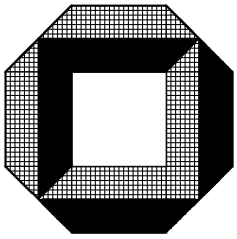
Proseminar

Künstliche Intelligenz



Universität Karlsruhe (TH)
Fakultät für Informatik
Institut für Algorithmen und Kognitive Systeme

Prof. Dr. J. Calmet
Dipl.-Inform. A. Daemi

Wintersemester 2003/2004

Institut für Algorithmen und Kognitive Systeme
Fakultät für Informatik
Universität Karlsruhe
Am Fasanengarten 5
76 128 Karlsruhe

Entscheidungsfindung

Vinh Phuc Dinh
Uta Hellinger

Inhaltsverzeichnis

4 Entscheidungsfindung **2**

4.1 Einleitung 2

4.2 Einfache Entscheidungsfindung 2

4.2.1 Nutzentheorie 2

4.2.2 Entscheidungstheoretische Agenten 8

4.3 Komplexe Entscheidungsfindung 10

4.3.1 Sequentielle Entscheidungsprobleme 10

4.3.2 Optimale Lösungen sequentieller Entscheidungsprobleme . 13

4.4 Zusammenfassung 20

Kapitel 4

Entscheidungsfindung

4.1 Einleitung

In den letzen Kapiteln haben wir gelernt, was ein Agent ist und wie er Probleme unter verschiedenen Umständen der Unwissenheit identifizieren kann.
In diesem Kapitel soll es nun darum gehen, aus der Vielfalt an Entscheidungsmöglichkeiten die Beste herauszusuchen.
Hierfür werden wir die Nutzentheorie einführen um mit ihr eine Ordnungsrelation auf den Entscheidungsmöglichkeiten zu definieren.

4.2 Einfache Entscheidungsfindung

Die Handlungsmöglichkeiten des Agenten sind eng mit dem zeitlichen Verlauf gekoppelt. Eine Chance, die zu einem Zeitpunkt noch vorhanden war, kann in der nächsten Zeitperiode schon vergangen sein.
In diesem Teilabschnitt wollen wir uns auf eine kleine Zeitperiode mit konstanten Zufallsvariablen und Entscheidungsmöglichkeiten beschränken.
Diese lokale Sicht erweitern wir in der zweiten Hälfte der Ausarbeitung auf beliebig viele Zeitintervalle.

4.2.1 Nutzentheorie

In der Einleitung sagten wir, unser Ziel wäre es, die zu einem Zeitpunkt „beste“ Entscheidung zu treffen. Aber was bedeutet das genau? Wie definieren wir „beste“? Der Duden schreibt hierzu :

> best... : In höchstem Maße od. Grade gut; so gut wie irgend möglich...

Unter gut findet man schließlich:

> gut : bestimmten Ansprüchen, Zwecken genügend

Der ersten Definition entnehmen wir, dass es sich um ein Optimierungsproblem (Maximierungsproblem) handelt. Die zweite Definition verrät uns, dass wir für die Beurteilung von Entscheidungen einen Anspruch oder Zweck definieren müssen.
Für Menschen ist bei moralischen Entscheidungen das Wertesystem der Regulator. Ob eine Bank Aktien einkauft oder sie verkauft berechnet der Computer mit Hilfe einiger Formeln, bei Google entscheiden ein Algorithmus und 6000 Server wie Suchergebnisse bewertet werden.
In allen Beispielen erkennen wir den systematischen Aufbau von...

1. Handlungsmöglichkeiten erkennen
2. Möglichkeiten bewerten
3. Optimale Entscheidung bzgl. unseres Maßstabes bestimmen (Algorithmus, BWL-Formeln, Gewissen...)
4. Aktion mit höchstem Nutzen ausführen(kaufen/verkaufen, Ranking erstellen...)

Die Nutzentheorie gibt uns eine mathematische Präzisierung der o.g. „Wertesysteme“.
Sie ist eine Abbildung $U : \mathit{Reale\ Welt} \to \mathbb{R}$, welche jeder Handlungsmöglichkeit ihren Nutzen zuweist. Das Ziel der Entscheidungsfindung ist die Maximierung von U unter Berücksichtigung aller zur Verfügung stehenden Möglichkeiten.

Ist der Wert des Nutzens von Bedeutung, so sprechen wir vom **kardinalen Nutzenprinzip**. Ist hingegen lediglich die Position relativ zu einem anderen Punkt von Bedeutung („x über y $\vee$ x unter y“), so spricht man vom **ordinalen Nutzenprinzip**.

Ein Beispiel für das kardinale Nutzenprinzip wäre Besitz und Vermögen. Nehmen wir an wir wären arm und hätten überhaupt kein Geld. Wenn ein Fremder uns ein Euro schenkt freuen wir uns, denn $1 EUR$ besitzen ist sicherlich besser als nichts zu besitzen. Würde er uns eine Millionen Euro schenken, wäre das aber noch besser als der Besitz des einen Euros. Die Differenz zum Ursprungswert ist relevant für die Beurteilung.

Ein Beispiel für das ordinale Nutzenprinzip wäre das Glücksgefühl. Wir können zu einem Zeitpunkt behaupten dass wir mit dem Zustand glücklich sind. Wir können sogar zwei Zeitpunkte daraufhin vergleichen. Aber es ist unmöglich Glück auf einer Skala zu messen. Wie sollten wir eine „Glückseinheit“ definieren? Legten wir die Skala fest, was würde dagegen sprechen diese Skala einfach zu verdoppeln, oder zu quadrieren...

Wir wollen folgende Notation einführen...

$A \succ B$	A wird gegenüber B streng bevorzugt
$A \sim B$	A ist gegenüber B indifferent
$A \succsim B$	A ist gegenüber B bevorzugt oder A und B sind indifferent

Unter einem Zustand A versteht man eine Wertebelegung aller im Modell vorkommenden Variablen. Die Erweiterung durch Wahrscheinlichkeiten nennt man Lotterie.

$$L = [p_1, C_1; p_2, C_2; ....p_n, C_n]$$

Hierbei bezeichnet p_i die Wahrscheinlichkeit, dass das Ereignis C_i eintritt.

Einen ceteris paribus[1] Zustand kann man als eine Handlung interpretieren.

Axiome der Nutzentheorie

- $A \succsim A$ (Reflexivität)
- $\forall A, B : A \succsim B \vee B \succsim A$ (Totalität)
- $A \succ B \wedge B \succ C \Rightarrow A \succ C$(Transitivität)
- $A \succ B \succ C \Rightarrow \exists p[p, A; (1-p), C] \sim B$ (Kontinuität)
  Wenn ein Zustand B ex., das zwischen A und C liegt, so ex. ein Wahrscheinlichkeitslos mit A und C, sodass das Los zu B indifferent ist.
- $A \sim B \Rightarrow [p, A; (1-p), C] \sim [p, B; 1-p, C]$ (Substituierbarkeit)
  Wenn ein Agent zwischen 2 Losen indiff. ist, dann ist er auch zw. 2 komplexeren Losen indiff., welche sich nur in A und B unterscheiden.
- $A \succ B \Rightarrow (p \succsim q \Leftrightarrow [p, A; (1-p), B] \succsim [q, A; (1-q), B])$ (Monotonie)
  Wenn A gegenüber B bevorzugt wird, so ist auch ein Los mit einer höheren Wahrscheinlichkeit für A besser.
- $[p, A; (1-p), [q, B; (1-q), C]] \sim [p, A; q(1-p)B; (1-p)(1-q), C]$ (Dekomposabilität)
  Lose können nach den Regeln der Wahrscheinlichkeit zusammengefasst werden. (z.B. Pfadregel)

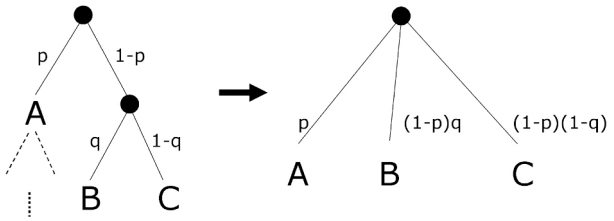


Abbildung 4.1: Dekomposabilität (Ersetzbarkeit) mittels Pfadregel

- Rationalprinzip
  Der Agent wählt immer die zu einem Zeitpunkt bestmögliche Option.

[1] Variation einer Variablen bei Konstanz der Restlichen

Die Nutzentheorie hat zum Ziel, das Verhalten(Aktion) und deren Auswirkungen zu bewerten. Alle mit den Axiomen syntaktisch ableitbaren Handlungen müssen sich mit der Realität decken (also semantisch ableitbar sein). Sie muß somit ein korrektes Modell darstellen (siehe Goos, S.13 : „Modell und Wirklichkeit“). Den Sachverhalt möchten wir hier am Beispiel der Transitivität demonstrieren.

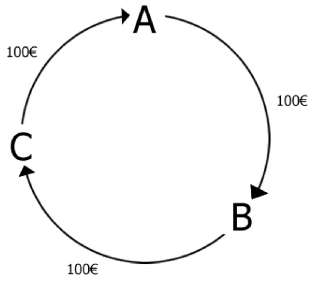


Abbildung 4.2: Bei Wegfall der Regeln wird das Modell falsch

Nehmen wir an die Transitivität sei kein zwingendes Axiom. Es sei also $A \succ B \succ C \succ A$. Der Einfachheit wegen nehmen wir an, es handle sich bei A, B und C um Konsumgüter. Ferner sei unsere Anfangsausstattung zum Zeitpunkt t o.B.d.A. das Konsumgut A. Da wir A gegenüber B präferieren, sind wir sicherlich bereit, B zu konsumieren, wenn wir dafür als Entschädigung ein wenig Geld bekommen würden, sagen wir 100EUR. Im folgenden Schritt tauschen wir B gegen C aus und erhalten nochmals 100EUR. Und schließlich geben wir auch C, das wir laut Vorraussetzung gegenüber A bevorzugen, auf, um weitere 100EUR zu kassieren. Nach drei Entscheidungen sind wir wieder bei A angelangt und haben 300EUR in der Tasche.

Mehrwertige Nutzentheorie

Es kann vorkommen, dass eine Nutzenfunktion mit nur einem Funktionswert die Wirklichkeit nicht präzise genug beschreiben kann. Das ist der Fall wenn viele unabhängige Kriterien bei der Entscheidungsfindung zusammen kommen. Man greift dann auf Funktionen zurück, die als Ergebnis anstatt einer Zahl einen Vektor liefern. Die Vektoren werden dann komponentenweise miteinander verglichen. Sind z.B. $A = (x_1, x_2, x_3)$ und $B = (y_1, y_2, y_3)$ und ist $x_i > y_i \forall 0 < i < 3$, dann wird A gegenüber B präferiert. Natürlich wird eine so klare Trennung mit steigender Attributmenge selten vorkommen. Die Realtität wird sehr viel komplexer sein und Überschneidungen werden auftreten (d.h. nicht alle x_i sind größer als y_i). Aber die Grundidee der Nutzentheorie bleibt bei der *mehrwertigen Nutzentheorie* erhalten, lediglich die mathematischen Zusammenhänge werden durch stochastische Methoden erschwert. Die Behandlung des Themas würde den Rahmen dieser Ausarbeitung sprengen, so dass der interessierte Leser auf weiterführende Literatur verwiesen wird (siehe z.B. Keeney).

Definition und NP-Vollständigkeit

Mit der Nutzentheorie sind wir nun befähigt, unser eigentliches Maximierungsproblem zu präzisieren:

$$MEU(A|E) = max(\sum_i P(Result_i(A)|Do(A), E) * U(Result_i(A)))$$

Der **maximal erwartete Nutzen**[2] ergibt sich als Produkt aus dem Nutzen eines Ereignisses und der Wahrscheinlichkeit, dass das Resultat des Ereignises bei Ausführung eintritt.
Genau genommen haben wir mit dieser Definition das Ziel der KI (und der Menschen) zusammengefasst. Könnten wir zu jedem Zeitpunkt *alle* Möglichkeiten erkennen und hätten wir eine adequate Nutzenfunktion, so könnten wir unser Vorhaben beliebig genau in die Zukunft „hineinplanen“, und gemäß der Nutzenfunktion optimieren.
Leider wächst der Entscheidungsbaum offensichtlich exponentiell an und kann somit nicht mehr polynomiell gelöst werden. Bei den meisten Anwendungen zeigt eine Option ferner erst in Kombinationen mit anderen Entscheidungen ihren „wahren“ Nutzen. So ist ein Husten an sich keine gefährliche Krankeit. Die Kombination von Husten und AIDS hingegen kann schnell zum Tod führen.

Menschliche Entscheidungen und deren Fehlbarkeit

Am Anfang des Kapitels setzten wir den Menschen als Maßstab für die Handlung eines Agenten. Wir nahmen axiomatisch an, dass ein Mensch stets rational und zu seinem Besten handeln würde. Dass dies oftmals nicht der Fall ist, zeigt der folgende Versuch der Psychologen Tversky und Kahneman vom Jahr 1982: Probanden wurden in zwei aufeinander folgenden Schritten je zwei verschiedene Lose angeboten unter denen sie jeweils eine aussuchen sollten.
Im ersten Schritt hatten sie zwischen Los A und Los B, dann im folgenden Schritt zwischen Los C und Los D zu entscheiden.

A : 80% Gewinn 4000EUR	C : 20% Gewinn 4000EUR
B : 100% Gewinn 3000EUR	D : 25% Gewinn 3000EUR

Die Mehrheit der Probanden wählten im ersten Fall das Los B und im zweiten Schritt das Los C. Nehmen wir an $U(0EUR) = 0$. So ist aus der ersten Entscheidung zu entnehmen : $U(3000EUR) \succ 0,8U(4000EUR)$. Analog hierzu im zweiten Fall : $0,2U(4000EUR) \succ 0,25U(3000EUR)$.Löst man die letztere Gleichung nach U(3000EUR) auf, erhält man $U(3000EUR) \prec 0,8U(4000EUR)$, was ein Widerspruch zur ersten Entscheidung darstellt.
Was ist falsch gelaufen? Mathematisch betrachtet haben die Probanden offensichtlich „falsch“ entschieden. Nutzentheoretisch wurde ihnen zwei gleiche Allokationen offeriert, sie entschieden sich beide Male aber unterschiedlich.[3]

[2] MEU - max. expected utility

[3] In der Volkswirtschaftslehre auch bekannt unter dem „Prinzip der offenbarten Präferenzen“

Bei nährerer Betrachtung erschließt sich der Fehler jedoch in einer anderen Quelle : Der Nutzenfunktion. Wir haben das *Prinzip des Bedauerns (regret)* nicht berücksichtigt! Es besagt, dass ein Mensch ein sicheres Los gegenüber einem Unsicheren bevorzugt, da das Bedauern über die verloren gegangene Chance sonst zu hoch wäre. Natürlich ist das Bedauern noch von vielen anderen Parametern wie der Höhe der Lose, dem Einkommen des Probanden, dem Charakter (risikoscheu od. risiko avers) usw. ... abhängig.
Wir sehen an dem Beispiel, dass unsere Modellierung von hoher Bedeutung für die Entscheidungsfindung ist! Eine falsche Modellierung kann mathematisch korrekte Ergebnisse liefern, aber die Ergebnisse werden sich nicht mit der Realität decken.

Was ist Wissen wert?

Wir sind bisher davon ausgegangen dass alle benötigten Informationen uns zum Zeitpunkt der Entscheidungsfindung zur freien Verfügung stehen. Dass das nicht realistisch ist sehen wir daran, dass in einem Unternehmen das Wissen über Produktionsprozesse und Technologien das eigentliche Kapital darstellt und als Firmengeheimnis streng bewacht wird. Gerade in unserer Informationsgesellschaft entscheidet die Information mehr denn je über den Erfolg oder Misserfolg eines Unterfangens.
Die Theorie über den Wert einer Information untersucht die Höhe der Zahlungsbereitschaft eines Entscheiders für eine zusätzliche, ihm sonst verborgen gebliebene Information.

Da wir mit Nutzenmaximierern zu tun haben können wir den Wert einer Information so definieren:

Wert der Information

Eine Information ist genau so viel wert, wie der durch sie *erwartete* erhöhte Nutzen. Formal:

Der erwartete Nutzen ohne die Information ist bekanntlich:

$$EU(\alpha|E) = \max_A \sum_i U(Result_i(A))P(Result_i(A)|Do(A), E)$$

Und mit Information ist der erwartete Nutzen:

$$EU(\alpha_{E_j}|E, E_j) = \max_A \sum_i U(Result_i(A))P(Result_i(A)|Do(A), E, E_j)$$

Da E_j eine Zufallsvariable (mit unbekanntem Anfangswert) ist, müssen wir über alle möglichen Werte e_{jk} für E_j mitteln und den Wert der Information schätzen.

Wir gehen von einer perfekten Information aus. Eine Information, welche mit einer Wahrscheinlichkeit <1 eintritt, können wir als perfekte Information simulieren, welche mit einem zusätzlichen Zufallsknoten verbunden ist.
Der *Wert der Information* (VPI=Value of Perfect Information) ergibt sich zu

$$VPI_E(E_j) = (\sum_k P(E_j = e_{jk}|E)EU(\alpha_{e_{jk}}|E, E_j = e_{jk})) - EU(\alpha|E)$$

Zuletzt sei noch angemerkt, dass VPI natürlich nie negativ sein kann. Eine zusätzliche Information kann nicht schaden weil wir Sie ignorieren können, sofern sie uns nicht weiterhilft.

Sehen wir uns zur Veranschaulichung ein Beispiel an:

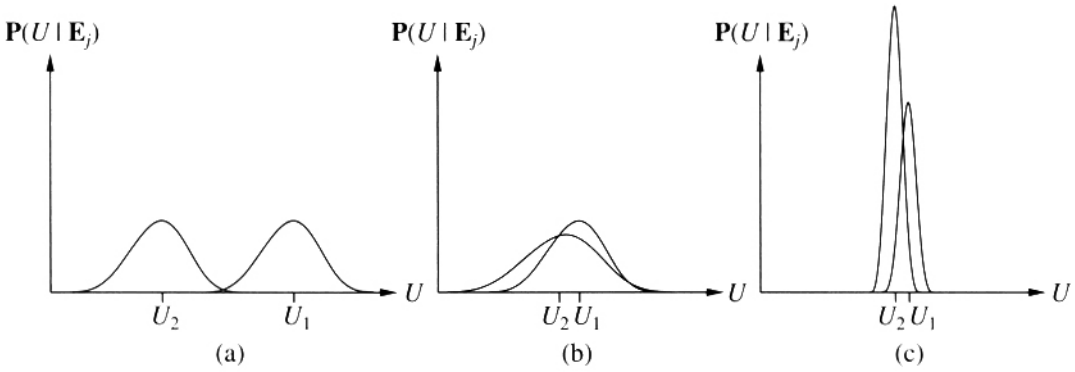


Abbildung 4.3: Bsp.: VPI in verschiedenen Situationen

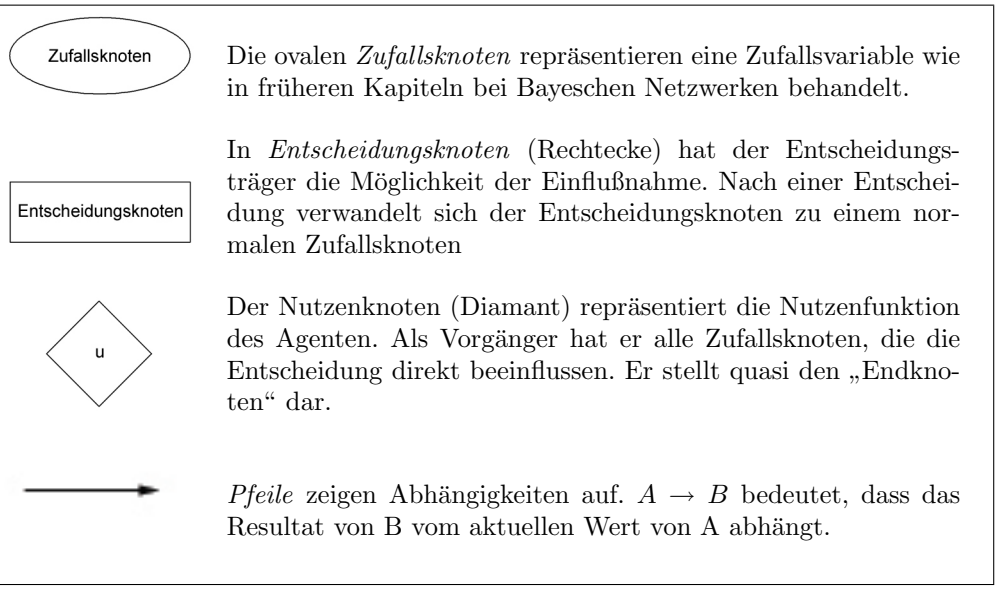
Seien A_1 und A_2 zwei verschiedene Routen durch ein Gebirge im Winter. A_1 ist ein Pass mit wenig Verkehr, A_2 eine windige und verdreckte Straße. Mit diesen Informationen werden wir A_1 klar gegenüber A_2 bevorzugen. Die zusätzliche Information müsste sehr viele neue Details beinhalten da unsere Situation schon so klar ist.[4]
Nehmen wir nun an A_1 und A_2 wären zwei in etwa gleich lange und schlechte Strassen, kurvig und voller Schlaglöcher. Die Wahrscheinlichkeit, dass wir durch zusätzliche Information in Form von Satellitenbildern etwas verbessern können, ist eher gering. Wir werden nicht so viel für die Information zahlen wollen. Erweitern wir die Situtation dadurch, dass wir nun ein Krankenwagen fahren und einen Schwerverletzten transportieren. Ein Stau auf eines der Strassen wäre fatal, auch wenn die Wahrscheinlichkeit hierfür noch so klein ist. Die Satellitenbilder sind in dieser Situation sicherlich sehr viel mehr wert als im vorherigen Beispiel. Wir schätzen eine kleine Verbesserung Δt(Zeit zum Zielort) als hoch ein.

4.2.2 Entscheidungstheoretische Agenten

Entscheidungsnetzwerke - eine Darstellungsform

Entscheidungsnetzwerke sind eine graphische Darstellung für Entscheidungsmöglichkeiten. Sie verbinden Bayesche Netzwerke mit zusätzlichen Aktions- und Nutzenknoten.

[4] Information ist nur dann was wert, wenn sie unsere Entscheidung zum Positiven hin verändert



Die Abbildung 5.3 zeigt ein typisches Entscheidungsnetzwerk. Ihr entnimmt man zum Beispiel, dass die pathologische Erweiterung der Arterie (Aortic Aneurysm) direkt von der gewählten Behandlung (Treatment) abhängt. Ob und mit welcher Wahrscheinlichkeit ein Patient an einer solchen Gefäßerweiterung leidet, entscheidet über Leben und Tod (Nutzen). Deshalb ist der „Aortic Aneurysm“-Knoten mit dem Nutzenknoten direkt verbunden.
Die mit den Zufallsknoten direkt verbundenen Zufallsknoten sind keine „echten“ Zufallsvariablen sonder fassen die vorrangehenden Zufallsvariablen lediglich zusammen. Man kann sie deshalb auch weglassen und die restlichen Zufallsknoten direkt mit dem Nutzenknoten verbinden. Eine solche Tabelle nennen wir *Aktion-Nutzen Tabelle* da der Nutzen hier direkt abgeleitet wird.
Setzen wir alle Entscheidungsknoten auf einen festen Wert, erhalten wir ein Bayesches Netz und können den Nutzen mittels eines solchen Entscheidungsnetzwerkes direkt durchgerechnen.

Aufbau eines Expertensystems

Hier soll in verkürzter Form der Aufbau eines Expertensystems anhand eines Krankheit-Diagnose-Systems erörtert werden.

1. Mindestens ein Domänenexperte und ein Wissensexperte besprechen die benötigte Funktionalität (in unserem Fall ein Facharzt und ein Informatiker)
2. Beide stellen gemeinsam ein Ursache-Wirkungsmodell auf (Symptome $\Rightarrow$ Krankheit)
3. Das Modell wird auf ein Entscheidungsmodell vereinfacht indem irrelevante Daten entfernt oder mit anderen Daten verschmolzen werden. Wichtige Daten, welche in der Literatur zusammen vorkommen, können unter Umständen ihrer Wichtigkeit nach getrennt werden (In med. Lehrbüchern

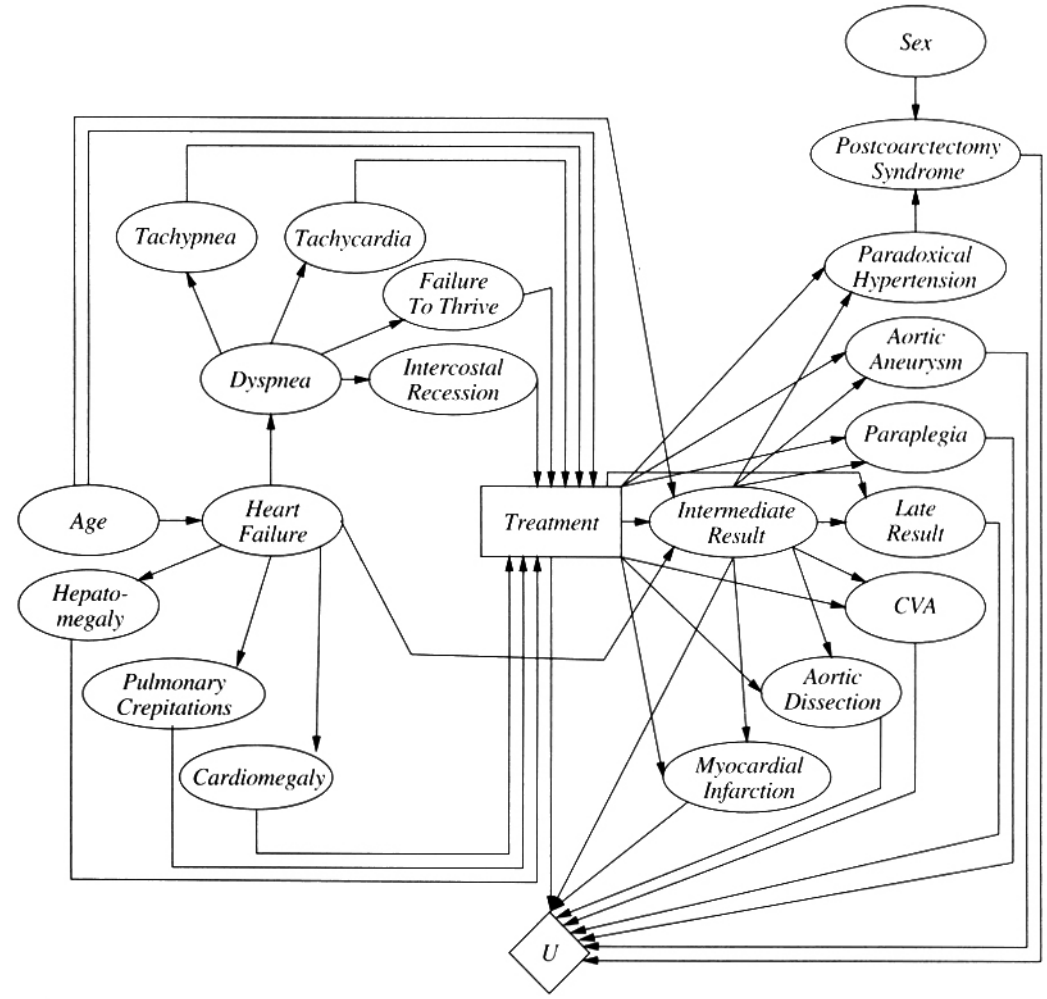


Abbildung 4.4: Beispiel eines Einflußdiagramms anhand einer arteriellen Dysfunktion ([1], S. 606

stehen oftmals auch Behandlungsmethoden. Die werden zur Diagnose nicht benötigt und können entfernt werden.)

4. Den Symptomen werden Wahrscheinlichkeiten und Nutzen zugeordnet (BayesŔegel, Literatur...)

5. Das Modell wird anhand einer *Rückwärtsanalyse* (Diagnose wird vorgegeben, daraus soll das System die Symptome berechnen. Diese werden dann mit der Fachliteratur verglichen) und eines *Sensitivitätschecks* (Eine kleine Veränderung der Nutzen und Wahrscheinlichkeitswerte sollten zu gleichen Ergebnissen führen) verifiziert und verfeinert.

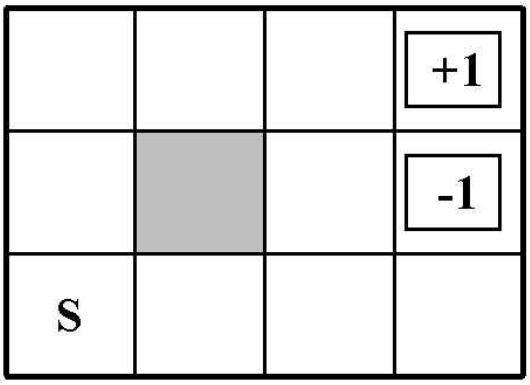


Abbildung 4.5: Abb.1: Die 3×4 Welt

4.3 Komplexe Entscheidungsfindung

Bisher wurden nur Probleme behandelt, bei denen es darum ging, eine einzelne Entscheidung zu treffen. Dieser Abschnitt soll Probleme behandeln, bei denen eine einzelne Entscheidung noch nicht zum Ziel führt, bei denen also eine ganze Reihe von Entscheidungen notwendig sind.

4.3.1 Sequentielle Entscheidungsprobleme

Sequentielle Entscheidungsprobleme erhält man in der Regel, wenn eine gewählte Aktion nicht sicher das gewünschte Ergebnis bringt. Der Agent muss dann in der Lage sein, solche unerwünschten Resultate zu korrigieren, d.h. er muss die für diesen Fall günstige Lösung wählen.

Was sind sequentielle Entscheidungsprobleme?

Am einfachsten wird der Charakter sequentieller Entscheidungsprobleme an einem Beispiel deutlich:

Als Umgebung sei eine 3×4 Welt wie in Abb.1 gegeben. Ein Agent soll in dieser Umgebung vom Startzustand S in einen der Endzustände, die mit "+1" und "-1" gekennzeichnet sind, gelangen. Dabei koste jeder Schritt 0.04. Das Erreichen des Endzustands "-1" koste 1, das Erreichen des Zustands "+1" bringe ein Entgelt von 1. Dieses Problem ist mit Mitteln aus dem letzten Abschnitt ohne weiteres lösbar. Hinzu kommt jetzt aber eine Zufallsvariable. Wenn der Agent in eine bestimmte Richtung laufen will, geht er nur mit 80%iger Wahrscheinlichkeit tatsächlich in diese Richtung. Mit je 10%iger Wahrscheinlichkeit geht er 90° weiter nach links bzw. rechts (vgl. Abb.2). Um nun in einen der beiden Endzustände zu kommen, reicht eine einmalige Entscheidung nicht aus, wie man sich leicht klar machen kann: Trifft der Agent am Anfang die Entscheidung, den Weg [hoch, hoch, rechts, rechts, rechts] zu wählen, so befindet er sich anschließend mit einer Wahrscheinlichkeit von $0.8^5 = 0.32768$, also nicht

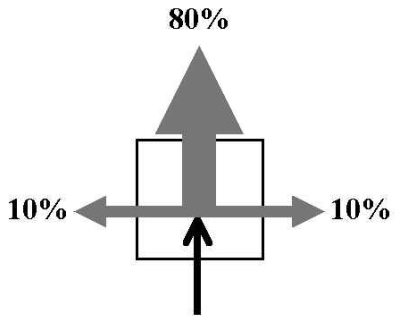


Abbildung 4.6: So bewegt sich der Agent

einmal einem Drittel tatsächlich im Zustand "+1". Diese Wahrscheinlichkeit ist so gering, dass es effektiver ist, wenn der Agent nach jedem Schritt neu entscheidet, in welche Richtung er weitergeht. Es ist zur Lösung dieses Problems also eine Reihe von Entscheidungen nötig. Deshalb nennt man solche Probleme auch sequentielle Entscheidungsprobleme.

Diese Art von Problemen lässt sich formal als Markoventscheidungsprozess (MDP) darstellen, wenn wir folgende Annahmen treffen:

1. Die Umgebung ist total überschaubar, d.h. der Agent weiß jederzeit, in welchem Zustand er sich befindet,
2. Die Kosten/Entgelte für die einzelnen Zustände addieren sich, wenn man sie nacheinander erreicht.

Der MDP besteht dann aus folgenden drei Komponenten:

- einem Anfangszustand z_0,
- einem Übergangsmodell T, wobei $T(z, a, z')$ die Wahrscheinlichkeit bezeichnet, dass der Agent durch Ausführen der Aktion a in den Zustand z' gelangt und
- einer Kostenfunktion $R(z)$

Einige Definitionen erlauben auch, dass die Kostenfunktion auch von der ausgeführten Aktion und dem Folgezustand abhängt; für diesen Fall erhält man $R(z, a, z')$.

Lösungen sequentieller Entscheidungsprobleme

Wie kann eine Lösung dieses Problems definiert werden? Da der Agent nach jedem Schritt eine neue Entscheidung treffen soll, wo er als nächstes hingeht, erhält man ein Entscheidungsproblem für jeden Zustand, den der Agent erreichen kann. Man entscheidet daher für jeden Zustand, welche Aktion der Agent dort wählen soll.

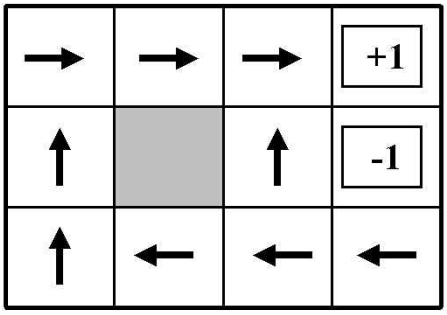


Abbildung 4.7: Mögliche Lösung für das Beispiel

Eine Lösung für obiges Beispiel wird also definiert, indem für jedes Feld festgelegt wird, in welche Richtung der Agent von dort aus gehen soll. Eine mögliche Lösung ist in Abb.3 zu sehen. Eine solche Lösung nennt man Taktik, üblicherweise mit π bezeichnet. $\pi(z)$ bezeichnet die Aktion, die im Zustand z gewählt wird. Ein Problem wird mit Hilfe der Taktik gelöst, indem in jedem Schritt die Aktion $\pi(z)$ mit z als aktuellem Zustand gewählt wird.

4.3.2 Optimale Lösungen sequentieller Entscheidungsprobleme

Im letzten Abschnitt haben wir gesehen, was sequentielle Entscheidungsprobleme sind und wie Lösungen dieser Probleme aussehen. In diesem Abschnitt sollen möglichst "gute" Lösungen bestimmt werden. Zunächst wird erläutert, wie man die Güte einer Lösung bestimmt und dann werden Methoden vorgestellt, um optimale Lösungen zu bestimmen.

Klassifizierung von MDPs

In diesem Abschnitt werden die sequentiellen Entscheidungsprobleme klassifiziert. In den weiteren Ausführungen werden dann nur noch einzelne Klassen betrachtet, nicht mehr alle MDPs.

Zunächst unterscheidet man zwischen Problemen mit sogenanntem endlichem und Problemen mit unendlichem Horizont. Endlicher Horizont bedeutet, dass der Agent nur eine endliche Zahl von Schritten zur Verfügung hat. Es ist leicht einzusehen, dass dann die Entscheidungen, die der Agent fällt, auch von der verbleibenden Zahl der Schritte abhängen. Um dies deutlich zu machen, verwenden wir nochmal das Beispiel aus 4.3.1. Der Agent befinde sich im Zustand (3,1). Wenn er nur noch 3 Schritte zur Verfügung hat, wird er nach oben gehen, da dies seine einzige Chance ist, noch in den Endzustand "+1" zu kommen. Wenn er noch 100 Schritte zur Verfügung hat, wird er dagegen den Weg über (1,1) nehmen, da er dabei nicht das Risiko eingeht, in den Endzustand "-1" zu gelangen. Da hier die gewählte Aktion nicht nur vom aktuellen Zustand, sondern auch von der verbleibenden Zeit abhängt, nennt man Taktiken für solche

Probleme variabel. Bei unendlichem Horizont ist es egal, ob der Agent im 100. Schritt oder im 1000. Schritt in einen Zustand gerät, da das Problem sich im Lauf der Zeit nicht ändert. In diesem Fall hängt die Entscheidung also nur vom Zustand ab, man spricht von einer statischen Taktik. Diese Taktiken sind einfacher zu handhaben als variable, da man weniger Zustände betrachten muss. Daher wollen wir uns auf Probleme mit unendlichem Horizont beschränken. Für Ausführungen über Probleme mit endlichem Horizont sei auf weiterführende Literstur verwiesen.

Ein weiterer Unterschied zwischen sequentiellen Entscheidungsproblemen ist, wie man Zustandsfolgen einen Nutzen zuordnet. Man unterscheidet prinzipiell zwischen zwei Möglichkeiten, den addierten und den diskontierten Kosten. Der Unterschied liegt darin, wie viel Einfluss mögliche Zustände in ferner Zukunft auf die Entscheidung haben.

Bei addierten Kosten werden die Kosten für jeden in Zukunft erreichten Zustand aufaddiert. Für eine Zustandsfolge $[z_0, z_1, z_2, \ldots]$ berechnet sich der Nutzen dann als $U_h([z_0, z_1, z_2, \ldots]) = R(z_0) + R(z_1) + R(z_2) + \ldots$

Wesentlich zweckmäßiger ist der Umgang mit diskontierten Kosten, bei denen in ferner Zukunft erreichte Zustände weniger beachtet werden als Zustände in naher Zukunft. Hier berechnet sich der Nutzen der Zustandsfolge $[z_0, z_1, z_2, \ldots]$ mit $U_h([z_0, z_1, z_2, \ldots]) = R(z_0) + \gamma \cdot R(z_1) + \gamma^2 \cdot R(z_2) + \ldots$ mit einem Diskontierungsfaktor γ zwischen 0 und 1. Je kleiner γ, desto weniger Einfluss haben Zustände in ferner Zukunft auf den Nutzen einer Folge von Zuständen. Für $\gamma = 1$ erhält man die addierten Kosten, bei denen jeder Zustand, der in Zukunft erreicht wird, den gleichen Einfluss hat.

Eine letzte Annahme, um den Umgang mit Taktiken zu erleichtern, ist, dass wir davon ausgehen, dass, falls der Agent in Zustand z_0 die Zustandsfolge $[z_0, z_1, z_2, \ldots]$ der Zustandsfolge $[z_0, z'_1, z'_2, \ldots]$ vorzieht, auch im nächsten Schritt die erste der zweiten Zustandsfolge vorzieht, d.h. er zieht $[z_1, z_2, \ldots]$ der Folge $[z'_1, z'_2, \ldots]$ vor.

Optimale Lösungen eines MDP

Die Qualität einer Taktik wird durch den zu erwartenden Nutzen bestimmt, den man bei deren Anwendung erhält. Eine optimale Taktik π^* ist also eine Taktik, die einen maximalen Nutzen erwarten lässt. Es ist leicht einzusehen, dass die optimale Taktik stark mit der Kostenfunktion $R(z)$ zusammenhängt. Falls der Agent in obigem Beispiel bei jedem Schritt ein Entgelt erhalten würde, würde er jedem Endzustand aus dem Weg gehen, da sein Nutzen in jedem Fall größer wird, solange er sich nicht in einem Endzustand befindet. Eine optimale Taktik für dieses Problem ist in Abb.4 zu sehen.

Bei der Bestimmung des Erwartungswertes des Nutzens einer Taktik bereitet der unendliche Horizont Probleme, da hier beim Bestimmen des Nutzens einer Zustandsfolge unendliche Summen auftreten können. Falls durch die Taktik garantiert ist, dass irgendwann ein Endzustand erreicht wird, so spricht man von einer anständigen Taktik. Wenn man mit einer nicht anständigen Taktik arbeitet, garantieren diskontierte Kosten, dass alle Summen beschränkt sind:

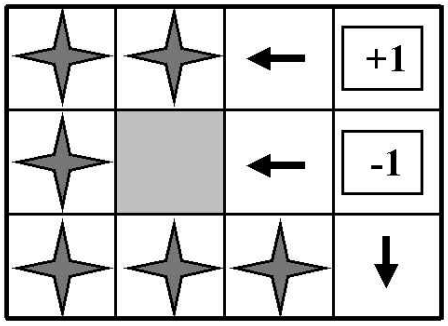


Abbildung 4.8: Lösung des Beislpiels, falls der Agent für jeden Schritt eine Belohnung erhält

$$U_h([z_0, z_1, z_2, \ldots]) = \sum_{t=0}^{\infty} \gamma^t R(z_t) \leq \sum_{t=0}^{\infty} \gamma^t R_{\max} = \frac{R_{\max}}{1-\gamma}$$

Im folgenden soll ein Algorithmus vorgestellt werden, mit dessen Hilfe eine optimale Lösung bestimmt werden kann.

Bestimmung optimaler Lösungen mit Hilfe von Wertiteration

Die Grundidee des hier vorgestellten Algorithmus ist, für jeden Zustand den Nutzen zu bestimmen und aus diesem dann die in jedem Zustand optimale Aktion zu wählen.

Die Bellman-Gleichungen Der Nutzen eines Zustands lässt sich in etwa so beschreiben, dass er dem erwarteten Nutzen der möglichen auf ihn folgenden Zustandsfolgen entspricht. Man bestimmt also den Nutzen eines Zustands dadurch, dass man einen Erwartungswert dafür berechnet, wieviel Kosten in Zukunft noch entstehen werden. Diese Zustandsfolgen hängen offensichtlich von der verwendeten Taktik ab, daher soll zunächst der Nutzen $U_p(z)$ eines Zustands bei Anwendung einer bestimmten Taktik bestimmt werden. Dieser ist:

$$U^{\pi}(z) = E\left[\sum_{t=0}^{\infty} \gamma^t R(z_t) \mid \pi, z_0 = z\right]$$

Mit dieser Definition ist der Nutzen eines Zustands $U(z) = U_{\pi^*}(z)$, wobei π^* die optimale Taktik ist.

Bemerkung: $U(z)$ und $R(z)$ haben verschiedene Bedeutungen: $R(z)$ drückt aus, welchen Gewinn man aktuell in diesem Zustand erreicht, während $U(z)$ den in Zukunft zu erwartenden Gewinn ausdrückt, also alles, was man nach diesem Zustand bekommt. Abb. 5 zeigt die Nutzen der Zustände für das obige Beispiel ($\gamma = 1$, $R(z) = -0.04$ für alle Nichtendzustände). Je näher der Agent

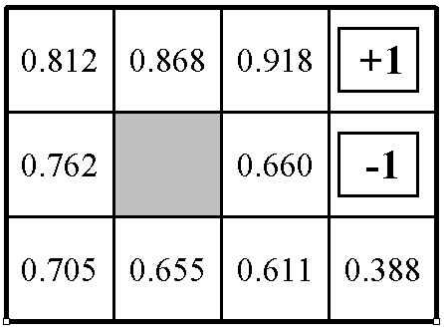


Abbildung 4.9: Nutzen der Zustände für das Beispiel

am Zustand "+1" ist, desto höher ist der Nutzen des Zustands, da weniger Schritte bis zum Endzustand "+1" benötigt werden.

Die Nutzenfunktion ermöglicht es dem Agenten, Aktionen nach der im Abschnitt über einfache Entscheidungen eingeführten Methode des maximalen erwarteten Nutzens (MEU = Maximum Expected Utility) auszuwählen, also

$$\pi^*(z) = \operatorname*{argmax}_a \sum_{z'} T(z, a, z')\, U(z')$$

Nach der Definition des Nutzens eines Zustandes hängt der Nutzen eines Zustandes direkt mit dem Nutzen des optimalen Folgezustandes zusammen:

$$U(z) = R(z) + \gamma \max_a \sum_{z'} T(z, a, z')\, U(z'),$$

er entspricht also den im aktuellen Zustandentstehenden Kosten zuzüglich des diskontierten erwarteten Nutzens des Folgezustands, der bei Wahl der optimalen Aktion erreicht wird. Die hier eingeführte Gleichung heißt Bellman-Gleichung. Die Nutzen der Zustände sind Lösungen der zugehörigen Bellman-Gleichungen.

$$\begin{aligned}
\text{Bsp.: } U(1,1) = -0.04 + \gamma \cdot \max\{\ & 0.8U(1,2) + 0.1U(2,1) + 0.1U(1,1), && (Up)\\
& 0.9U(1,1) + 0.1U(1,2), && (Left)\\
& 0.9U(1,1) + 0.1U(2,1), && (Down)\\
& 0.8U(2,1) + 0.1U(1,2) + 0.1U(1,1)\ \} && (Right)
\end{aligned}$$

Durch Einsetzen der Zahlen aus Abb. 4.9 sieht man, dass die beste Aktion ist, nach oben zu gehen.

Der Wertiterationsalgorithmus Der Wertiterationsalgorithmus, der eine optimale Lösung liefern soll, baut auf den Bellman-Gleichungen auf. In einer Welt mit n Zuständen gibt es n Gleichungen, eine für jeden Zustand. Es gibt also n Gleichungen mit n Unbekanntem $U(z_i)$, $i = 1, \ldots, n$. Allerdings sind die Gleichungen nichtlinear, da der max-Operator in ihnen vorkommt. Daher wählt man einen iterativen Ansatz zur Lösung des Gleichungssystems. Ausgehend von willkürlich gewählten Anfangswerten berechnet man die rechte Seite

function VALUE-ITERATION(mdp, ε) **returns** a utility function
  inputs: mdp, an MDP with states S, transition model T,
    reward function R
    discount γ
    ε, the maximum error allowed in the utility of any state
  local variables: U,U', vectors of utilities for states in S, initially zero
    δ, the maximum change in the utility of any state in an iteration

  repeat
    $U \longleftarrow U'$; $\delta \longleftarrow 0$
    for each state s **in** S **do**
      $U'[s] \longleftarrow R[s] + \gamma \max_a \sum_{s'} T(s, a, s')\, U[s']$
      if $|U'[s] - U[s]| > \delta$ **then** $\delta \longleftarrow |U'[s] - U[s]|$
  until $\delta < \varepsilon(1-\gamma)/\gamma$
  return U

Abbildung 4.10: Implementierung der Wertiteration nach [1]

der Gleichungen und setzt den so errechneten Wert in die linke Seite der Gleichung ein. Dabei aktualisiert man bereits errechnete Werte mit Hilfe des für die Nachbarzustände errechneten Nutzens. Dieses Verfahren wiederholt man solange, bis ein Gleichgewicht erreicht ist. Eine sogenannte Bellman-Aktualisierung sieht also folgendermaßen aus:

$$U_{i+1}(z) \leftarrow R(z) + \gamma \max_a \sum_{z'} T(z, a, z')\, U_i(z'),$$

wobei $U_i(z)$ der Nutzen des Zustands z im i-ten Iterationsschritt ist. Unendlich häufige Anwendung garantiert, dass irgendwann ein Gleichgewicht erreicht wird. Die so gefundenen Werte müssen Lösungen der Bellman-Gleichungen sein, wie weiter unten begründet sogar die einzigen, und die zugehörige Taktik ist optimal. In Abb. 6 ist eine Formulierung des Algorithmus in Pseudocode angegeben.

Konvergenz des Wertiterationsalgorithmus Oben wurde behauptet, dass der Algorithmus konvergiert und eindeutige Lösungen liefert. Das soll in diesem Abschnitt gezeigt werden. Außerdem soll gezeigt werden, dass der Fehler beim Nutzen der Zustände schnell verschwindet, so dass der Algorithmus terminiert.

Bei der Bellman-Aktualisierung handelt es sich um eine Kontraktion mit Kontraktionsfaktor c, es gilt also $\|BU_i - BU_i'\| \leq \gamma \|U_i - U\|$, wobei B der Operator für die Bellman-Aktualisierung ist, BU_i ist also die Bellman-Aktualisierung auf U_i angewendet. Nach dem Banachschen Fixpunktsatz hat eine Kontraktion genau einen Fixpunkt, gegen den die Lösungen bei wiederholter Anwendung des Operators konvergieren. Die Lösungen, die durch Anwendung der Bellman-Aktualisierung berechnet werden, sind also eindeutig.

Als nächstes wollen wir eine Abschätzung der Zahl der benötigten Iterationsschritte angeben. Es gilt: $||BUi - BU|| <= \gamma||Ui - U||$

Es ist bereits bekannt, dass die Nutzen der Zustände alle im Intervall $[-R_{\max}/(1-\gamma); +R_{\max}/(1-\gamma)]$ liegen. Das bedeutet, dass für den Anfangsfehler gilt:

$$\|U_0 - U\| \leq 2R_{\max}/(1-\gamma).$$

Angenommen, N sei die Zahl der Iterationen, die man benötigt, um einen Fehler zu erreichen, der kleiner als ε ist. Es soll also gelten: $\gamma_N + 2R_{\max}/(1-\gamma) \leq \varepsilon$. Dazu werden $N = \left\lceil \log\left(\frac{2R_{\max}}{\varepsilon(1-\gamma)}\right) / \log(\frac{1}{\gamma}) \right\rceil$ Iterationen benötigt. Für Werte von γ, die nahe bei 0 liegen, erhält man eine schnelle Konvergenz des Verfahrens. Nachteilig sind Werte nahe 1, da dann N sehr groß wird, es sind also sehr viele Iterationsschritte nötig, um eine gewisse Genauigkeit des Ergebnisses zu garantieren.

Man kann zeigen, dass es genügt, wenn sich die Werte für den Nutzen der Zustände nur wenig ändern, um eine gewisse Genauigkeit der Lösung zu garantieren. Es gilt nämlich:

$$\|U_{i+1} - U_i\| < \varepsilon(1-\gamma)/\gamma \;\Rightarrow\; \|U_{i+1} - U\| < \varepsilon$$

Dies ist auch die Abbruchbedingung im Wertiterationsalgorithmus (s. Abb. 6).

Bisher wurde nur der absolute Fehler betrachtet. Was wirklich interessiert, ist jedoch, wie gut die Entscheidungen sind, die der Agent auf Grund der berechneten Werte trifft, d.h. inwieweit die angenäherten Werte eine Taktik implizieren, die mit der optimalen Taktik übereinstimmt.

Sei $U_\pi(z)$ der Nutzen eines Zustands bei Anwendung der Taktik π. Dann ist der Taktikverlust $\|U_\pi - U\|$ der größtmögliche Verlust, den der Agent erfahren kann, wenn er statt der optimalen Taktik π^* die Taktik π anwendet. Der Taktikverlust von π hängt mit dem Fehler U_i folgendermaßen zusammen: Wenn $\|U_i - U\| < \varepsilon$, dann ist $\|U_\pi - U\| < 2\varepsilon\frac{\gamma}{1-\gamma}$. In der Praxis kommt es oft vor, dass die optimale Taktik erreicht ist, obwohl der Fehler noch relativ groß ist.

Jetzt hat man alles, was man braucht, um Wertiteration in der Praxis zu verwenden. Ma weiß, dass das Verfahren zu den richtigen Werten konvergiert und dass man den Fehler beschränken kann, wenn man das Verfahren nach einer endlichen Zahl von Iterationsschritten abbricht und können die Abweichung von der optimalen Taktik beschränken.

Taktikiteration

Aus dem vergangenen Abschnitt ist bekannt, dass man eine optimale Taktik auch finden kann, ohne die genauen Werte des Nutzens der einzelnen Zustände zu kennen. Wenn eine mögliche Aktion eindeutig besser ist als alle andern, dann ist der genaue Nutzen des Zustands uninteressant. Dieses Wissen impliziert eine andere Möglichkeit, um optimale Taktiken zu finden.

Der Taktikiterationsalgorithmus wiederholt die beiden folgenden Schritte ausgehend von einer Ausgangstaktik π_0:

Taktikauswertung: zu einer gegebenen Taktik π berechne $U_i = U_\pi$, den Nutzen jedes Zustands, falls die Taktik π verwendet würde. Taktikverbesserung:

function POLICY-ITERATION(mdp) **returns** a policy
  inputs: mdp, an MDP with states S, transition model T
  local variables: U,U', vectors of utilities for states in S, initially zero
  π, a policy vector indexed by state, initially random

  repeat
    $U \longleftarrow$ POLICY-EVALUATION(π, U, mdp)
    unchanged? $\longleftarrow$ true
    for each state s **in** S **do**
      if $\max_a \sum_{s'} T(s,a,s')U[s'] > \sum_{s'} T(s,\pi[s],s')U[s']$ **then**
        $\pi[s] \longleftarrow \text{argmax}_a \sum_{s'} T(s,a,s')U[s']$
        unchanged? $\longleftarrow$ false
  until *unchanged*?
  return P

Abbildung 4.11: Implementierung der Taktikiteration nach [1]

Berechne eine neue MEU-Taktik π_{i+1}, indem einen Schritt vorausgeschaut wird (aufbauend auf U_i)

Der Algorithmus terminiert, wenn der Schritt der Taktikverbesserung keine Änderung des Nutzens bringt. Es ist bekannt, dass die Nutzenfunktion U_i ein Fixpunkt der Bellman-Aktualisierung und damit Lösung der Bellman-Gleichungen ist. Somit ist π eine optimale Taktik. Da es für eine Umgebung mit endlich vielen Zuständen nur endlich viele Taktiken gibt, und da man zeigen kann, dass jeder Iterationsschritt eine verbesserte Taktik liefert, muss das Verfahren terminieren. Der Algorithmus ist in Abb.8 in Pseudocode angegeben.

Es stellt sich heraus, dass uns eine Vereinfachung der Bellman-Gleichungen gelungen ist, die den Nutzen eines Zustands z (bei Anwendung der Taktik π) mit dem Nutzen der Nachbarzustände verbindet:

$$U_i(z) = R(z) + \gamma \sum_{z'} T(z, \pi_i(z), z')\, U_i(z')$$

Der entscheidende Punkt ist, dass diese Gleichungen linear sind, da der max-Operator wegfällt. Für n Zustände haben wir n Gleichungen mit n Unbekannten, die in $O(n^3)$ gelöst werden können. Für kleine Umgebungen ist Taktikiteration oft der effizienteste Lösungsweg. Für größere Zustandsmengen ist $O(n^3)$ ein sehr hoher Aufwand. Glücklicherweise können wir auf exakte Taktikiteration verzichten. Stattdessen können wir einige vereinfachte Iterationsschritte machen, um eine verhältnismäßig gute Annäherung der Nutzen zu erhalten. Die vereinfachte Bellman-Aktualisierung für dieses Verfahren sieht folgendermaßen aus:

$$U_{i+1}(z) \leftarrow R(z) + \gamma \sum_{z'} T(z, \pi_i(z), z')\, U_i(z')$$

Dies wird k-mal wiederholt, um die nächste Nutzenabschätzung zu erhalten. Den Algorithmus, den wir so erhalten, nennt man modifizierte Taktikiteration (modified policy iteration).

4.4 Zusammenfassung

Dieses Kapitel zeigt uns, wie wir unter Zuhilfenahme der Nutzentheorie Entscheidungen und Präferenzen mathematisch erfassen können.

1. Die Entscheidungstheorie vereint die Nutzentheorie und die Wahrscheinlichkeitstheorie und befähigen uns, Entscheidungen gemäß unserer Präferenzen zu treffen.

2. Ein rational agierender Agent ist ein Agent, der seine Nutzenfunktion maximiert. Seine Entscheidungen sind somit stets auf seine Nutzenfunktion zurückführbar.Stimmt die Nutzenfunktion nicht mit den Entscheidungen des Agenten überein, so muss die Nutzenfunktion entsprechend abgeändert werden.

3. Entscheidungsnetzwerke stellen eine einfache Möglichkeit dar, Abhängigkeiten bezüglich Entscheidungen und ihren Nutzen graphisch darzustellen. Sie sind eine Erweiterung von Bayeschen Netzen um Entscheidungs- und Nutzenknoten.

4. Der Wert einer Information ist die Differenz vom *erwarteten* Nutzen mit, und dem Nutzen ohne diese Information.

5. Ein Entscheidungstheoretischer Agent wählt gemäß der Entscheidungstheorie aus. Er weiß wieviel ihm eine Information wert ist und seine Entscheidungen sind immer so gut wie die ihm zugrunde liegende Nutzenfunktion.

Im zweiten Teil des Kapitel haben wir sequentielle Entscheidungsprobleme kennengelernt. Zu deren Lösung reicht eine einzelne Entscheidung nicht aus, man benötigt statt dessen eine Reihe von Entscheidungen. Eine formale Darstellung dieser Probleme liefern die MDPs (=Markov Decision Process).

Eine Lösung eines sequentiellen Entscheidungsproblems erhält man, indem man für jeden Zustand eine Aktion angibt, die der Agent ausführen soll, wenn er dorthin gelangt. Solche Lösungen heißen auch Taktik.

Die Lösung eines MDPs heißt optimal, wenn sie den größtmöglichen Nutzen liefert. Der Nutzen eines Zustands lässt sich bestimmen, indem man das erwartete Entgelt berechnet, das man erhält, wenn man von diesem Zustand aus weitergeht.

Zur Bestimmung solcher Probleme gibt es 2 Algorithmen, den Wertiterationsalgorithmus und den Tatktikiterationsalgorithmus. Bei der Wertiteration berechnet man iterativ den Nutzen der einzelnen Zustände und leitet daraus die optimale Taktik ab. Bei der Taktikiteration wird ausgehend von einer Taktik versucht, eine Verbesserung zu erreichen, indem man die aus der Taktik resultierenden Nutzen der einzelnen Zustände betrachtet.

Literaturverzeichnis

[1] Russel S., Norvig P.: *Artificial Intelligence – A Modern Approach*, Second Edition, Prentice Hall, 2003.

[2] Varian: *Grundzüge der Mikroökonomik*, Fünfte Auflage, Oldenbourg, 2003.

[3] *Deutsches Universalwörterbuch A-Z*, 2. Auflage, 1989.

[4] Goos, Gerhard: *Vorlesung über Informatik - Band I*, 3. überarbeitete Auflage, 2000.

[5] Keeney, R.L., Raiffa, H.: *Decisions with Multiple Objectives*, Cambridge University Press, UK, 1993.

[6] Bundesministerium für Bildung und Forschung: *Informationsgesellschaft Deutschland - Aktionsplan 2006*, http://www.bmbf.de/pub/aktionsprogramm_informationsgesellschaft_2006.pdf.