

Künstliche Intelligenz

Prof. J. Fürnkranz

Technische Universität Darmstadt — Wintersemester 2009/10

Termin: 10. 3. 2010

Name:

Vorname:

Matrikelnummer:

Fachrichtung:

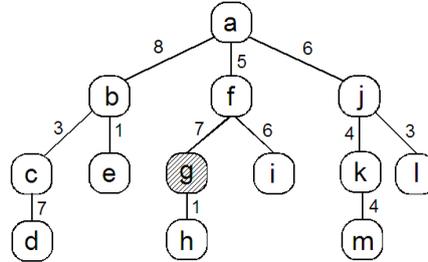
Punkte: (1) (2) (3) (4) (5) **Summe:**

- **Aufgaben:** Diese Klausur enthält auf den folgenden Seiten 5 Aufgaben zu insgesamt 100 Punkten. Jede Aufgabe steht auf einem eigenen Blatt. Kontrollieren Sie *sofort*, ob Sie alle sechs Blätter erhalten haben!
- **Zeiteinteilung:** Die Zeit ist knapp bemessen. Wir empfehlen Ihnen, sich zuerst einen kurzen Überblick über die Aufgabenstellungen zu verschaffen, und dann mit den Aufgaben zu beginnen, die Ihnen am besten liegen.
- **Papier:** Verwenden Sie nur Papier, das Sie von uns ausgeteilt bekommen. Bitte lösen Sie die Aufgaben auf den dafür vorgesehenen Seiten. Falls der Platz nicht ausreicht, vermerken sie dies bitte und setzen die Lösung auf der letzten Seite fort. Brauchen Sie zusätzlich Papier (auch Schmierpapier), bitte melden.
- **Fragen:** Sollten Sie Teile der Aufgabenstellung nicht verstehen, bitte fragen Sie!
- **Abschreiben:** Sollte es sich herausstellen, dass Ihre Lösung und die eines Kommilitonen über das zu erwartende Maß hinaus übereinstimmen, werden beide Arbeiten negativ beurteilt (ganz egal wer von wem in welchem Umfang abgeschrieben hat).
- **Ausweis:** Legen Sie Ihren *Studentenausweis* und *Lichtbildausweis* sichtbar auf Ihren Platz. Füllen Sie das Deckblatt sofort aus!
- **Hilfsmittel:** Zur Lösung der Aufgaben ist ein von Ihnen selbst handschriftlich beschriebenes DIN-A4-Blatt erlaubt. Gedruckte Wörterbücher sind für ausländische Studenten erlaubt, elektronische Hilfsmittel (Taschenrechner, elektronische Wörterbücher, Handy, etc.) sind verboten! Sollten Sie etwas verwenden wollen, was nicht in diese Kategorien fällt, bitte klären Sie das *bevor* Sie zu arbeiten beginnen.
- **Aufräumen:** Sonst darf außer Schreibgerät, Essbarem, von uns ausgeteiltem Papier und eventuell Wörterbüchern nichts auf Ihrem Platz liegen. Taschen bitte unter den Tisch!

Gutes Gelingen!

Aufgabe 1 Suche (22 Punkte)

Betrachten Sie folgenden Suchbaum für eine beim Knoten a beginnenden Suche



Die Zahlen neben den Kanten geben die Kosten an, die benötigt werden, um über diese Kante zu wandern.

1-a Geben Sie für die folgenden Suchverfahren jeweils den Suchrand (die *Fringe* bzw. *Open-List*) zu dem Zeitpunkt an, als g zur Expansion ausgewählt wird. Beachten Sie auch die Reihenfolge der Knoten in der Fringe!

1. Tiefensuche (ohne Berücksichtigung der Kosten)
2 Punkte g, i, j
2. Breitensuche (ohne Berücksichtigung der Kosten)
2 Punkte g, i, k, l, d
3. Uniform Cost Search. Geben Sie hier auch die Kosten der Knoten in der Fringe an.
4 Punkte $g : 12, m : 14, d : 18$

1-b Führen Sie, beginnend beim Knoten a , eine A*-Suche aus bis der Knoten g zur Expansion selektiert wird, wobei folgende Heuristik h verwendet wird:

	a	b	c	d	e	f	g	h	i	j	k	l	m
$h(\cdot) =$	8	1	3	7	4	5	0	3	6	2	1	5	4

Geben Sie alle Zwischenschritte an, d.h. welche Knoten jeweils in der Fringe-List sind (d.h. offen sind), deren Werte für $g(n)$, $h(n)$ und $f(n)$ sowie, welcher Knoten jeweils ausgewählt wird bzw. warum die Suche abgebrochen wird.

8 Punkte

<table style="border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr><th style="border-right: 1px solid black; border-bottom: 1px solid black;">Node</th><th style="border-bottom: 1px solid black;">f</th><th style="border-bottom: 1px solid black;">g</th><th style="border-bottom: 1px solid black;">h</th></tr> </thead> <tbody> <tr><td style="border-right: 1px solid black;">1. a</td><td>8</td><td>0</td><td>8 ←</td></tr> <tr><td style="border-right: 1px solid black;">b</td><td>9</td><td>8</td><td>1</td></tr> <tr><td style="border-right: 1px solid black;">f</td><td>10</td><td>5</td><td>5</td></tr> <tr><td style="border-right: 1px solid black;">j</td><td>8</td><td>6</td><td>2 ←</td></tr> </tbody> </table> <table style="border-collapse: collapse;"> <thead> <tr><th style="border-right: 1px solid black; border-bottom: 1px solid black;">Node</th><th style="border-bottom: 1px solid black;">f</th><th style="border-bottom: 1px solid black;">g</th><th style="border-bottom: 1px solid black;">h</th></tr> </thead> <tbody> <tr><td style="border-right: 1px solid black;">4. k</td><td>11</td><td>10</td><td>1 ←</td></tr> <tr><td style="border-right: 1px solid black;">l</td><td>14</td><td>9</td><td>5</td></tr> <tr><td style="border-right: 1px solid black;">c</td><td>14</td><td>11</td><td>3</td></tr> <tr><td style="border-right: 1px solid black;">e</td><td>13</td><td>9</td><td>4</td></tr> <tr><td style="border-right: 1px solid black;">g</td><td>12</td><td>12</td><td>0</td></tr> <tr><td style="border-right: 1px solid black;">i</td><td>17</td><td>11</td><td>6</td></tr> </tbody> </table>	Node	f	g	h	1. a	8	0	8 ←	b	9	8	1	f	10	5	5	j	8	6	2 ←	Node	f	g	h	4. k	11	10	1 ←	l	14	9	5	c	14	11	3	e	13	9	4	g	12	12	0	i	17	11	6	<table style="border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr><th style="border-right: 1px solid black; border-bottom: 1px solid black;">Node</th><th style="border-bottom: 1px solid black;">f</th><th style="border-bottom: 1px solid black;">g</th><th style="border-bottom: 1px solid black;">h</th></tr> </thead> <tbody> <tr><td style="border-right: 1px solid black;">2. b</td><td>9</td><td>8</td><td>1 ←</td></tr> <tr><td style="border-right: 1px solid black;">f</td><td>10</td><td>5</td><td>5</td></tr> <tr><td style="border-right: 1px solid black;">k</td><td>11</td><td>10</td><td>1</td></tr> <tr><td style="border-right: 1px solid black;">l</td><td>14</td><td>9</td><td>5</td></tr> </tbody> </table> <table style="border-collapse: collapse;"> <thead> <tr><th style="border-right: 1px solid black; border-bottom: 1px solid black;">Node</th><th style="border-bottom: 1px solid black;">f</th><th style="border-bottom: 1px solid black;">g</th><th style="border-bottom: 1px solid black;">h</th></tr> </thead> <tbody> <tr><td style="border-right: 1px solid black;">5. l</td><td>14</td><td>9</td><td>5</td></tr> <tr><td style="border-right: 1px solid black;">c</td><td>14</td><td>11</td><td>3</td></tr> <tr><td style="border-right: 1px solid black;">e</td><td>13</td><td>9</td><td>4</td></tr> <tr><td style="border-right: 1px solid black;">g</td><td>12</td><td>12</td><td>0 ←</td></tr> <tr><td style="border-right: 1px solid black;">i</td><td>17</td><td>11</td><td>6</td></tr> <tr><td style="border-right: 1px solid black;">m</td><td>18</td><td>14</td><td>4</td></tr> </tbody> </table>	Node	f	g	h	2. b	9	8	1 ←	f	10	5	5	k	11	10	1	l	14	9	5	Node	f	g	h	5. l	14	9	5	c	14	11	3	e	13	9	4	g	12	12	0 ←	i	17	11	6	m	18	14	4	<table style="border-collapse: collapse;"> <thead> <tr><th style="border-right: 1px solid black; border-bottom: 1px solid black;">Node</th><th style="border-bottom: 1px solid black;">f</th><th style="border-bottom: 1px solid black;">g</th><th style="border-bottom: 1px solid black;">h</th></tr> </thead> <tbody> <tr><td style="border-right: 1px solid black;">3. f</td><td>10</td><td>5</td><td>5 ←</td></tr> <tr><td style="border-right: 1px solid black;">k</td><td>11</td><td>10</td><td>1</td></tr> <tr><td style="border-right: 1px solid black;">l</td><td>14</td><td>9</td><td>5</td></tr> <tr><td style="border-right: 1px solid black;">c</td><td>14</td><td>11</td><td>3</td></tr> <tr><td style="border-right: 1px solid black;">e</td><td>13</td><td>9</td><td>4</td></tr> </tbody> </table>	Node	f	g	h	3. f	10	5	5 ←	k	11	10	1	l	14	9	5	c	14	11	3	e	13	9	4
Node	f	g	h																																																																																																																							
1. a	8	0	8 ←																																																																																																																							
b	9	8	1																																																																																																																							
f	10	5	5																																																																																																																							
j	8	6	2 ←																																																																																																																							
Node	f	g	h																																																																																																																							
4. k	11	10	1 ←																																																																																																																							
l	14	9	5																																																																																																																							
c	14	11	3																																																																																																																							
e	13	9	4																																																																																																																							
g	12	12	0																																																																																																																							
i	17	11	6																																																																																																																							
Node	f	g	h																																																																																																																							
2. b	9	8	1 ←																																																																																																																							
f	10	5	5																																																																																																																							
k	11	10	1																																																																																																																							
l	14	9	5																																																																																																																							
Node	f	g	h																																																																																																																							
5. l	14	9	5																																																																																																																							
c	14	11	3																																																																																																																							
e	13	9	4																																																																																																																							
g	12	12	0 ←																																																																																																																							
i	17	11	6																																																																																																																							
m	18	14	4																																																																																																																							
Node	f	g	h																																																																																																																							
3. f	10	5	5 ←																																																																																																																							
k	11	10	1																																																																																																																							
l	14	9	5																																																																																																																							
c	14	11	3																																																																																																																							
e	13	9	4																																																																																																																							

Anmerkung: g wird hier generiert, aber noch nicht zur Expansion ausgewählt, da k eventuell noch eine billigere Lösung sein könnte!

Anmerkung: g wird hier zur Expansion ausgewählt. Da es keine günstigere Lösung geben kann, wird hier gestoppt.

1-c Nehmen Sie eine Heuristik h_0 an, die alle Kosten mit 0 schätzt, d.h. $h_0(\cdot) = 0$ für alle Knoten. Beantworten Sie folgende Fragen und begründen Sie jeweils Ihre Antwort.

1. Ist die Heuristik h_0 zulässig?

1 Punkte Ja, da sie alle Kosten unterschätzt.

2. Ist die Heuristik h_0 konsistent?

1,5 Punkte Ja, da jeder Kantenübergang Kosten ≥ 0 hat, und diese Kosten mit 0 geschätzt werden.

3. Wird A* mit h_0 im allgemeinen eine optimale Lösung finden?

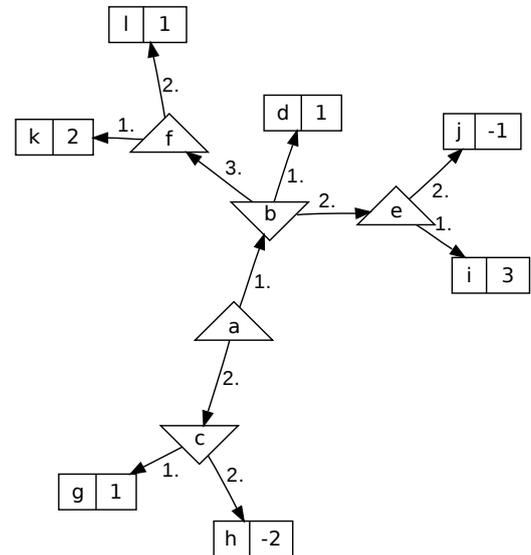
1,5 Punkte Ja, da h_0 zulässig ist.

4. A* mit h_0 ist äquivalent zu einem anderen, in der Vorlesung kennengelernten Algorithmus. Zu welchem?

2 Punkte Zur Uniform Cost Search (siehe 1-b-3.)

Aufgabe 2 Spielbaum-Suche (20 Punkte)

In nebenstehender Grafik ist ein Spielbaum abgebildet. Die Evaluierungen an den Blättern, dargestellt durch Rechtecke (Knotenname, Evaluierungswert) stellen die spieltheoretischen Werte aus der Sicht des MAX-Spielers dar. MAX-Knoten sind jeweils als Dreiecke mit der Spitze nach oben (\triangle) und MIN-Knoten jeweils als Dreiecke mit der Spitze nach unten (∇) dargestellt. Gerichtete Kanten, die von einem Knoten ausgehen, zeigen auf die Nachfolger des Knoten. Darüberhinaus ist die Reihenfolge der Nachfolger mit den Zahlen an den Kanten vorgegeben.



2-a Geben Sie den MiniMax-Wert des Knotens a an.

2 Punkte $MiniMax(a) = 1$

2-b Führen Sie eine Alpha-Beta Suche durch, wobei der Knoten a mit einem alpha-beta-Fenster von $[0, 5]$ aufgerufen wird. Die Nachfolger eines Knotens werden in der jeweils angegebenen Reihenfolge durchsucht.

Nehmen Sie an, daß der Alpha-Beta-Algorithmus wie in der Vorlesung durch folgende beiden Funktionen implementiert ist:

- $MIN-VALUE(State, \alpha, \beta)$: berechnet und retourniert den Wert des Min-Knotens $State$ mit gegebenem α und β .
- $MAX-VALUE(State, \alpha, \beta)$: berechnet und retourniert den Wert des Max-Knotens $State$ mit gegebenem α und β .

In jeder dieser beiden Funktionen werden nun unmittelbar nach dem Funktionsaufruf die übergebenen Parameter auf dem Bildschirm ausgegeben (also z.B. durch `println(State, α , β)`).

Geben Sie die Ausgaben beim Durchsuchen des Baumes mit dem Aufruf $MAX-VALUE(a, 0, 5)$ an.

8 Punkte

$a, 0, 5$

$b, 0, 5$

$d, 0, 5$

$e, 0, 1$

$i, 0, 1$

$f, 0, 1$

$k, 0, 1$

$c, 1, 5$

$g, 1, 5$

Die Knoten j, l und h werden gepruned.

- 2-c Der Max-Spieler nimmt bei seiner Zugauswahl an, daß der Min-Spieler immer den Zug macht, der das Ergebnis minimiert. Was passiert, wenn der Min-Spieler im tatsächlichen Spiel andere Züge macht?
- Der Max-Spieler muß einen neuen Spiel-Baum aufbauen
 - Der Max-Spieler wird ein zumindest gleich gutes Ergebnis erzielen
 - Der Min-Spieler kann dadurch möglicherweise den Max-Spieler überlisten
 - Der Max-Spieler sollte Minimax-Suche statt Alpha-Beta einsetzen

Begründen Sie Ihre Antwort(en).

5 Punkte Beabsichtigte Lösung war Punkt 2. Die Spielbaum-Suche maximiert für den Max-Spieler die untere Schranke für das Endergebniss. Die untere Schranke tritt ein, falls der Gegenspieler auch optimal spielt. Weicht der Gegner davon ab, kann kein schlechteres Ergebnis resultieren.

Punkt 1 und 4 werden auch als korrekt bewertet, falls sie ausreichend begründet werden. Wenn man sich bei der Spielbaumsuche sich primär für die Ermittlung einer guten Strategie interessiert, können Punkt 1 und 4 entfernt sinnvoll sein.

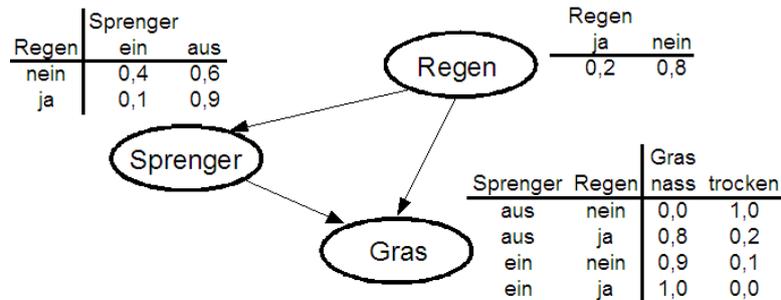
Punkt 2 muss aber in jedem Fall angekreuzt sein. Bei Fehlen gab es Punktabzug. Punktabzug gab es auch bei Ankreuzen von Punkt 3. Bei nichtausreichender Begründung von Punkt 1 und 4 werden Teilpunkte vergeben.

- 2-d In der Vorlesung haben wir eine Erweiterung des Minimax-Algorithmus für mehrere Spieler kennen gelernt. Erklären Sie kurz die Funktionsweise einer Variante für drei Spieler.

5 Punkte

Aufgabe 3 Bayes'sche Netze (16 Punkte)

Gegeben sei folgendes Bayes'sches Netz, das die Ereignisse Regen (ja/nein), (Rasen-)Sprenger (ein/aus) und Gras (nass/trocken) und deren Zusammenhänge repräsentiert.



Beantworten Sie folgende Fragen, und geben Sie jeweils eine kurze Begründung bzw. eine entsprechende Berechnung des Ergebnisses an:

3-a Wie groß ist die Wahrscheinlichkeit, daß das Gras nass ist, es regnet und der Rasensprenger aus ist?

4 Punkte

$$\begin{aligned} P(g_{nass}, r_{ja}, s_{aus}) &= P(g_{nass}|s_{aus}, r_{ja}) \times P(s_{aus}|r_{ja}) \times P(r_{ja}) = \\ &= 0.8 \times 0.9 \times 0.2 = 0.144 \end{aligned}$$

3-b Wie groß ist die Wahrscheinlichkeit, daß das Gras nass ist, wenn es regnet und der Rasensprenger aus ist?

2 Punkte

$$P(g_{nass}|r_{ja}, s_{aus}) = 0.8$$

Anmerkung: Diesen Wert kann man direkt aus der Tabelle ablesen.

3-c Wie groß ist die Wahrscheinlichkeit, daß es regnet und der Rasensprenger aus ist, wenn das Gras nass ist?

6 Punkte

$$P(r_{ja}, s_{aus}|g_{nass}) = \frac{P(g_{nass}, r_{ja}, s_{aus})}{P(g_{nass})} = \frac{0.144}{0.452} = \frac{36}{113} \approx 0.32$$

da

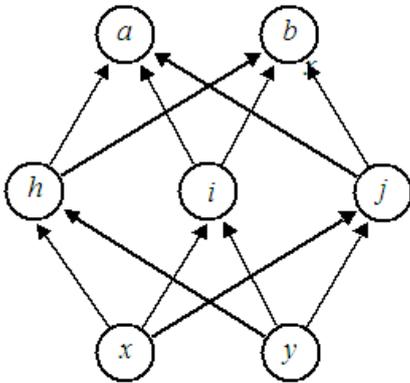
$$\begin{aligned} P(g_{nass}) &= \sum_{r \in \{ja, nein\}} \sum_{s \in \{aus, ein\}} P(g_{nass}|r, s) \times P(s|r) \times P(r) \\ &= P(g_{nass}|r_{ja}, s_{aus}) \times P(s_{aus}|r_{ja}) \times P(r_{ja}) + \\ &\quad P(g_{nass}|r_{ja}, s_{ein}) \times P(s_{ein}|r_{ja}) \times P(r_{ja}) + \\ &\quad P(g_{nass}|r_{nein}, s_{aus}) \times P(s_{aus}|r_{nein}) \times P(r_{nein}) + \\ &\quad P(g_{nass}|r_{nein}, s_{ein}) \times P(s_{ein}|r_{nein}) \times P(r_{nein}) \\ &= 0.8 \times 0.9 \times 0.2 + \\ &\quad 1.0 \times 0.1 \times 0.2 + \\ &\quad 0.0 \times 0.6 \times 0.8 + \\ &\quad 0.9 \times 0.4 \times 0.8 \\ &= 0.144 + 0.020 + 0 + 0.288 = 0.452 \end{aligned}$$

- 3-d Erklären Sie kurz die Grundidee von Rejection Sampling zur Bestimmung einer approximativen Wahrscheinlichkeit für Queries an ein Bayes'sches Netz.

4 Punkte Beim Rejection Sampling werden zufällig Instanzen für eine Query $P(X|e)$ generiert. Wenn eine Instanz nicht mit den Evidenzwerten e übereinstimmt, wird sie verworfen, ansonsten wird der Zähler für die beobachtete Ausprägung der X -Variablen hochgezählt. Nach vielen Iterationen konvergieren die aus der Normalisierung dieser Zähler geschätzten Wahrscheinlichkeiten. Das Verfahren ist aber in der Praxis uninteressant, da zu viele Instanzen verworfen werden müssen.

Aufgabe 4 Neuronale Netze (18 Punkte)

Gegeben sei folgendes Neuronales Netz mit der Identität als Aktivierungsfunktion, d.h. $g(x) = x$.



$$\begin{aligned} W_{x,h} &= 0.3 & W_{h,a} &= -0.3 \\ W_{x,i} &= -0.1 & W_{i,a} &= -0.8 \\ W_{x,j} &= 0.2 & W_{j,a} &= -0.4 \\ W_{y,h} &= 0.7 & W_{h,b} &= 0.2 \\ W_{y,i} &= -0.4 & W_{i,b} &= 0.2 \\ W_{y,j} &= -0.6 & W_{j,b} &= 0.6 \end{aligned}$$

- 4-a Berechnen Sie die Outputs (a, b) für die Eingabe $x = 1$ und $y = -1$. Geben Sie auch alle relevanten Zwischenresultate an (z.B. die Aktivierung der Zwischenknoten).

5 Punkte

$$in_h = W_{x,h} \cdot x + W_{y,h} \cdot y = 0.3 - 0.7 = -0.4$$

$$in_i = -0.1 - (-0.4) = 0.3$$

$$in_j = 0.2 - (-0.6) = 0.8$$

Die identische Aktivierungsfunktion läßt gibt die Aktivierungswerte unverändert weiter, d.h. $out_x = in_x$.

$$in_a = W_{h,a} \cdot out_h + W_{i,a} \cdot out_i + W_{j,a} \cdot out_j = (-0.3) \cdot (-0.4) + (-0.8) \cdot 0.3 + (-0.4) \cdot 0.8 = -0.44$$

$$in_b = W_{h,b} \cdot out_h + W_{i,b} \cdot out_i + W_{j,b} \cdot out_j = 0.2 \cdot (-0.4) + 0.2 \cdot 0.3 + 0.6 \cdot 0.8 = 0.46$$

Die Ausgabewerte bleiben wiederum unverändert.

- 4-b Nehmen Sie nun an, dass das Netzwerk für obigen Input $(x, y) = (1, -1)$ die Ausgabe $(a, b) = (-0.2, 0.9)$ liefern soll. Die Lernrate sei $\alpha = 0.5$.

1. Berechnen Sie die Fehlerterme Δ_a und Δ_b

3 Punkte $g'(x) = 1$

$$\Delta_a = Err_a \cdot g'(in_a) = (-0.2 - (-0.44)) \cdot 1 = 0.24$$

$$\Delta_b = Err_b \cdot g'(in_b) = (0.9 - 0.46) \cdot 1 = 0.44$$

2. Berechnen Sie die Fehlerrate Δ_h

3 Punkte

$$\Delta_h = W_{h,a} \cdot \Delta_a \cdot g'(in_h) + W_{h,b} \cdot \Delta_b \cdot g'(in_h) = (-0.3) \cdot 0.24 + 0.2 \cdot 0.44 = -0.072 + 0.088 = 0.016$$

3. Berechnen Sie die Gewichtsänderung für das Gewicht $W_{h,a}$

3 Punkte

$$W_{h,a} \leftarrow W_{h,a} + \alpha \cdot \Delta_a \cdot out_h = -0.3 + 0.5 \cdot 0.24 \cdot (-0.4) = -0.348$$

- 4-c Angenommen, Sie können den Hidden Layer dieses Netzes beliebig vergrößern. Welche Art von Funktionen könnten Sie dann in den Outputs a und b zumindest lernen? Was ändert sich, wenn beliebige Aktivierungsfunktionen verwendet werden können?

4 Punkte

Nur lineare Funktionen. Ist jede beliebige Aktivierungsfunktion erlaubt, können alle stetigen Funktionen gelernt werden.

Aufgabe 5 Verschiedenes (24 Punkte)

5-a Was ist das sogenannte Frame-Problem, und wie wird es mit dem in der Vorlesung kennen gelernten Repräsentationsformalismus aus dem STRIPS-Planner gelöst?

4 Punkte Das Frame-Problem ist die Tatsache, daß man in einer logischen Repräsentation der Welt im Situationskalkül auch Regeln dafür definieren muß, daß die meisten Aktionen an den meisten Zuständen nichts ändern. Bei STRIPS ist das nicht mehr notwendig, da immer nur ein konkreter Weltzustand betrachtet wird, und nur mehr die durch Aktionen hervorgerufenen Änderungen (durch die Add- und Delete-Listen) repräsentiert werden müssen.

2 Punkte für Erklärung, die sich auf Zustände statt Zustandsänderungen konzentriert.

5-b Beantworten Sie folgende Fragen zu Graphplan:

1. Warum ist die Anzahl der Zustands-Literale von Schicht zu Schicht monoton steigend?

3 Punkte

Weil jedes einmal im Graphen vorhandene Literal durch die Persistenz-Aktionen in den folgenden Schichten erhalten bleibt.

2. Warum ist die Anzahl der Mutex-Verbindungen zwischen diesen Literalen fallend?

4 Punkte

Je mehr Literale vorhanden sind, desto mehr Aktionen werden möglich, und desto eher ist es möglich ein Paar von Literalen, die bisher einen Mutex-Link hatten, durch ein Aktionen-Paar zu erreichen, daß sich nicht gegenseitig ausschließt.

3. Was kann man aus diesen beiden Sachverhalten für den Planungsgraphen folgern?

3 Punkte

Dass der Planungsgraph endlich ist, und somit sein Aufbau konvergiert.

5-c Erklären Sie kurz die Idee des Temporal-Difference Learnings ohne dabei Formeln zu verwenden.

4 Punkte Die Grundidee des Temporal Difference Learning ist, daß man in jedem Lernschritt versucht, die Bewertung der momentanen Situation an die Bewertung der Situation einen (oder mehrere) Schritte später anzugleichen.

2 Punkte für Erklärung "Temporal Difference"

2 Punkte für Erklärung des Trainingssignals.

5-d Erklären Sie den Zusammenhang zwischen dem Chinese Room Argument und dem Turing Test. Wie, glauben Sie, beurteilt Searle den Turing Test?

Hinweis: Sie müssen weder den Turing Test noch das Chinese Room Argument erklären, für diese Erklärungen gibt es keine Punkte.

6 Punkte Das Szenario des Chinesischen Raumes ist dem Turing-Test nachempfunden. Searle möchte damit zeigen, daß, selbst wenn aus dem Raum perfekte Antworten kommen, d.h. das System den Turing-Test bestehen würde, es klar ist, daß nichts in dem Raum chinesisch versteht, d.h. daß hier keine "Intentionalität" vorhanden sein kann, die seiner Ansicht nach Intelligenz bedingt. Der Turing-Test ist daher, laut Searle, nicht zulässig.