

Grundlagen der Wissensverarbeitung-Übungsblatt 1

Übungsgruppe 2; Tom Kastek (4kastek@inf), Phil Sehlmeier (4sehlmeier@inf) · WiSe 17/18

1 Exercise 1.1: Search Space Properties

Fully observable vs partially observable:

- Voll beobachtbar ist eine Umgebung, in der zu einem Zeitpunkt ein Agent alles beobachten kann.
- Teilweise beobachtbar ist eine Umgebung, die sich beispielsweise verändert oder nicht voll einsehbar ist, da Teile des Systems vielleicht verdeckt sind. Um Aussagen über die Zeit machen zu können, müssen sich hierbei Zustände gemerkt werden.
- Vorteil der vollen Beobachtbarkeit ist, dass für ein bestimmtes Ziel eines Agenten im System direkt kalkuliert werden kann, was der optimale Weg für das Erreichen des Ziels im System ist.
- Während Schach ein Beispiel für ein voll beobachtbares System ist (jeder Zug kann berechnet werden), wäre ein Beispiel für ein teilweise beobachtbares System ein Kartenspiel. Der Agent kennt nur die Karten, die er selbst sieht, bzw. bei Einsatz eines Speichers für den Ablagestapel jene Karten, die dort liegen, jedoch kann er keine Aussage darüber treffen, welche Karten welcher andere Spieler auf der Hand hat oder noch im Deck liegen.

Discrete vs continuous

- Diskrete Umgebungen sind solche, in denen die Anzahl der Möglichkeiten eines Agenten zum Handeln begrenzt sind. Beispielsweise kann sich eine Schachfigur im Spiel nur an wenige bestimmte Punkte bewegen.
- Kontinuierliche Umgebungen sind solche, in denen die Anzahl der Möglichkeiten eines Agenten zum Handeln unendlich sind.
- Diskrete Umgebungen haben den Vorteil, dass sich eine optimale Möglichkeit der nächsten Aktion kalkulieren lässt. Hingegen haben kontinuierliche Umgebungen den Vorteil, dass eine größere Varianz an Möglichkeiten herrscht, zwischen denen sich ein Agent entscheiden kann.

Deterministic vs stochastic

- Deterministisch handeln Agenten, wenn Aktionen immer das selbe Ergebnis haben. Beispielsweise ist die Position eines Bauerns im Schach, wenn er zwei Stellen vorrückt, bei jedem Spiel dieselbe Stelle.
- Stochastisch handeln Agenten hingegen, wenn ihre Aktion zu einem zufälligem Ergebnis führt. Das Ergebnis eines Würfelwurfs ist beispielsweise nie voraussehbar.
- Deterministische Umgebungen haben den Vorteil, dass sie sich exakt berechnen lassen und sich ein gewünschtes Ergebnis dementsprechend genau kalkulieren lässt.
- Stochastische Umgebungen zu berechnen hat den Vorteil, die Flexibilität und Risiken der Umgebungen zu erkennen.

Grundlagen der Wissensverarbeitung-Übungsblatt 1

Übungsgruppe 2; Tom Kastek (4kastek@inf), Phil Sehlmeier (4sehlmey@inf) · WiSe 17/18

2 Exercise 1.2.1: Search Space 1

Es handelt sich im Beispiel einer App zur Planung einer optimalen Route im öffentlichen Nahverkehr um ein State-Space Problem. Als States haben wir die einzelnen Haltestellen, die wir erreichen können. Der Start State ist unsere Position, bzw. kann jeder State in unserer Umgebung sein, da wir überall starten könnten. An jeder Station kann eine Person abhängig von den vorbeifahrenden Bussen anliegende Haltestellen erreichen. Also bringt die Aktionsfunktionen eine Person von einer Station zur nächsten. Unser Zielstatus kann theoretisch jeder State in der Umgebung sein, aber es wird zu Beginn einer vom Agenten gewählt, der erreicht werden soll. In dieser Umgebung wird versucht, eine optimale Lösung zu finden. Dies wird getan, indem jeder Übergang eine Bewertung erhält. Diese Bewertung sollte gleich der benötigten Zeit sein, da eine Person voraussichtlich möglichst schnell an ihr Ziel gelangen möchte. Die Nodes in unserem Graphen sind also die einzelnen Haltestellen, die Edges die Verbindung zwischen zwei Haltestellen in Form einer Bus-/Bahnfahrt.

3 Exercise 1.2.2a

Unsere States in dieser Umgebung sind alle möglichen Füllungen der zwei Krüge. Krug K1 kann dabei bis zu vier Liter und Krug K2 bis zu drei Liter gefüllt werden. Da wir noch nicht wissen, ob es nicht erreichbare States gibt, gehen wir davon aus, dass alle Kombinationen ein State sind:

$K1 = 0L, K2 = 0L$

$K1 = 1L, K2 = 0L$

...

$K1 = 0L, K2 = 1L$

...

$K1 = 4L, K2 = 3L$

Damit ergeben sich insgesamt 20 mögliche Kombinationen (5 mögliche Füllungen für den 4-Liter-Krug, 4 für den 3-Liter-Krug) einer Füllung.

Unser Start State ist dabei $K1=0L, K2=0L$. Als Ziel ergibt sich ein Set aus States für alle $K1=2L, K2=?L$. Wir haben nun folgende möglichen Übergänge:

- Ein Krug kann zu jederzeit - wenn er nicht schon voll ist - komplett aufgefüllt werden, z.B. $K1=1L, K2=2L \rightarrow K1=4L, K2=2L$
- Ein Krug kann zu jederzeit mit seinem Inhalt komplett auf den anderen Krug füllen, bzw. soviel übergeben, bis der andere Krug voll ist. Z.B. $K1=3L, K2=2L \rightarrow K1=4L, K2=1L$
- Ein Krug kann seinen Inhalt jederzeit komplett verschütten. Z.B. $K1=3L, K2=2L \rightarrow K1=0L, K2=2L$

Grundlagen der Wissensverarbeitung-Übungsblatt 1

Übungsgruppe 2; Tom Kastek (4kastek@inf), Phil Sehlmeier (4sehlmeier@inf) · WiSe 17/18

Daraus ergibt sich, dass K1 z.B. folgendermaßen auf 2L gefüllt werden kann:

K1 = 0L, K2 = 0L | K2 füllen

K1 = 0L, K2 = 3L | K2 in K1 schütten

K1 = 3L, K2 = 0L | K2 füllen

K1 = 3L, K2 = 3L | K2 in K1 schütten

K1 = 4L, K2 = 2L | K1 verschütten

K1 = 0L, K2 = 2L | K2 in K1 schütten

K1 = 2L, K2 = 0L | Goal State

4 Exercise 1.2.2b

Mit teurem Wein darf im Beispiel aus a) nichts mehr verschüttet werden. Also gibt es nur noch die Übergänge des Auffüllens oder der Übergabe. Hiermit ist es nicht möglich, das Problem, in K1 2 Liter zu füllen, zu lösen.

- Maximal darf nur 5 Liter ausgegeben werden. dies folgt daraus, dass wenn K1 auf 2L gefüllt wurde, K2 maximal zu 3L gefüllt sein kann.
- Also muss nach dem ersten Füllen eine Übergabe stattfinden.
- Wenn also K1 zuerst befüllt wird, kann nur 3L an K2 übergeben werden. Nun kann nicht nachgefüllt werden, also könnte K2 nur zurück an K1 kippen, was eine Schleife ergäbe.
- Wenn K2 zuerst befüllt wird, können wir die 3L nur komplett übergeben. Nun kann K2 nicht voll nachgefüllt werden. Also können wir nur K1 voll füllen, was den bereits beschriebenen Zustand ergäbe K1=4L, K2=0L.