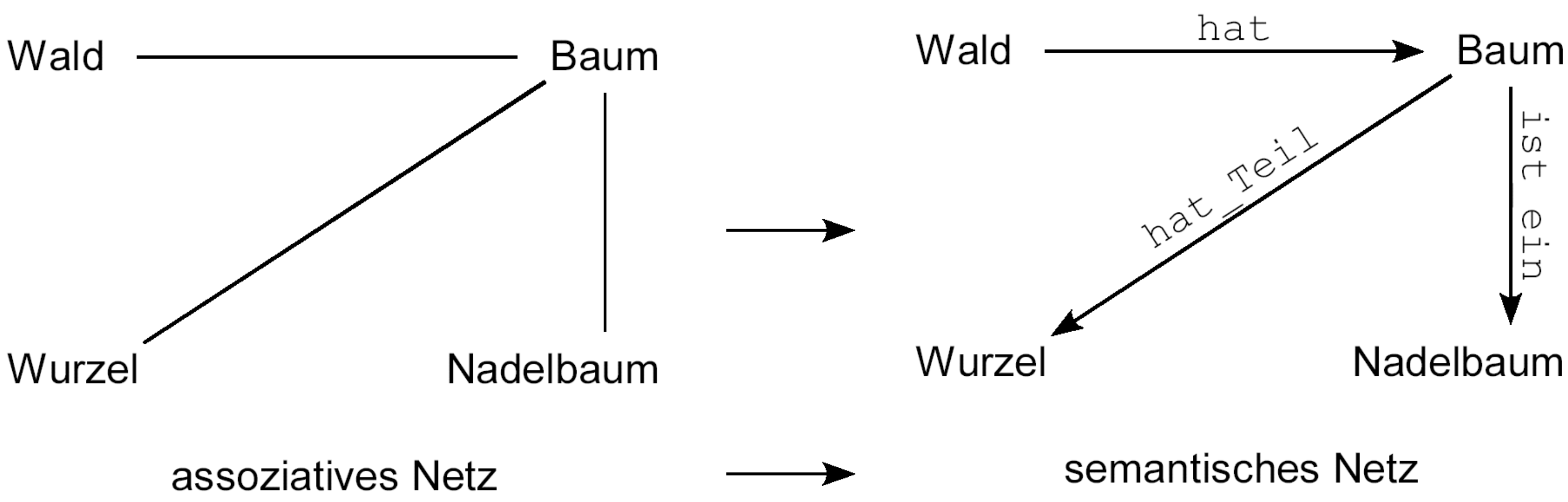
Regel 4: $4\mathrm{T} \cdot \mathrm{G} \;\Rightarrow\; \mathrm{L}$

Fakten: $3\mathrm{T}, \mathrm{GS}, \mathrm{W}, \mathrm{GP}$

Regelwerk und Fakten



Semantische Netze



Beziehung Prädikatenlogik ↔ semant. Netze

$$\text{farbe(gelb, Haus)} \Longrightarrow \text{gelb} \xrightarrow{\text{farbe}} \text{Haus}$$

Darstellung eines Computers als Frame

Frame-Name

Computer

Slots

CPU	Mikroprozessor
Hauptspeicher	16 MB
Betriebssystem	Unix
Peripherie	Standardausrüstung
⋮	⋮

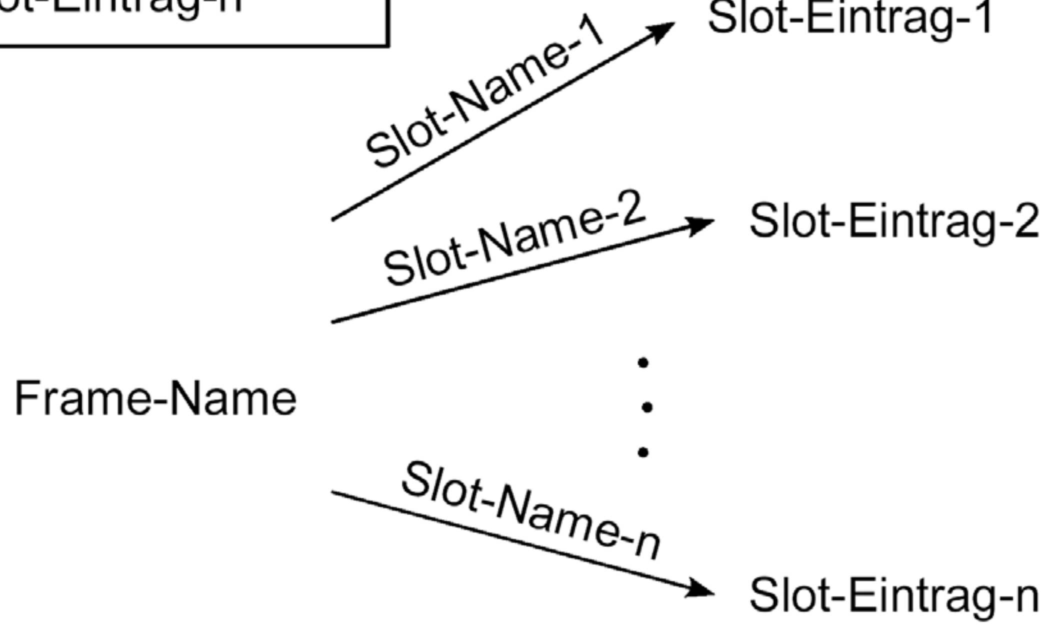
Fillers

können wieder Frames sein

Vergleich zw. Frame und semantischem Netz

Frame-Name

Slot-Name-1	Slot-Eintrag-1
Slot-Name-2	Slot-Eintrag-2
⋮	⋮
Slot-Name-n	Slot-Eintrag-n



Frame-Abgleich

Frage lautet hier: Welche Blumen mit gelber Farbe gibt es ?

?

Bezeichnung	Blume
Farbe	Aktuell {gelb}

gesucht wird hier nach dem Frame-Namen

Antwort wird gefunden durch Suche nach den Namen von Frames, welche die gewünschte Wissensstruktur erfüllen