Reinforcement Learning

Reinhard Kuntz
Hauptseminar Maschinelles Lernen

16.06.2005

Abstract

In dieser Arbeit werden nach einer abgrenzenden Definition des Begriffes Reinforcement Learning und dem Vorstellen eines Szenarios grundlegende Aspekte wie das Agentenkonzept und ein formales Interaktionsmodell vorgestellt. Darauf aufbauend werden problematische Teilaspekte von Reinforcement Learning Szenarien detailliert betrachtet. Dabei werden schrittweise Lösungsansätze vorgestellt und in die grundlegenden Reinforcement Learning Methoden überführt.

1 Einleitung

Bei sehr vielen Lernverfahren im Bereich des Maschinenlernens lernt ein System anhand von Beispielen, die durch einen wissenden und außerhalb des Systems stehenden Lehrer gegeben werden. In vielen Fällen ist es jedoch nicht durchführbar, ausreichend viele korrekte und repräsentative Beispiele für alle Situationen zu erstellen [Sutton, Barto 1998, S. 7-9]. Unter Umständen sind entsprechend gute Lösungsbeispiele auch überhaupt nicht bekannt und sollen durch das lernende System erst erstellt werden. In solchen Situationen kommt das Reinforcement Learning (Verstärkendes Lernen, im Weiteren mit VL abgekürzt) zum Zuge. Der Lerner beim VL muss also auch in völlig unbekannten Umgebungen zurecht kommen und dazulernen - vergleichbar mit der Situation, dass man ein Spiel lernt, ohne dass einem die Regeln erklärt werden und man lediglich gesagt bekommt, ob man verloren oder gewonnen hat.

Bei der Beobachtung von Lebewesen stellt man schnell fest, dass das VL eine sehr natürliche Art zu lernen ist. Ein Gazellenkalb beispielsweise begibt sich schon wenige Minuten nach der Geburt auf seine Füße. Lediglich eine halbe Stunde später kann es sich mit 20 Meilen pro Stunde fortbewegen [Sutton, Barto 1998, S. 6]. Nicht ganz so schnell, aber nicht weniger faszinierend, ist es bei einem Kind, das laufen lernt. Es bekommt nicht gesagt, setze einen Fuß vor den anderen. Es probiert einfach und erlernt das Laufen. Dieser Lernprozess ist möglich, da es einen Ansporn gibt. Macht das Kind etwas nicht richtig, fällt es hin, was einer Bestrafung gleich kommt. Macht es aber alles richtig, kann es sich schneller und angenehmer von einem Ort zum anderen bewegen, was eine Art Belohnung ist. Es wird also versuchen einen möglichst optimalen ”Gefühlszustand” zu erreichen.

Dieses Prinzip in einen angenehmeren Zustand gelangen zu wollen und deshalb zu probieren, bis eine möglichst gute Lösung gefunden ist, wird im Bereich des Maschinenlernens nachgebildet und als VL bezeichnet.

2 Definition / Abgrenzung

Das VL lässt sich nicht durch eine bestimmte Lernmethode definieren. Es stellt eine spezielle Klasse von Lernproblemen bzw. Lernsituationen dar [Sutton, Barto 1998, S. 7-9]. Zum Lösen dieser Lernprobleme können dabei grundsätzlich jegliche Lernmethoden verwendet werden.

Wie schon in Abschnitt 1 angedeutet, lassen sich Lernsituationen unter anderem anhand des verfügbaren Feedbacktyps unterscheiden. Im Gebiet des Maschinenlernens werden normalerweise drei Fäl-

le unterschieden - überwachtes, nicht überwachtes und verstärkendes Lernen [Russel, Norvig 2004, S. 794f]. Beim überwachten Lernen werden dem Lerner korrespondierende Eingabe- und Ausgabebeispiele gegeben. Anhand dieser kann dann eine Funktion erlernt werden, die zukünftige Eingaben in entsprechende Ausgaben überführt. Auch beim nicht überwachten Lernen werden Eingabebeispiele gegeben, jedoch ohne spezifische Ausgabewerte. Das VL schließlich ist der allgemeinste Fall. Hier muss der Lerner völlig ohne gegebene Beispiele auskommen und kann lediglich anhand von Verstärkung bzw. Belohnung lernen. Oftmals ist dabei das Unterproblem zu bewältigen, dass gelernt werden muss, wie die Umgebung, in der sich der Lernende befindet, funktioniert. Es wird jedoch nicht ausgeschlossen, dass Vorwissen vorhanden ist. So könnte beispielsweise für einen Lerner die Skatregeln bekannt sein. Er weiß dadurch jedoch nicht, wie hoch er reizen muss und in welcher Reihenfolge er die Karten spielen sollte, um zu gewinnen.

Charakteristisch für VL-Lernsituationen ist das Lernen durch Interaktion, um dadurch ein Ziel zu erreichen [Sutton, Barto 1998, S. 7-9]. Hierbei spielen insbesondere gezieltes Ausprobieren und verzögerte Belohnungen eine gewichtige Rolle. Bestrafung wird im VL als geringe oder negative Belohnung betrachtet. VL stellt also den Versuch dar, das, wie in Abschnitt 1 gezeigt, sehr natürliche Lernvorgehen durch gezieltes Ausprobieren auf das Maschinenlernen abzubilden. Das Kernproblem beim VL, dass so in keiner anderen Art des Maschinenlernens vorkommt, ist, entscheiden zu müssen, ob es lohnend ist neue Lösungsmöglichkeiten auszuprobieren oder einen bewährten Ansatz zu wählen. Es muss also ein Gleichgewicht gefunden werden zwischen neuen Erkenntnissen, die noch höhere Belohnungen bringen könnten, und den Belohnungen bekannter Wege [Sutton, Barto 1998, S. 7-9].

3 Beispielszenario

In diesem Abschnitt soll als erster Schritt ein fiktives aber durchaus realistisches VL-Szenario vorgestellt werden. In einem zweiten Schritt wird dieses Szenario stark vereinfacht. In den folgenden Abschnitten wird das vereinfachte Szenario dann zum Erläutern verschiedener Aspekte und Methoden des VL herangezogen.

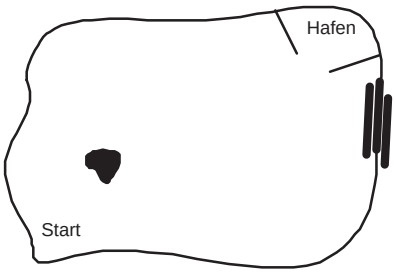

Abbildung 1: Stausee aus dem Szenario

3.1 Szenario

Ein Ingenieursteam möchte für ein Boot einen Autopilotsystem entwickeln. Als Zwischenziel haben die Ingenieure sich darauf geeinigt, dass das Boot auf einem Stausee, wie in Abbildung 1 gezeigt, von einem festgelegten Startpunkt aus sicher in den Hafen gelangen soll.

Auf dem See gibt es oft Windböhen aus verschiedensten Richtungen. Deshalb müssen die Entwickler davon ausgehen, dass das Steuern des Bootes in eine Richtung nicht immer zum gewünschten Effekt führt. Ein Vorgeben einer festen Route durch vordefinierte Steuerbefehle ist also nicht möglich.

Dem System steht zur Orientierung ein Radar zur Verfügung, mit dem es in kurzen Zeitabständen seine Position bestimmen kann. Die in Abbildung 1 gezeigte Umgebung ist also für das Autopilotsystem beobachtbar. Mithilfe der Beobachtungen soll das Boot steuern. Es sollen natürlich Kollision mit der Insel (siehe Abbildung 1) und dem Ufer vermieden werden. Ebenso darf das Boot der Staumauer (siehe Abbildung 1 rechts) nicht zu nahe kommen, da dies zum Kentern des Bootes führen würde. Kollisionen werden allerdings lediglich als Gefahr für das Boot und nicht für Insassen betrachtet, im Gegensatz zum Kentern des Bootes. Im Zweifel ist demnach eine Kollision einem Kentern vorzuziehen. Weiter soll das Boot einen

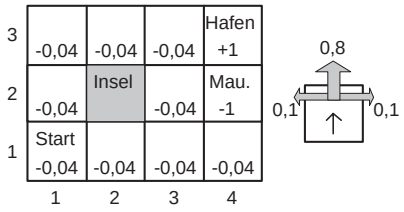

Abbildung 2: 3x4-Welt (in Anlehnung an [Russel, Norvig 2004, 750])

möglichst schnellen bzw. direkten Weg zum Hafen wählen. Die Sicherheit hat jedoch Vorrang vor der Geschwindigkeit.

Die Entwickler wollen in späteren Schritten das Autopilotsystem auch für den Einsatz in anderen Gewässern einsetzen. Das soll möglich sein, ohne für jedes Gewässer entsprechend Beispiele für gute Fahrwege zu entwerfen. Deshalb soll das System mit Hilfe von VL entwickelt werden.

3.2 Vereinfachtes Szenario

Das vorgestellte Szenario soll nun vereinfacht werden, um damit in folgenden Abschnitten verschiedene Aspekte des VL besser verdeutlichen zu können, ohne unnötig zu verwirren.

Wie schon erläutert soll für das System das Gewässer beobachtbar und somit die aktuelle Position bekannt sein. Legt man über das Gewässer ein Gitternetz entsteht eine vereinfachte Darstellung, wobei die Quadrate als mögliche Positionen zu verstehen sind. Das Gewässer kann dann als 3x4-Welt betrachtet werden, wie in Abbildung 2 dargestellt. Dabei soll die Startposition (1,1) sein und der Hafen (3,4). Die Staumauer befindet sich in (2,4) und die Insel in (2,2).

Zur Vereinfachung gehen wir davon aus, dass das Boot nur vier Bewegungsrichtungen kennt - oben, unten, links und rechts. Kollidiert das Boot mit der Insel oder dem Strand, verbleibt es in der Ausgangsposition.

Die durch mögliche Windböhen entstehende Unsicherheit der Steueraktionen soll, wie in Abbildung 2 rechts dargestellt, ”simuliert” werden. Jede Steueraktion soll mit einer Wahrscheinlichkeit von 80% den gewünschten Effekt erzielen. Mit einer Wahrscheinlichkeit von je 10% weicht das Ergebnis nach links oder rechts von der Steuerrichtung ab, wie in Abbildung 2 für die Aktion oben dargestellt.

4 Interaktionsmodell

Verstärkendes Lernen ist lernen durch Interaktion wie in 2 schon erläutert. Um eine einheitliche Sicht bzw. ein einheitliches Verständnis für VL-Situationen zu erlangen, ist ein Modell für den Lernenden, die Lernsituation und die Interaktion des Lernenden erforderlich. Ein entsprechendes Interaktionsmodell wird in diesem Abschnitt vorgestellt, wobei zunächst das grundlegende allgemeine Agentenkonzept erläutert wird. Darauf aufbauend wird eine Formalisierung des Interaktionsmodells durchgeführt.

4.1 Agent und Umwelt

Für VL-Situationen bietet es sich an, den Lernenden als Agenten zu betrachten. Das allgemeine Agentenkonzept ist als ein Werkzeug zur Analyse von Systemen oder als Vorstellungsmodell zu sehen. Es lässt sich nicht formal definieren und stellt keine absolute Charakterisierung von Agenten dar. Deshalb ist eine eindeutige Unterteilung der Welt in Agenten und nicht-Agenten nicht möglich. Ein Agent kann demnach eine Maschine, ein Roboter, ein Mensch oder ein Computerprogramm sein [Kuntz 2005, S. 2]. Außerdem ist es möglich, Systeme oder Programme als Zusammensetzung mehrerer miteinander interagiernder Agenten zu betrachten. Eine weitere Möglichkeit ist, verschiedene Ebenen der Abstraktion bei der Betrachtung eines Systems einzuführen, so dass ein Agent wiederum aus mehreren Agenten bestehen kann.

Die wichtigsten Eigenschaften eines Agenten sind die Eigenständigkeit (Autonomie), die Interaktion mit seiner Umwelt, die andauernde Verfügbarkeit bzw. Aktivität des Agenten sowie das zielgerichtete Agieren des Agenten [Kuntz 2005, S. 2]. Aus

diesen charakteristischen Eigenschaften von Agenten ergibt sich, dass sie zyklisch arbeiten. Es werden dabei drei Phasen unterschieden - die "Informationsaufnahme", die "Wissensverarbeitung und Entscheidung" und die "Aktionsausführung" [Kuntz 2005, S. 4]. Dieser Zyklus entspricht der in [Lippold 2005, S. 2] dargestellten "Lernsituation Agent" bzw. "lernen durch experimentieren" aus dem dort vorgestellten "Rahmenmodell für das Maschinelle Lernen". Allerdings werden die Elemente jedes Zyklus in [Lippold 2005, S. 2] in 4 Phasen eingeteilt, wobei dieselben Elemente innerhalb eines Zyklus enthalten sind.

Der Agent nimmt in der ersten Phase Informationen aus der Umwelt auf. In der letzten Phase führt er Aktionen "auf" dieser aus. Als Umwelt wird in diesem Zusammenhang alles bezeichnet, womit der Lernende bzw. der Agent interagiert. Im Endeffekt also alles außerhalb des Agenten [Sutton, Barto 1998, S. 51f]. Hierbei wird immer ein für den Agenten relevanter Ausschnitt der Welt auf einer für die Aufgabe des Agenten passenden Abstraktionsebene betrachtet. Die Umwelt kann dabei auch andere Agenten beinhalten [Kuntz 2005, S. 3].

Bei der Verwendung des Agentenkonzeptes im VL ist zu beachten, dass alle Elemente, die nicht vom Agenten direkt beeinflusst bzw. kontrolliert werden können, als Umwelt anzusehen sind. Hierdurch soll die Kapselung des eigentlichen VL-Problems erreicht werden. Diese logische Grenze zwischen Umwelt und Agent entspricht hierdurch oftmals nicht der physikalischen [Sutton, Barto 1998, S. 53f].

Für das in 3 vorgestellte Szenario heißt das, dass beispielsweise das Radar als Teil der Umwelt zu betrachten ist, obwohl es physisch Teil des Bootes bzw. des Autopilotsystems ist.

Im allgemeinen Agentenkonzept besteht die Interaktion des Agenten mit der Umwelt zum einen aus den Aktionen des Agenten, die er mit Hilfe seiner Aktuatoren ausführt, zum anderen aus dem Wahrnehmen der Umwelt in der jeweils aktuellen Situation. Im Kontext des VL ist zu beachten, dass die Wahrnehmung des Agenten differenzierter betrachtet wird, weil Belohnungen für das VL notwendig sind. Deshalb wird zwischen Belohnungen und sonstigen Wahrnehmungen unterschieden. Belohnungen werden zwar oft als Teil der Wahrnehmungen betrachtet, müssen aber vom Agenten als solche erkennbar sein. Aus diesem Grund wird in [Russel, Norvig 2004, S. 927] von einer "festen Verdrahtung" für die Belohnung gesprochen und in [Sutton, Barto 1998, S. 52] werden Belohnungen sogar als komplett von der Wahrnehmung getrennt betrachtet.

4.2 Formales Interaktionsmodell

Um besser arbeiten zu können, soll nun das Interaktionsmodell noch etwas vereinfacht und formalisiert werden. Das verwendete formale Interaktionsmodell ist an das in [Sutton, Barto 1998, S. 52] angelehnt.

In Abbildung 3 ist oben rechts der Agent abgebildet. Für jeden Arbeitszyklus dessen definieren wir einen in kausaler Ordnung stehenden Zeitstempel $t \in \{0, 1, 2, ...\}$. In jedem Zeitschritt bzw. Zyklus nimmt der Agent seine Umgebung wahr. Er erhält eine Situation $s_t \in S$, wie in Abbildung 3 links oben dargestellt. Dabei ist S die Menge aller möglichen Situationen, die der Agent aus der Umwelt wahrnehmen kann.

Bezogen auf die 3x4-Welt (siehe 3.2) heißt das, dass der Agent zunächst die Situation (1,1) wahrnimmt. Die Menge S besteht dabei aus allen Feldern (x,y).

In der 2. Phase wählt der Agent eine Aktion $a_t \in A(s_t)$. Die in einer Situation s_t vom Agenten durchführbaren Aktionen sind dabei durch $A(s_t)$ festgelegt.

In dem im Abschnitt 3.2 vorgestellten Szenario ergibt sich an dieser Stelle ein Sonderfall, da in jeder Situation die Aktionen rechts, links, oben und unten möglich sind.

Einen Zyklus bzw. Zeitschritt später (in Abbildung 3 links unten dargestellt) erhält der Agent die Belohnung b_{t+1} für seine Aktion a_t und befindet sich in einer neuen Situation s_{t+1}.

Die Belohnung wird durch einen numerischen Wert

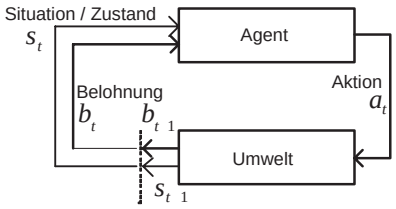

Abbildung 3: Interaktionsmodell beim VL (in Anlehnung an [Sutton, Barto 1998, S. 52]

repräsentiert und ist durch eine Belohnungsfunktion festgelegt, die vom Agenten nicht verändert werden kann. Diese Funktion ist als Abbildung von Situationen s_t bzw. Situation-Aktion-Paaren (s_t, a_t) auf eine Ausprägung der Belohnung zu betrachten. Sie definiert gut und schlecht für den Agenten, wobei die Belohnungsfunktion selbst für den Agenten unbekannt ist.

In der 3x4-Welt soll die Belohnungsfunktion lediglich von der Situation abhängig sein. Wie in der Abbildung 2 schon eingetragen, sei die Belohnung für das Erreichen des Zieles $+1$, für das Heranfahren an die Staumauer -1 und für jede andere Situation $-0,04$, wodurch erreicht wird, dass der Agent möglichst schnell zum Ziel kommen möchte.

Zu beachten ist, dass in sehr vielen VL-Szenarien die Situation s_{t+1} und die Belohnung b_{t+1} als stochastisch angesehen werden müssen und nicht deterministisch von s_t und a_t abhängen. Dies gilt auch für die 3x4-Welt, da durch die Möglichkeit der Windböhen die Ausführung der Aktionen als unsicher zu betrachten ist, wie schon in 3.2 beschrieben.

Die Ergebniswahrscheinlichkeit für jede Aktion in jeder Situation wird meist als Übergangsmodell oder Transitionsmodell bezeichnet [Russel, Norvig 2004, S. 750]. Wenn alle möglichen Folgesituationen s^x_{t+1} der Situation s_t und der Aktion a_t mit Hilfe von $x \in [1, 2, ...]$ durchnummeriert werden, lässt sich die Wahrscheinlichkeit, dass die Situation s^2_{t+1} eintritt, durch $T(s_t, a_t, s^2_{t+1})$ ausdrücken. Ob dem Agenten das Transitionsmodell bekannt ist, hängt von der VL-Methode ab (siehe Abschnitte 6 und 7).

Beispielsweise sind die Situationen $s^1_{t+1} = (1, 2)$, $s^2_{t+1} = (1, 1)$ und $s^3_{t+1} = (2, 1)$ in der 3x4-Welt die möglichen Folgesituationen von $s_t = (1, 1)$ und $a_t = oben$.

5 Das VL-Problem

Im letzten Abschnitt wurde schrittweise ein formales Interaktionsmodell für VL-Situationen gebildet. Darauf aufbauend soll in diesem Abschnitt auf verschiedene problematische Teilaspekte von VL-Situationen eingegangen werden. Grundlegende Lösungsansätze für die einzelnen Aspekte werden dabei vorgestellt.

5.1 Sequentielle Entscheidungen

In VL-Situationen muss der Agent, um seine Aufgabe zu erfüllen, mehrere Entscheidungen treffen, die jeweils von den vorausgehenden abhängig sind. Diese sog. sequentiellen Entscheidungen müssen also einen Kontext bzw. eine Vergangenheit mit einbeziehen. Dieser Sachverhalt lässt sich an dem Beispiel eines Gespräches verdeutlichen, in dem ein Gesprächspartner ”ja” sagt. Das ”ja” kann verschiedenste Bedeutungen haben, in Abhängigkeit vom bisherigen Inhalt des Gespräches [Sutton, Barto 1998, S. 61]. Es müssen also alle vergangenen Situationen bei jeder Entscheidung mitbetrachtet werden. Im schlimmsten Fall können das unendlich viele werden, wenn der Agent sehr lange arbeitet, was dann nicht implementierbar ist.

Für die meisten VL-Szenarien ist es aber möglich, durch eine entsprechende Wahl der Situationsrepräsentation Pfadunabhängigkeit bzw. Markov’sche Situationen zu erhalten. Die Markov-Annahme besagt, dass der aktuelle Zustand lediglich von einem endlichen Verlauf von vorhergegangenen Situationen abhängig ist [Russel, Norvig 2004, S. 661]. Man spricht dann auch von Markov’schen Situationen.

Die 3x4-Welt ist Markov’sch, da die Aktionen nur von der aktuellen Position abhängen. Anders ausgedrückt enthält eine Situation mit der Position alles relevante aus der Vergangenheit. Man spricht

hier von Markov'schen Situationen erster Stufe. Wenn es in der 3x4-Welt nur die Aktionen drehen nach links bzw. rechts und fahren gäbe, müsste die alte Fahrtrichtung zusätzlich bekannt sein. Dieser Sachverhalt könnte allerdings ebenfalls Markov'sch gemacht werden, indem in die Situationsrepräsentation nicht nur die Position, sondern auch die Ausrichtung des Bootes aufgenommen würde. Beispielsweise mit (x,y,r), wobei x und y die Position darstellen und r die Bootsausrichtung mit den Werten - rechts, links, oben, unten - angibt.

Liegen Markov'sche Situationen vor, so spricht man bei dem in 4.2 vorgestellten Transitionsmodell vom Markow'schen Transitionsmodell [Russel, Norvig 2004, 750]. Ein sequentielles Entscheidungsproblem mit einer vollständig beobachtbaren Umwelt, bei welchem ein Markow'sches Transitionsmodell und eine additive Wertfunktion (siehe 5.2) vorliegt, wird Markov-Entscheidungs-Prozess (MEP) genannt. Ein MEP ist durch einen Ausgangszustand, ein Transitionsmodell und eine Wertfunktion vollständig definiert [Russel, Norvig 2004, S. 751].

5.2 Ziele und Belohnungen

Wie im Szenario in Abschnitt 3 beschrieben, ist es das Ziel, dass das Boot den Hafen bzw. (3,4) möglichst schnell, vor allem aber sicher erreicht. Es stellt sich also die Frage, wie sich das allgemein für einen VL-Agenten beschreiben lässt.

Am Beispiel der 3x4-Welt ist ersichtlich, dass die Belohnungen, die dem Agenten zur Verfügung stehen, kurzfristig sind. Sie beziehen sich lediglich auf eine Situation - formaler betrachtet auf eine Situation oder ein Situation-Aktion-Paar (siehe 4.2). Um von der Startsituation (1,1) in den Hafen (3,4) zu gelangen, ist jedoch eine Sequenz von Aktionen des Agenten notwendig. Dabei erhält er ebenfalls eine Sequenz von Belohnungen $b_{t+1}, b_{t+2}, b_{t+3}, ...$ welche den einzigen Anhaltspunkt für "gut" oder "schlecht" für den Agenten bietet [Sutton, Barto 1998, S. 57f]. Fraglich nur, welcher Teil der Belohnungssequenz vom Agenten maximiert werden soll, wo beispielsweise in (2,1) die Belohnung für jede Aktion -0,04 ist.

Wir sind also an einem langfristigen Maß interessiert, dem Wert einer Situation. Dieser drückt aus, wie gut eine Situation für das Erreichen des Zieles ist. Der langfristige Wert einer Situation besteht gewissermaßen aus der Summe der Belohnungen, die der Agent von dieser Situation aus in der Zukunft erreichen kann. Während Belohungen explizit gegeben werden, muss der Wert über die gesamte Situationsfolge in der Lebenszeit des Agenten geschätzt und zurückverfolgt werden [Sutton, Barto 1998, S. 7-9].

Im einfachsten Fall lässt sich der Wert einer Situation durch die sog. additive Wertfunktion

$$W(s_t) = b_{t+1} + b_{t+2} + b_{t+1} + ... + b_T$$

berechnen, wobei T der letzte Zeitpunkt bzw. eine terminale Situation ist [Sutton, Barto 1998, S. 57f], [Russel, Norvig 2004, S. 753ff].

In der 3x4-Welt ist dies der Fall. Der Agent hat seine Aufgabe beendet, wenn er im Hafen (3,4) angelangt ist, oder er (2,4) erreicht hat. Weitere Beispiele für solche endlichen Anwendungssequenzen finden sich bei Spielen mit den entsprechenden "game over Situationen".

Ist jedoch $T = \infty$, erhalten wir bei der additiven Wertfunktion eine unendliche Summe, die nicht berechenbar ist. In solchen Fällen muss dann die Methode der verminderten Werte herangezogen werden [Sutton, Barto 1998, S. 57f].

$$\begin{aligned} W(s_t) &= b_{t+1} + \gamma b_{t+2} + \gamma^2 b_{t+1} + ... \\ &= \textstyle\sum_{k=0}^{\infty} \gamma^k b_{t+k+1} \end{aligned}$$

Die Verminderungsrate ist dabei durch γ dargestellt, wobei $0 \leq \gamma \leq 1$ gilt. Setzt man γ gleich Eins, erhält man die oben dargestellte additive Wertfunktion, die also ein Sonderfall der verminderten Wertfunktion ist. Wird γ kleiner Eins gewählt, erhalten wir immer eine endliche Summe, auch wenn die Reihe unendlich ist. Um so mehr man den Wert von γ Null annähert, wird bei der Berechnung des Wertes die Zukunft unbedeutender, bis hin zu $\gamma = 0$, wo lediglich die erhaltene Belohnung, also der lokale Wert, betrachtet wird.

Wir haben mit der verminderten Wertfunktion ein Werkzeug, um rückwirkend den Wert jeder Situa-

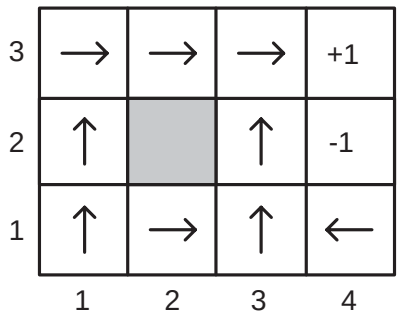

Abbildung 4: Eine Strategie für die 3x4-Welt

tion im Bezug zu einem längerfristigen Ziel zu berechnen. Das Ziel eines VL-Agenten könnte somit vereinfachend ausgedrückt als das Bestreben beschrieben werden, zu jedem Zeitpunkt zu versuchen, die Folgesituation mit dem größten Wert zu erreichen.

5.3 Strategie

Offen ist nun noch die Frage wie eine Lösung für ein VL-Problem bzw. vereinfacht für einen MEP aussieht. Eine feste Aktionsfolge stellt keine Lösung dar, da diese wegen der Ergebnisunsicherheit der Aktionen nicht immer zum Ziel führt, wie in 3.1 beschrieben. Eine Lösung muss also angeben, was der Agent in jeder möglichen Situation tun soll.

Ein solcher Aktionsplan für jede mögliche Situation wird als Strategie [Sutton, Barto 1998] oder Taktik [Russel, Norvig 2004, S. 751] bezeichnet und mit π notiert. Vorstellen kann man sich eine Strategie als eine Funktion $\pi(s_t)$, die abhängig von der wahrgenommenen Situation s_t eine Aktion zurück gibt. Dabei ist die zurückgegebene Situation die aus Sicht der Strategie zu präferierende. Diese Funktion kann realisiert sein durch eine einfache Tabelle bis hin zu komplexen Berechnungsprozessen. In realen Anwendungen sind Strategien oft als stochastisch zu betrachten [Sutton, Barto 1998, S. 7-9].

Eine für die 3x4-Welt mögliche Strategie ist in Abbildung 4 dargestellt. Sie ist in der Hinsicht optimal, dass versucht wird, schnellstmöglich den Hafen (3,4) zu erreichen. Dabei wird allerdings vernachlässigt, dass das auch möglichst sicher geschehen sollte. Vereinfacht ausgedrückt ist das Lösen von VL-Problemen das Suchen einer optimalen Strategie π^*, die dann verwendet wird.

6 Nutzen lernen

Wenn die Lösung eines VL-Problems eine optimale Strategie ist, muss ein VL-Agent die Güte einer Strategie bewerten bzw. herausfinden können. In diesem Abschnitt soll erläutert werden, wie der Agent den Nutzen einer Strategie lernen kann. In [Russel, Norvig 2004, S. 929ff] wird dieser Lernvorgang als passives VL bezeichnet, während das Lernen des Nutzen in [Sutton, Barto 1998] lediglich als Teil des VL betrachtet wird.

6.1 Direkte Nutzenschätzung

Geht man davon aus, dass eine Strategie π festgelegt ist, sind dadurch die Aktionen des Agenten mit $a_t = \pi(s_t)$ ebenfalls festgelegt. Es ist dem Agenten aber nicht bekannt, wie gut die Strategie ist. Gewissermaßen muss eine Strategiebewertung bzw. eine Bestimmung des Nutzens der Situationen in Abhängigkeit von der Strategie durchgeführt werden. Die Nutzenfunktion $N^\pi(s_t)$ [Russel, Norvig 2004, S. 929f] ist also gesucht, wobei dem Agent das Transitionsmodell $T(s_t, a_t, s_t^x)$ und die Belohnungsfunktion unbekannt sind.

Um die Nutzenfunktion herauszufinden, lässt man den Agenten mehrere Versuche durchführen und erhält dadurch Zustands- und Belohnungsfolgen. In der 3x4-Welt enden diese entweder mit dem Ziel (3,4) oder der Situation (2,4). Mögliche Folgen wären beispielsweise [Russel, Norvig 2004, S. 930]:

$$(1,1)_{-0,04} \rightarrow (1,2)_{-0,04} \rightarrow (1,3)_{-0,04} \rightarrow (1,2)_{-0,04} \rightarrow (1,3)_{-0,04} \rightarrow (2,3)_{-0,04} \rightarrow (3,3)_{-0,04} \rightarrow (4,3)_{+1}$$

$$(1,1)_{-0,04} \rightarrow (1,2)_{-0,04} \rightarrow (1,3)_{-0,04} \rightarrow (2,3)_{-0,04} \rightarrow (3,3)_{-0,04} \rightarrow (3,2)_{-0,04} \rightarrow (3,3)_{-0,04} \rightarrow (4,3)_{+1}$$

$(1,1)_{-0,04} \rightarrow (2,1)_{-0,04} \rightarrow (3,1)_{-0,04} \rightarrow$
$(3,2)_{-0,04} \rightarrow (4,2)_{-1}$

Am Ende jeder Folge - nach dem Erreichen einer terminalen Situation - kann für jede Situation, die Situationsfolge rückwärts gehend, der Wert mittels der verminderten Wertfunktion berechnet werden. Dabei kann hier $\gamma = 1$ gesetzt werden. Der Nutzen einer Situation berechnet sich dann aus dem Mittel alter Wertberechnungen und der neuen Berechnung. Jede Folge kann somit als Stichprobe angesehen werden. Nach unendlich vielen Versuchen konvergiert das Stichprobenmittel irgendwann zum wahren erwarteten Nutzen [Russel, Norvig 2004, S. 930].

$$N^{\pi}(s_t) = \sum_{k=0}^{\infty} \gamma^k b_{t+k+1} | \pi$$

6.2 Passive adaptive dynamische Programmierung

Bei der direkten Nutzenschätzung wird nicht beachtet, dass der Nutzen der Situationen voneinander abhängig ist.

Dieser Sachverhalt wird durch die sog. vereinfachte Bellmann-Gleichung ausgedrückt. Dabei ist der Nutzen jeder Situation gleich seiner eigenen Belohnung plus dem erwarteten Nutzen seiner Nachfolgersituationen [Russel, Norvig 2004, S. 931], wie in der folgenden Gleichung beschrieben.

$$N^{\pi}(s_t) = b_t + \gamma \sum_{x=0}^{x=max} T(s_t, \pi(s_t), s_{t+1}^x) N^{\pi}(s_{t+1}^x)$$

Durch diese Gleichung ergibt sich allerdings das Problem, dass für den Agenten das Transitionsmodell nicht bekannt ist. Er kann es aber lernen, wenn für ihn die Umwelt beobachtbar ist.

Eine Möglichkeit ist, das Transitionsmodell als Tabelle zu repräsentieren. Dabei trägt der Agent nach jedem Zyklus zum Situation-Aktion-Paar die wirkliche Nachfolgerposition ein. Die Transitionswahrscheinlichkeit lässt sich dann mittels der Aktionsfrequenz schätzen.

In den drei Pfaden in 6.1 wurde die Aktion rechts in (1,3) dreimal ausgeführt. In zwei dieser drei Fälle war das Resultat die Situation (2,3) - was gleichbedeutend mit der Schätzung $2/3$ für $T((1,3), rechts, (2,3))$ ist [Russel, Norvig 2004, S. 932].

Durch das Erlernen des Transitionsmodells ist es möglich mittels der sog. passiven adaptiven dynamischen Programmierung (ADP) den Nutzen zu lernen. Wie schon bei der direkten Nutzenschätzung macht der Agent hierfür Versuche. Dabei erlernt er das Transitionsmodell und verwendet es gleichzeitig zusammen mit den erhaltenen Belohnungen zur Berechnung des Nutzens der einzelnen Situationen mit Hilfe der vereinfachten Bellmann-Gleichung.

Verwendet man beispielsweise in der 3x4-Welt die in Abbildung 4 gezeigte Strategie, erhält man folgendes Gleichungssystem, wenn davon ausgegangen wird, dass der Agent das Transitionsmodell schon erlernt hat:

$N^{\pi}(1,1) = -0,04 + 0,8N^{\pi}(2,1) + 0,1N^{\pi}(1,1) + 0,1N^{\pi}(1,2)$
$N^{\pi}(1,2) = -0,04 + 0,8N^{\pi}(1,3) + 0,1N^{\pi}(1,2) + 0,1N^{\pi}(1,2)$
...

In jedem Zyklus haben wir also für m Situationen m Gleichungen mit m Unbekannten, die mit Standartmethoden in der Zeit $O(m^3)$ gelöst werden können [Russel, Norvig 2004, S. 762f].

6.3 Temporale Differenz (TD)

Das im letzten Abschnitt vorgestellte passive ADP birgt ein Problem. Bei großen Situationsräumen werden die notwendigen Berechnungen zu groß - bei Backgammon beispielsweise wird es notwendig, etwa 10^{50} Gleichungen mit 10^{50} Unbekannten in jedem Zyklus zu lösen [Russel, Norvig 2004, S. 932]. Beim TD-Ansatz versucht man das zu umgehen. Dabei ist die grundlegende Idee, dass zunächst die, bei idealer Nutzenschätzung geltende, lokal Bedingung definiert wird. Das kann beispielsweise durch die vereinfachte Bellmann-Gleichung geschehen. In einem weiteren Schritt bildet man eine Aktualisierung (auch TD-Gleichung genannt), die die bisherigen Schätzungen in Richtung der idea-

len Schätzung verschiebt.

Wenn ein Übergang von der Situation s_t in die Situation s_{t+1} erfolgt, kann die Aktualisierung durch

$$N^\pi(s_t) = N^\pi(s_t) + \alpha(b_t + \gamma N^\pi(s_{t+1}) - N^\pi(s_t))$$

erreicht werden. α stellt in der Gleichung den Lernratenparameter dar, der die lokale Veränderung begrenzt.

Es lässt sich zeigen, dass obwohl nur die beobachteten Nachfolger aktualisiert werden, der Durchschnittswert von $N^\pi(s_t)$ auf den korrekten Wert konvergiert. Verwendet man anstelle eines festen Wertes für α eine proportional zum Auftreten einer Situation steigende Funktion, konvergiert sogar $N^\pi(s_t)$ selbst auf den korrekten Wert. Zu beachten ist, dass die TD-Methode im Gegensatz zu passiven APD-Methode kein Transitionsmodell benötigt.

7 VL-Methoden

Im letzten Abschnitt wurden einige Möglichkeiten gezeigt, wie ein Agent unter einer festen Strategie den Nutzen von Situationen lernen kann. Um nach dem Verständnis von [Russel, Norvig 2004] einen aktiven VL-Agenten oder nach [Sutton, Barto 1998] eine vollständige VL-Situation vorliegen zu haben, muss der Agent auch seine Handlungen infolge gelernten Nutzens ändern.

Hierbei ergibt sich das sog. Explorationsproblem, das am Beispiel des n-armigen Banditen von verschiedenen Wissenschaftlern in der Vergangenheit ausführlich untersucht wurde. Der einarmige Bandit ist eine Glücksspielmaschine. Der Spieler kann eine Münze einwerfen und an einem Hebel ziehen. Falls er gewinnt, kann er die Gewinne einsammeln. Der n-armige Bandit ist eine fiktive Erweiterung des einarmigen Banditen und hat n-Arme bzw. Hebel. Der Spieler muss also stets einen Hebel wählen. Dabei muss er sich entscheiden, ob er einen Hebel wählt, der in der Vergangenheit Gewinne eingebracht hat oder ob er einen bisher noch nicht verwendeten Hebel betätigt, der unter Umständen einen höheren Gewinn bringen könnte [Russel, Norvig 2004, S. 938f].

Es ist möglich für eine derartige Problemstellung ein sinnvolles Schema zu finden, das irgendwann zum optimalen Verhalten des Agenten führt. Bei einem solchen Schema muss jede Aktion in jeder Situation unbegrenzt oft durchgeführt werden, um zu verhindern, dass eine optimale Aktion nicht als solche erkannt wird, weil zufällig ein ungewöhnlich schlechtes Ergebnis erreicht wurde. Andererseits muss dafür gesorgt werden, dass bei ausreichenden Kenntnissen nicht weiter getestet wird, sondern die erfolgversprechendsten Aktionen ausgeführt werden [Russel, Norvig 2004, 937ff].

In diesem Abschnitt sollen nun die in Abschnitt 6 vorgestellten Verfahren - ADP und TD - so angepasst werden, dass sie das Explorationsproblem berücksichtigen. Darauf aufbauend wird im Anschluss eine weitere VL-Methode vorgestellt.

7.1 Adaptive dynamische Programmierung

Ein Ansatz das Explorationsproblem zu lösen, ist Aktionen zu gewichten, also eine hohe Gewichtung für selten ausprobierte Aktionen zu vergeben. Das kann implementiert werden, indem schlecht erkundeten Situation-Aktion-Paaren ein erhöhter Nutzen zugeteilt wird. Dadurch erhalten wir ein Anfangsverhalten des Agenten, wie wenn in der gesamten Umwelt hohe Belohnungen verteilt wären.

Dazu definieren wir eine Funktion $anz(a, s)$, die die Anzahl wie oft die Aktion a in der Situation s durchgeführt wurde, zurück gibt. Desweiteren sei eine Explorationsfunktion $expl$ gegeben, die bestimmt, ob höherer Nutzen oder selten besuchte Situationen bevorzugt werden. In der im Anschluss aufgeführten Funktionsbeschreibung von $expl$ ist anz_e ein feststehender Parameter und $W^+_{s_t}$ eine optimistische Schätzung des bestmöglichen Wertes (siehe 5.2) der Situation s_t. $W^+_{s_t}$ ist stets größer als die übergebenen Werte von c. Das bewirkt, dass jedes Situation-Aktion-Paar mindestens anz_e mal ausgeführt wird.

$$expl(c, d) = \begin{cases} W^+_{s_t} & wenn\, d < anz_e \\ c & sonst \end{cases}$$

Wie in Abbildung 5 zu sehen ist, kann die Nutzen-

gleichung aus 6.2 nun an das Explorationsproblem angepasst werde. Zu beachten ist, dass nicht mehr die Aktion $\pi(s_t)$ betrachtet und ausgeführt wird. Stattdessen wird die Aktion ausgeführt, die laut der N^+ Funktion (siehe Abbildung 5) den höchsten Nutzen erbringt. Wir haben also einen ADP-Agenten, der nach dem gelernten Nutzen N^+ handelt.

7.2 Q-lernen

Unter Berücksichtigung des Explorationsproblems ist die offensichtlichste Veränderung zum passiven TD-Agenten, dass nun keine feste Strategie mehr verwendet werden kann. Damit der Agent nun eine Situationsvorausschau machen kann, um mit dem gelernten Nutzen entscheiden zu können, was zu tun ist, benötigt er wie schon der ADP-Agent ein Transitionsmodell [Russel, Norvig 2004, S. 941].

An dieser Stelle gelang Watkins 1989 ein entscheidender Durchbruch. Er erfand das sog. Q-Lernen, eine abgewandelte Form des TD-Lernens, bei dem nicht der Nutzen, sondern eine Aktion-Wert-Repräsentation gelernt wird [Sutton, Barto 1998, S. 148]. Dabei wird unter $Q(a_t, s_t)$ der Wert der Ausführung einer Aktion a_t in der Situation s_t verstanden. Ein Agent, der eine Q-Funktion lernt, benötigt kein Transitionsmodell, weder für das Lernen noch für die Ausführung [Russel, Norvig 2004, S. 941]. Wie schon für die Nutzen kann, wie in Abbildung 6 gezeigt, eine Aktualisierungsfunktion für das Q-Lernen gebildet werden.

7.3 Verallgemeinerndes Lernen

Bei den bisher vorgestellten Methoden wurde implizit davon ausgegangen, dass die Nutzenwerte bzw. die Q-Werte für jedes Eingabetupel gespeichert werden. Bei einer kleinen Situationsmenge ist das kein Problem, bei größeren hingegen schlicht unmöglich. Ein Ansatz, dieses Problem zu lösen, ist, eine Funktionsannäherung zu verwenden. Das bedeutet, dass eine Funktion zur Repräsentation gebildet wird, die eine Annäherung an die realen Nutzen- bzw. Q-Werte darstellt.

Eine Möglichkeit wäre, eine gewichtete lineare Funktion

$$\hat{N}_\Theta(s_t) = \Theta_1 f_1(s_t) + \Theta_2 f_2(s_t) + ... + \Theta_m f_m(s_t)$$

zu verwenden, die eine Menge von Basisfunktionen (Merkmale) aufweist. Mit VL-Methoden können dann die Parameter

$$\Theta_1, \Theta_2, ..., \Theta_m$$

erlernt werden und so die Werte der Funktion $\hat{N}_\Theta(s_t)$ dem wahren Nutzen $N(s_t)$ angenähert werden.

Ein solches Vorgehen hat neben der enormen Komprimierung der zu speichernden Darstellung den Effekt, dass von ”erlebten Situationen” auf noch nicht ”erlebte Situationen” geschlossen werden kann. Es findet also eine induktive Verallgemeinerung statt [Russel, Norvig 2004, S. 943].

Leider legt man mit der Wahl der Form der Funktion $\hat{N}_\Theta(s_t)$ den Hypothesenraum, die Menge der möglichen Annäherungsfunktionen, fest. Hierdurch entsteht ein Konflikt zwischen dem Wählen eines möglichst großen Hypothesenraumes - damit die reale Nutzenfunktion sicher enthalten ist - und dem Wählen eines möglichst kleinen Hypothesenraumes - damit die Annäherung möglichst schnell konvergiert [Russel, Norvig 2004, S. 944].

Beschreibt man die Situationen in der 3x4-Welt anhand der Koordinaten (x,y), wäre eine Annäherungsgleichung der Form

$$\hat{N}_\Theta(s_t) = \Theta_0 + \Theta_1 x + \Theta_2 y$$

möglich [Russel, Norvig 2004, S. 944]. Angenommen $(\Theta_0; \Theta_1; \Theta_2)$ liegen bei $(0,5; 0,2; 0,1)$, erhalten wir also eine Schätzung von $0,8$ für $\hat{N}_\Theta(1,1)$. Angenommen bei einem weiteren Versuch bekämen wir aber den Nutzenwert 0,4 für die Situation (1,1), so müssen wir eine Anpassung der Schätzung vornehmen, indem wir die Parameter der Gleichung anpassen. Das geschieht am besten nach jedem Versuch, wobei zu beachten ist, dass durch eine Anpassung der Schätzung für eine Situation die Schätzungen für alle Situationen verändert werden [Russel, Norvig 2004, S. 945]

Um die Schätzung $\hat{N}_\Theta(s_t)$ anzupassen, benötigen wir eine Fehlerfunktion. Hierfür kann

$$N^+(s_t) = b_t + \gamma \underbrace{max}_{a_t} \left[expl \left(\sum_{x=0}^{x=max} T(s_t, a_t, s_{t+1}^x) N^+(s_{t+1}^x) \, , \, anz(a_t, s_t) \right) \right]$$

Abbildung 5: ADP-Funktion unter Beachtung des Explorationsproblems

$$Q(a_t, s_t) = Q(a_t, s_t) + \alpha \left(b_t + \gamma \underbrace{max}_{a_{t+1}} Q(a_{t+1}, s_{t+1}) - Q(a_t, s_t) \right)$$

Abbildung 6: Aktualisierungsfunktion für das Q-Lernen

die sog. Widroff-Hoff-Regel verwendet werden [Russel, Norvig 2004, S. 944]. Der Fehler E berechnet sich dann durch

$$E_j(s) = (\hat{N}_\Theta(s_t) - N_j(s_t))^2/2$$

wobei mit j der Versuch gekennzeichnet ist. Die Änderungsrate des Fehlers bezüglich Θ_i ergibt sich durch die partielle Ableitung

$$\frac{\partial E_i}{\partial \Theta_i}$$

Die Änderung der Parameter Θ_i um den Fehler zu senken, berechnet sich somit durch

$$\Theta_i = \Theta_i + \alpha(N_j(s_t) - \hat{N}(s_t)) \frac{\partial \hat{N}_\Theta(s_t)}{\partial \Theta_i}$$

[Russel, Norvig 2004, S.944]. Der Parameter α begrenzt dabei die Anpassung, um Überanpassung auf eine Situation zu verhindern.

Für die 3x4-Welt erhalten wir somit die drei folgenden Anpassungsfunktionen:

$$\Theta_0 = \Theta_0 + \alpha(N_j(s_t) - \hat{N}_\Theta(s_t))1$$
$$\Theta_1 = \Theta_1 + \alpha(N_j(s_t) - \hat{N}_\Theta(s_t))x$$
$$\Theta_2 = \Theta_2 + \alpha(N_j(s_t) - \hat{N}_\Theta(s_t))y$$

8 Ausblick

Nach einer abgrenzenden Definition von VL wurde im Rahmen dieser Arbeit ein beispielhaftes Szenario eingeführt. Aufbauend auf das vorgestellte formale Interaktionsmodell wurden kritische Teilaspekte der VL-Problematik beleuchtet und die grundlegenden Methoden zur Lösung dieser vorgestellt. Weiter wurde gezeigt, wie grundsätzlich vorgegangen werden kann, um einen Agenten zu befähigen, Nutzen zu erlernen, und wie das Explorationsproblem angegangen werden kann. Außerdem wurde die Methode der Verallgemeinerung beim VL vorgestellt.

Im begrenzten Rahmen dieser Arbeit wurden grundlegende Methoden vorgestellt. Allerding kommen diese in der beschriebenen Weise in der Praxis selten zum Einsatz. Beispielsweise ergibt sich beim verallgemeinernden Lernen das Problem, dass die vorgestellte Methode nur funktioniert, wenn die reale Nutzenfunktion linear ist. Liegt eine Szenario vor, das nicht zu einer linearen Funktion appoximiert, gestaltet sich das verallgemeinernde Lernen sehr viel schwieriger.

Beim verallgemeinernden Lernen sowie den anderen vorgestellten Methoden des VL gibt es außerdem eine vielzahl von Optimierungen bezüglich des Speicherbedarfs sowie der Rechenzeit, die in der Praxis Anwendung finden. Außerdem konnte nicht auf Anpassungen bezüglich nicht oder nur teilweise beobachtbarer Umgebungen eingegangen werden. Dieser Sachverhalt hängt dann auch eng mit der Frage zusammen, ob es besser ist, ein Transitionsmodell und eine Nutzenfunktion oder eine Aktion-Wert-Funktion ohne Transitionsmodell zu lernen.

Meiner Meinung nach stellt das VL für die Robotik und andere Anwendungsgebiete des VL eine sehr interessante und in Zukunft zunehmend relevante Technik dar, zumal moderne, in der Praxis verwendete VL-Methoden zunehmend die vorgestellten grundlegenden VL-Techniken mit Planungsmethode kombinieren. Auch ist das VL ausreichend gereift, um in industriellen Anwendungen Verwendung zu finden.

Literatur

[Kuntz 2005] Reinhard Kuntz. *Intelligente Software-Agenten*. Seminar Kognitionswissenschaft und Intelligente Agenten, 2005

[Lippold 2005] Dietmar Lippold. *Ein Rahmenmodell für das Maschinelle Lernen*. Institut für Intelligente Systeme, Universität Stuttgart, 2005

[Russel, Norvig 2004] Stuart Russel, Peter Norvig. *Künstliche Intelligenz - Ein Moderner Ansatz*. 2. Auflage, Pearson Education Deutschland, 2004

[Sutton, Barto 1998] Richard S. Sutton, Andrew G. Barto. *Reinforcement Learning - An Introduction*. A Bradford Book; MIT Press; Cambridge, Massachusetts, 1998