

Tischfußball: Mensch versus Computer

Thilo Weigel
Atrics GmbH, Freiburg im Breisgau

Bernhard Nebel
Albert-Ludwigs-Universität, Freiburg im Breisgau

28. März 2008

Zusammenfassung Tischfußball ist wesentlich einfacher als richtiger Fußball. Als Domäne für Roboter hat man aber die gleichen Herausforderungen wie in anderen Domänen. Sensoren geben verrauschte Werte, Aktionen müssen unter Zeitdruck ausgewählt werden und ihre Ausführung ist im Allgemeinen nicht perfekt. KiRo und StarKick sind zwei Systeme, die gebaut wurden um Tischfußball gegen Menschen zu spielen. Sie sind die ersten Systeme die existierende, physikalische Spiele gegen Menschen spielen können und dabei auf einem vergleichsweise hohen Niveau spielen. Daneben eignen sie sich um verschiedene KI-Methoden im Kontext zu evaluieren, z.B. Aktionsselektionsmethoden, wie im Papier gezeigt wird.

Abstract Table soccer is much simpler than real soccer. Nevertheless, one faces the same challenges as in all other robotics domains. Sensors are noisy, actions must be selected under time pressure and the execution of actions is often less than perfect. KiRo and StarKick are two systems that have been built to play this game against humans. These systems are interesting because they are the first computerized physical games that are played on a level competitive with experienced humans. Furthermore, these systems enable us to evaluate different AI techniques in context, e.g., action selection methods, as is shown in the paper.

1 Einleitung

Simon und Newell prophezeiten in den Fünfzigern des letzten Jahrhunderts [8], dass Computer in kurzer Zeit intellektuelle Fähigkeiten haben würden, die denen von Menschen vergleichbar wären und dass Computer innerhalb von zehn Jahren in der Lage sein würden, den Schachweltmeister zu schlagen. Der amtierende Schachweltmeister wurde allerdings erst 1997 geschlagen, gut vierzig Jahre nach der Vorhersage.

Obwohl dieser Sieg beeindruckend war, relativiert er sich doch, wenn man andere Fähigkeiten von Computern und Robotern mit denen des Menschen vergleicht. Insbesondere haben Roboter extreme Schwierigkeiten sich in Umgebungen zurechtzufinden, die nicht extra für sie zugeschnitten wurden. Konsequenterweise ist heutzutage in der Künstlichen Intelligenz auch nicht mehr das reine Denken im Fokus. Vielmehr ist intelligentes Handeln innerhalb einer realen, physikalischen Umwelt in den Vordergrund getreten.

Mackworth [6] schlug dann in diesem Sinne Roboterfußball als eine Domäne vor, um die Fähigkeiten von Robotern in dynamischen Umgebungen zu evaluieren. Ende der Neunziger griffen Kitano und andere [5] diese Idee auf und schlugen vor, Roboterfußball als ein TestszENARIO zu definieren und regelmäßig Wettbewerbe und Workshops zu diesem Thema durchzuführen. Die Vision ist dabei:

By the year 2050, develop a team of fully autonomous humanoid robots that can win against the human world soccer champion team.

Schaut man die sich die Wettbewerbe heutzutage an, so finden dort recht ansprechende Spiele statt. Allerdings scheint die Vision, dass Roboter in einem realen, physikalischen Spiel gegen Menschen antreten, noch weit entfernt. Während dies wohl im Falle des Fußballs für längere Zeit so bleiben wird, speziell auch wegen der Verletzungsgefahr, so kann man sich doch bei Spielen wie Tischtennis vorstellen, dass dort Mensch gegen Maschinen antreten können.

Speziell auch für das in Deutschland sehr populäre *Tischfußball* erschien es uns möglich, ein System zu konzipieren, das gegen den Menschen antreten kann. Auf der einen Seite ist der Raum der Beobachtungen und Aktionen so weit eingeschränkt, dass er beherrschbar erscheint. Auf der anderen Seite ergeben sich aufgrund der Dynamik, der Interaktion mit den Gegenspielern und den immer gegebenen Sensor- und Aktorikunsicherheiten genügend interessante Herausforderungen.

Nachdem im Jahre 2000 mit der Entwicklung eines Prototypen für die computerbasierte Steuerung der Stangen eines Kicker-Tisches begonnen worden war,

konnte erstmals im Frühjahr 2002 ein voll funktionsfähiger Prototyp eines autonomen Tischfußball-Roboters der Öffentlichkeit vorgestellt werden. Da das Tischfußballspiel in Deutschland gemeinhin auch als Kicker bekannt ist, wurde dem Roboter der Name *KiRo* für *Kicker-Roboter* gegeben. KiRo wies bereits in der ersten Version ein für Menschen sehr anspruchsvolles Spielniveau auf und wurde in den Medien und bei einer Vielzahl von Ausstellungen im Rahmen von Messen (Hannover-Messe) und Konferenzen (RoboCup) sehr positiv aufgenommen. Um neben der Weiterverfolgung der wissenschaftlichen Interessen auch KiRos Marktpotenzial auszuloten, wurde dieser in enger Zusammenarbeit mit der *Gauselmann AG* bis zur Marktreife weiterentwickelt.

Mit diesem Gerät kann man, ebenso wie gegen einen Schachcomputer, seine Fähigkeiten im Tischfußball messen oder auch trainieren. Bei Tests in Spielhallen war StarKick in der Lage, 85% aller Spiele zu gewinnen [11].

Wissenschaftlich gesehen eignet sich die Domäne und damit auch das Gerät hervorragend, um z.B. verschiedene Arten der Verhaltenskontrolle in nicht-trivialen Umgebungen zu evaluieren [10], Verhaltensadaption zu studieren [14], oder auch verschiedene Arten des Lernens von Fähigkeiten zu entwickeln [17].

2 Hardwarearchitektur

Da in Deutschland Kicker-Tische der Firma *Löwen* sowohl im Freizeit- als auch im Turnierbereich sehr populär sind, wurde für den ersten Prototyp ein *Löwen Home-Soccer* Tisch verwendet. Dieser bietet eine Spielfläche von 1200×680 Millimetern und verfügt über acht Spielerstangen, wobei jeweils vier Stangen auf einer Seite des Tisches bedienbar sind. Insgesamt sind pro Mannschaft elf Spielfiguren auf die Stangen verteilt, so dass jeweils ein Torwart, zwei Verteidiger, fünf Mittelfeldspieler und drei Angreifer gleichzeitig bewegt werden um den Ball zu spielen.

Bei der Entwicklung des ersten KiRo-Prototyps wurde darauf Wert gelegt, dass die Kontrolleinheiten zur Steuerung der Rotation und Translation einer Spielerstange mit wenigen Handgriffen montierbar sind und auch leicht an Tische anderer Hersteller angebracht werden können [13]. Abbildung 1 zeigt ein Modell einer solchen Kontrolleinheit.

Angetrieben werden die Stangen mit *Faulhaber* DC-Servomotoren *3863H024C* in Verbindung mit *Faulhaber* *MCDC2805* Motor-Controllern und *Faulhaber* *IE-512* Inkrementalgebern. Ein im Jahre 2001 handelsüblicher *AMD Athlon* PC mit *700 MHz* und *256 MB RAM* wurde für die Steuerung des Gesamtsystems verwen-

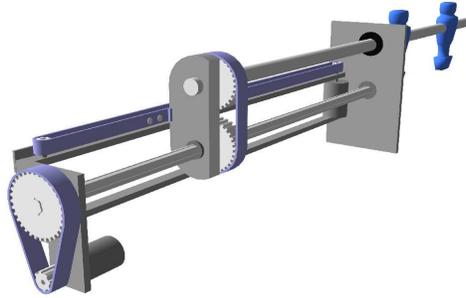


Abbildung 1: Eine Steuereinheit für die Translation und Rotation einer Stange.

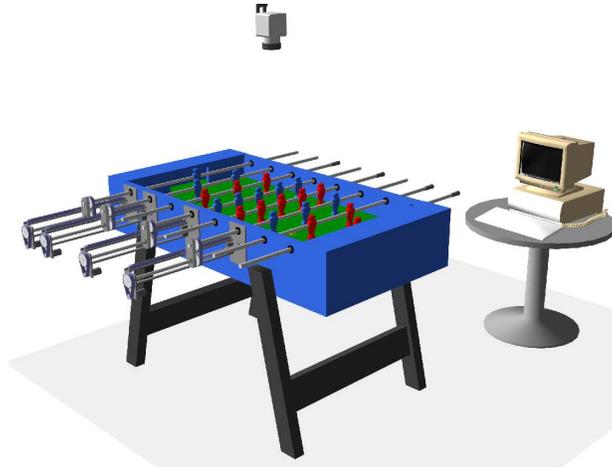


Abbildung 2: Die Hardwarearchitektur des ersten KiRo-Prototyps.

det. Abbildung 2 zeigt die Hardwarearchitektur von KiRo.

Für die Erkennung des Balles und der Spielfiguren wurde eine analoge *Phytec VCAM-110* Farbkamera in Verbindung mit einem *Phytec pciGRABBER-4* Framgrabber eingesetzt, was zu einer Bildrate von 50 Hz (bei halbierten Auflösung von 384×288 Bildpunkten). Mit einem Weitwinkelobjektiv ausgestattet wurde die Kamera so über dem Tisch montiert, dass sie das gesamte Spielfeld überblickt.

Neben der Kamera als Hauptsensor wurden die Informationen der Motor-Encoder verwendet, um die Positionen der von KiRo kontrollierten Stangen alternativ zu erfassen. Aus Sicherheitsgründen wurden zusätzlich Lichtgitter in das System integriert, wobei diese Sensordaten jedoch nicht aktiv für das Spiel verwendet werden.

Um den hohen industriellen Ansprüchen bezüglich Sicherheit, Robustheit und Produktdesign zu genügen, entwickelte die Firma *Gauselmann AG* in Kooperation



Abbildung 3: StarKick – Die kommerzielle Version des KiRo Systems.

mit der Universität Freiburg einen komplett neuen Kicker-Tisch, der mit Hinsicht auf seine Vermarktung speziell für Spiele zwischen einem menschlichen und einem künstlichen Team konzipiert wurde [11, 12]. Abbildung 3 zeigt ein Foto dieses, nun *StarKick* genannten, Tisches. Während der Aufbau des Spielfeldes und die grundsätzlichen Spieleigenschaften identisch zur ersten KiRo-Version sind, wurde die Sensorik und Motorik des Tisches komplett überarbeitet.

Da das bisherige Bildverarbeitungssystem trotz Selbstadaption [15] sehr anfällig für sich verändernde Lichtbedingungen war, der Ball oft von den Spielerstangen und den Spielfiguren verdeckt wurde und das System leicht durch Objekte zwischen Kamera und Spielfläche gestört werden konnte, wurde der neue Tisch als geschlossenes System entworfen und eine infrarotempfindliche Schwarzweiß-Kamera verwendet. Abbildung 4 zeigt das Innere dieses Tisches.

Die Kamera ist nun, wie in Abbildung 5(a) angedeutet, auf der Bodenplatte im Tischinneren montiert und erfasst das Spielfeld über einen Spiegel. Die Haupt-

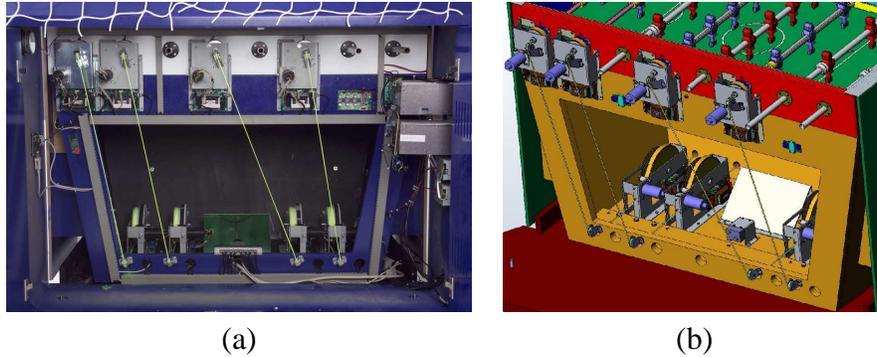


Abbildung 4: Ein Foto (a) und eine CAD Zeichnung (b) des Tischinneren des StarKick-Gerätes.

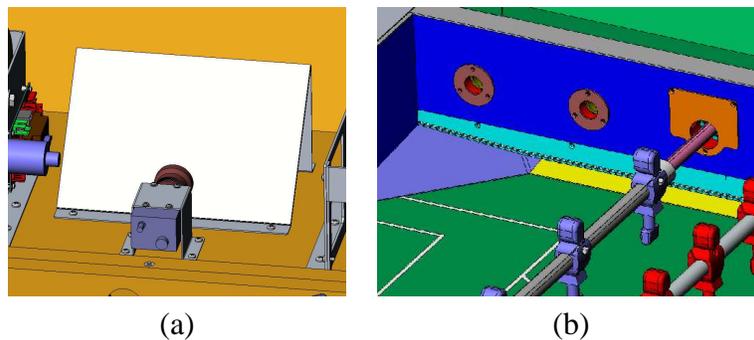


Abbildung 5: Die Komponenten der StarKick-Bildverarbeitung: Kamera und Spiegel (a) sowie die Beleuchtungs-LEDs (b).

Lichtquellen sind ca. 300 Infrarot-LEDs, die direkt über dem Spielfeld entlang der Tischseiten angebracht sind. In Abbildung 5(b) ist die LED-Reihe an einer Tischseite unterhalb der Spielerstangen direkt über dem Spielfeldabschluss in türkis zu sehen.

Die Spielfläche erscheint zwar als grün für den menschlichen Betrachter, ist aber für Licht im Infrarotspektrum durchlässig. Durch diesen Aufbau hebt sich der Ball in aller Regel deutlich von dem sonst dunklen Hintergrund des Kamerabildes ab. Die Spielfiguren können auf diese Weise allerdings nicht erkannt werden, so dass das Bildverarbeitungssystem nur noch zur Erkennung des Balles verwendet wird.

Aus dem Tisch ragen nun lediglich die Stangen zur Kontrolle des menschlichen Teams heraus. Die von KiRo kontrollierten Stangen bestehen aus einem

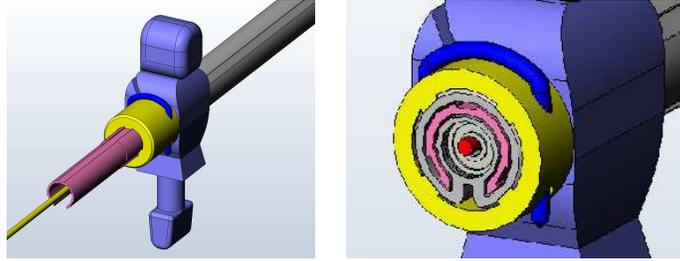


Abbildung 6: Die Struktur einer von KiRo kontrollierten Spielerstange. Das innere Rohr ist pink, das äußere grau dargestellt.

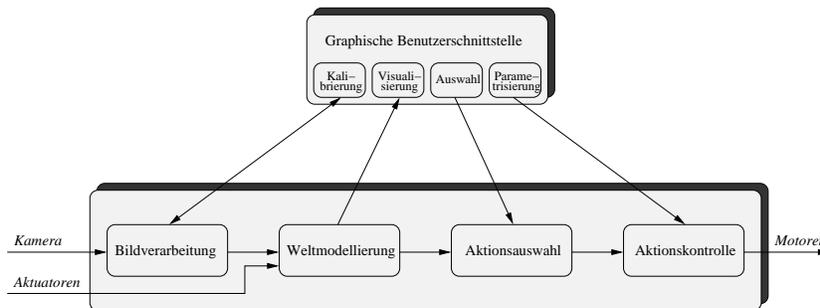


Abbildung 7: Die Softwarearchitektur von KiRo.

inneren und äußeren Profilrohr, wobei die Spielfiguren auf das äußere Rohr geklemmt sind. Abbildung 6 zeigt, wie eine solche Stange aufgebaut ist.

Für den neuen Tisch wurde die gleiche Antriebskombination bestehend aus *Faulhaber* Motoren, Motor-Controllern und Inkrementalgebern verwendet. Zur Steuerung kam ein im Jahre 2003 handelsüblicher *Intel* PC mit 2,4 GHz und 512 MB RAM zum Einsatz.

3 Softwarearchitektur

Softwareseitig werden KiRos Spielerstangen von einer zentralen Kontrollinstanz gesteuert. Diese realisiert eine graphische Benutzerschnittstelle und vollzieht in einem separaten Prozess den eigentlichen Kontrollzyklus, der 50 Mal pro Sekunde durchlaufen wird. Abbildung 7 zeigt schematisch den Aufbau der Kontrollsoftware.

Um eine möglichst kurze Reaktionszeit des Systems zu erreichen, sind *Bild-*

verarbeitung und Handlungskontrolle synchronisiert. Dies bedeutet, dass ein Zyklus in weniger als 20 Millisekunden durchlaufen wird und im Anschluss auf das nächste Kamerabild gewartet wird. Sobald ein neues Kamerabild verfügbar ist, wird dieses ausgewertet und die Positionen der Spielfiguren und die Position des Balles geschätzt.

Diese Schätzungen werden in ein *Weltmodell* integriert, das alle für KiRo relevanten Umweltinformationen bereit hält. Auf Basis der Daten der Kamera und der Motorencoder generiert das Weltmodell zudem zusätzliche Informationen. So werden aufgrund zurückliegender Positionsschätzungen die Bewegungsrichtung und Geschwindigkeit des Balles und der Spielfiguren geschätzt und aufgrund der Encoderdaten Situationen erkannt, in denen der Ball zwischen einer Spielfigur und der Spielfläche eingeklemmt ist.

Basierend auf dem Weltmodell entscheidet die *Aktionsauswahl*, welches Verhalten aus einer Menge möglicher Basisverhalten zu Ausführung kommen soll. Hierzu wurden alternative Verfahren implementiert und evaluiert. Die *Aktionskontrolle* schließlich ist für die Ausführung eines Basisverhaltens verantwortlich. Hierzu werden in jedem Zyklus auf Basis des aktuellen Weltmodells die geeigneten Kommandos für die Motoren berechnet.

4 Aktionsauswahl

Um die Komplexität vieler Aufgaben handhaben zu können, wird die Kontrolle des Verhaltens eines Roboters üblicherweise hierarchisch organisiert. Eine häufige Strukturierung besteht hierbei in der Definition so genannter *Grundfertigkeiten*, die als Kontrollmodule für klar definierte Aktionen jeweils aufgrund der aktuellen Situation geeignete Kommandos für die Aktuatoren berechnen. Ein *Aktionsauswahlverfahren* entscheidet dann, welche Aktion bzw. Aktionen in einer gegebenen Situation zur Ausführung kommen sollen. Insbesondere muss dabei aber berücksichtigt werden, dass einerseits die Sensorik und die Aktuatorik eines Roboters immer mit einer gewissen Unsicherheit und Ungenauigkeit behaftet sind und andererseits die Umgebung sich gewöhnlich auch ohne Zutun des Roboters kontinuierlich verändert. Die Aktionskontrolle sollte also stets auf die neusten Sensorinformationen reagieren und die Aktionsauswahl sollte ständig neu evaluieren, welche Aktion die momentan angemessenste ist.

Es existiert eine Vielzahl von Architekturen, die auf unterschiedliche Art und Weise die *reaktiven* und *deliberativen* Anteile bei der Verhaltenskontrolle eines Roboters kombinieren. Insbesondere in dynamischen Umgebungen wie dem Tisch-

fußballspiel ist ein Vergleich zwischen eher reaktiv oder eher deliberativ ausgerichteten Aktionsauswahlverfahren sehr aufschlussreich. Die Verhaltenskontrolle beim Tischfußballspiel wurde daher sowohl mittels eines Entscheidungsbaumes [13] als auch mittels entscheidungstheoretischen Planens [10] realisiert. Im Folgenden werden nach einer Beschreibung der zur Verfügung stehenden Grundfertigkeiten beide Verfahren vorgestellt und verglichen.

Während eines Spieles verfügen KiRo und StarKick über eine Reihe von Möglichkeiten, in das Spielgeschehen einzugreifen. Hierbei wird zwischen den folgenden grundlegenden Aktionen unterschieden:

- Die **DefaultAction** verschiebt und rotiert eine Stange so, dass die Spielfiguren bezüglich des Tisches zentriert sind und nach unten zeigen.
- **KickBall** verschiebt eine Stange so, dass eine Spielfigur auf gleicher Höhe mit dem Ball ist und rotiert die Stange dann in Spielrichtung, um den Ball zu schießen.
- **MoveKickBall** verschiebt eine Stange mit eingeklemmten Ball so, dass dieser seitlich wegrollt und rotiert die Stange nach einer kurzen Verzögerung nach vorne, um den Ball zu schießen.
- **BlockBall** verschiebt und rotiert eine Stange so, dass eine der Spielfiguren den Ball an der Position abfängt, an der er die Stange voraussichtlich passieren wird.
- **ClearBall** verschiebt eine Stange so, dass sich eine Spielfigur an der Position befindet, an der der von hinten heranrollende Ball erwartet wird. Dabei wird die Stange in Horizontalstellung rotiert, um den Ball für einen späteren Schuss durch *KickBall* zunächst passieren zu lassen.
- **BlockAtPos** verschiebt und rotiert eine Stange so, dass sie eine bestimmte Position blockiert.
- **ClearAtPos** verschiebt eine Stange zu einer bestimmten Position und rotiert die Stange in Horizontalstellung, um den von hinten heranrollenden Ball passieren zu lassen.
- **StopBall** verschiebt eine Stange so, dass sich eine Spielfigur an der Position befindet, an der der heranrollende Ball erwartet wird und hält dabei die Stange in horizontaler Stellung bis durch eine schnelle Rotation der heranrollende Ball zwischen der Spielfigur und dem Spielfeld eingeklemmt wird.

Da alle Stangen außer dem Torwart über mehr als eine Spielfigur verfügen, muss in der Regel entschieden werden, mit welcher Figur eine Aktion tatsächlich ausgeführt werden soll. Da sich jede Spielfigur nur innerhalb eines gewissen Bereiches bewegen kann und sich die Bereiche benachbarter Figuren überlappen können, entstehen oft Situationen, in denen eine Zielposition von zwei Spielfiguren erreicht werden kann. In diesen Fällen wird diejenige Spielfigur ausgewählt, die sich einerseits so nahe wie möglich an der Zielposition befindet und andererseits an der Zielposition noch über so viel wie möglich Bewegungsspielraum verfügt, wobei stets diejenige Figur bevorzugt wird, die im letzten Zyklus ausgewählt wurde.

5 Aktionsauswahl durch einen Entscheidungsbaum

Für die Aktionsauswahl von KiRo und StarKick wurde zunächst ein einfacher binärer Entscheidungsbaum verwendet, bei dem an jedem Knoten jeweils eine *ja/nein* Entscheidung getroffen wird [13]. Abbildung 8 zeigt den Entscheidungsbaum, so wie er für das kommerzielle StarKick-Gerät verwendet wird. Die Aktion *StopBall* zeigte in Experimenten zwar eine hohe Erfolgsquote [16], birgt aber dennoch ein größeres Risiko, den Ball zu verfehlen und wird daher in den aktuellen Geräten nicht eingesetzt. Der Entscheidungsbaum wählt für jede Spielerstange auf die gleiche Art und Weise eine Aktion aus.

Falls die Ballposition unbekannt ist, bleibt dem System nur die *DefaultAction*. Ist der Ball hingegen sichtbar, so wird zunächst überprüft, ob sich der Ball in unmittelbarer Nähe einer Spielfigur befindet und daher mit *KickBall* geschossen werden kann. Ist dies nicht der Fall, so wird ermittelt, ob sich der Ball in Spielrichtung vor der Spielerstange befindet. Trifft dies zu, so wird weiter geprüft, ob die Stange den Ball direkt mittels *BlockBall* blocken muss, oder ob eine andere, dem Ball nähere Stange dies bereits tut und daher mittels *BlockAtPos* eine gute, seitlich versetzte Verteidigungsposition eingenommen werden soll. Befindet sich der Ball hinter der Stange, so wird eine *Clear*-Aktion ausgeführt. Um ein hektisches Spielverhalten zu vermeiden, wird der Ball hierbei nur dann antizipiert, wenn er sich relativ nahe bei der Spielerstange befindet.

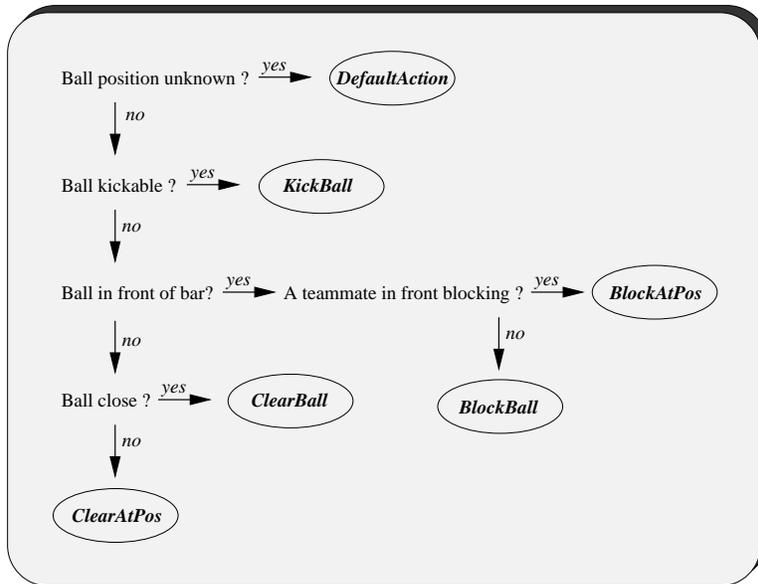


Abbildung 8: Der Entscheidungsbaum zur Verhaltenskontrolle von KiRo.

6 Aktionsauswahl durch entscheidungstheoretisches Planen

Eine Alternative zum Gebrauch eines Entscheidungsbaumes zur Aktionsauswahl liegt in der Verwendung entscheidungstheoretischen Planens. Hierbei wird die Unsicherheit der Aktionsausführung sowie der Nutzen eines Zustandes mit in die Auswahlentscheidung einbezogen, wobei man die Folgen der eigenen Aktionen soweit wie möglich antizipiert.

Um über einen möglichst feinauflösenden Zustandsraum zu verfügen und dennoch die Anzahl der Zustände handhaben zu können, wurde eine Form des *Real Time Dynamic Programming* verwendet, bei dem stets nur ein *partieller Suchbaum* erzeugt und evaluiert wird [4]. Hierzu wird der Zustandsraum nur implizit definiert und für jeden Planungsvorgang der relevante Ausschnitt des Zustandsraumes mittels einer Simulation dynamisch generiert.

Befindet sich ein Team nicht in Ballbesitz, so kann es nur auf gegnerische Aktionen reagieren.. In der Verteidigung findet daher für alle Spielerstangen eine rein reaktive Verhaltenskontrolle statt. Geplant wird immer nur dann, wenn sich der Ball bei einer Spielerstange der eigenen Mannschaft befindet, wobei Aktionssequenzen immer spätestens bei eigenem Torerfolg oder gegnerischem Ballbesitz

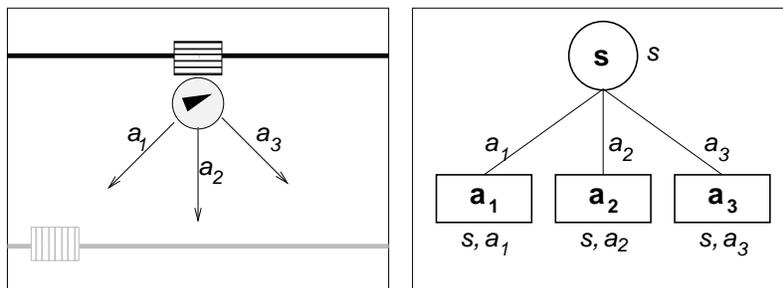


Abbildung 9: Die Erzeugung dreier Aktionsknoten. Der Ball ist durch einen Kreis gekennzeichnet, seine Bewegungsrichtung durch einen Pfeil.

enden. Da immer nur höchstens eine Spielerstange Einfluss auf den Ball nehmen kann, wird auch nur für diese aktiv geplant. zugewiesen.

Der Zustandsraum wird in Form eines Suchbaumes erzeugt. Ausgangspunkt ist jeweils ein mit einem Spielzustand s korrespondierender *Zustandsknoten* s . Aus der Menge A aller KiRo zu Verfügung stehenden Aktionen werden nun zunächst alle in s ausführbaren Aktionen $\{a_1, a_2, \dots, a_n\} \subseteq A$ bestimmt und für jede dieser Aktionen ein korrespondierender *Aktionsknoten* a_i erzeugt. Da diese Knoten verschiedene Handlungsalternativen in Zustand s repräsentieren, ergibt sich der Nutzen U des Zustandsknotens s als der maximale Nutzen der Nachfolgeknoten: $U(s) = \max_{0 < i \leq n} U(a_i)$.

Abbildung 9 illustriert diesen Vorgang anhand einer Spielsituation s , in der der Ball diagonal von links auf eine eigene Spielfigur zurollt. In dieser Situation hat der Spieler die Wahl, entweder geradeaus oder schräg nach links bzw. rechts zu schießen. Demgemäß werden drei Aktionsknoten erzeugt.

Die möglichen Reaktionen des Gegners werden als gleichzeitig mit den eigenen Aktionen stattfindend und mit einer gewissen Eintretenswahrscheinlichkeit assoziierte *exogene Ereignisse* modelliert¹. Hierzu wird basierend auf dem Weltzustand s eine Menge von potenzielle Aktionen des Gegners $\{o_j\}$ betrachtet, die Auftretenswahrscheinlichkeiten $p(o_j)$ haben. Hierbei gehen wir vereinfachend davon aus, dass der Gegner jeweils mit einer Wahrscheinlichkeit von 0,5 die Stange entweder nicht bewegt oder sie entsprechend dem *BlockBall*-Verhalten positioniert. Natürlich wären hier gelernte Gegnermodelle von Vorteil.

Da die Reaktionen des Gegners als unabhängig von den eigenen Aktionen

¹Diese Abstraktion ist für das Tischfußballspiel durchaus adäquat, da sich die Spieler in der Regel tatsächlich gleichzeitig bewegen und eine Reaktion auf eine Aktion des Gegenspielers nur in Ausnahmefällen, wie zum Beispiel bei sehr langsamen Bällen möglich ist.

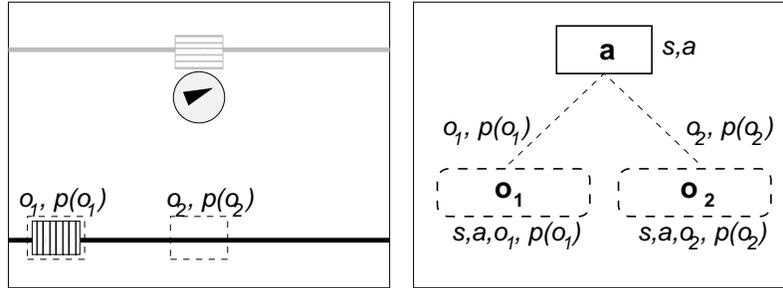


Abbildung 10: Die Erzeugung zweier Gegnerknoten.

angesehen werden, wird für jeden Aktionsknoten a_i eine Menge von *Gegnerknoten* erzeugt. Der Nutzen eines Knotens a_i ist der erwartete Nutzen seiner Nachfolgerknoten: $U(a_i) = \sum_{j=1}^m p(o_j)U(o_{ij})$. Abbildung 10 zeigt zwei mögliche Reaktionen des Gegners in der Spielsituation s : Mit einer gewissen Wahrscheinlichkeit $p(o_1)$ belässt er die Spielerstange an ihrer aktuellen Position und mit einer gewissen Wahrscheinlichkeit $p(o_2)$ zieht er einen geraden Schuss antizipierend auf gleiche Höhe mit der ballführenden Spielfigur.

Jeder Gegnerknoten enthält somit Informationen über den Weltzustand s , eine mögliche eigene Aktion a_i und eine mögliche Reaktion des Gegners o_j . Hierauf basierend werden potenzielle Konsequenzen als eine Menge $\{c_k\}$ mit Auftretenswahrscheinlichkeiten $p(c_k)$ betrachtet. Dabei haben wir als Konsequenzen nur Erfolg und Misserfolg der Aktion betrachtet, wobei die Erfolgswahrscheinlichkeiten $p(c_k)$ aufgrund von beobachteten Daten aus Spielen geschätzt werden.

Auf Basis jedes Tupels $\langle s, a_i, o_j, c_k \rangle$ kann nun jeweils ein neuer Zustand s_{ijk} mit korrespondierendem Zustandsknoten s_{ijk} per Simulation erzeugt werden. Hierbei wird die Evolution des Weltzustandes immer so weit simuliert, bis einer der Spieler wieder den Ball manipulieren kann. In der Regel wird also immer der Übergang von einer Spielerstange zu der nächsten simuliert. Um die Echtzeitanforderungen des zu erfüllen wurde dabei ein grobes, dafür aber effizienteres Simulationsverfahren eingesetzt [9].

Falls die Simulation ergibt, dass sich wieder eine eigene Spielerstange in Ballbesitz befindet, so ist der Zustandsknoten s_{ijk} Ausgangspunkt für eine weitere Aktion des eigenen Teams und der Nutzen berechnet sich in Abhängigkeit weiterer antizipierter Aktionen, Reaktionen und Konsequenzen. Endet die simulierte Ballbewegung hingegen bei einer gegnerischen Stange, so wird dieser Pfad nicht weiter verfolgt und der Nutzen von s_{ijk} basierend auf einer Heuristik geschätzt.

In dynamischen Umgebungen wie dem Tischfußballspiel muss in der Regel

innerhalb weniger Millisekunden eine der Spielsituation angemessene Aktion ausgewählt werden. Statt einen Plan bis zu einem weit in der Zukunft liegenden Ziel zu entwerfen, ist es daher sinnvoller, die Rechenzeit zu begrenzen und nur mit einem begrenzten Horizont in die Zukunft zu planen.

Im Allgemeinen sind nur die Nutzenwerte der Zielzustände (gefallenes Tor) a priori gegeben sind. Die Nutzenwerte von Zuständen, zu denen bei einem begrenzten Horizont geplant wird, müssen deshalb geschätzt werden. Hierzu wird eine *Nutzenfunktion* erstellt, die einen Spielzustand auf einen reellen Wert abbildet. Die Nutzenwerte der Zustände werden dann entsprechend einer *Rollback Prozedur* [4] berechnet. Der Nutzen eines Zustandes s ist hierbei analog zum *Value Iteration Verfahren* jeweils der maximal zu erwartende Nutzen bezüglich aller in diesem Zustand ausführbaren Aktionen a_i .

Die heuristische Schätzung der Nutzenwerte basiert auf der Position, Richtung und Geschwindigkeit des Balles. Grundsätzlich ist der geschätzte Nutzen umso höher, umso weiter der Ball von den Seitenbanden entfernt ist und umso näher er sich beim gegnerischen Tor befindet. Je langsamer sich der Ball bewegt, desto leichter kann der Ball unter Kontrolle gebracht werden. Daher werden Positionen unter den eigenen Stangen umso stärker bevorzugt und Positionen unter gegnerischen Stangen umso stärker benachteiligt, umso langsamer der Ball sich bewegt. Da schnelle Bällen wiederum die Torgefahr erhöhen, werden Positionen, von denen der Ball in das gegnerische Tor rollen kann, desto stärker bevorzugt, je schneller sich der Ball bewegt. Da die Positionen der Spielfiguren nicht vorhergesehen werden können, werden sie bei der Bewertung vernachlässigt.

In einem abschließenden Test wurden die beiden für KiRo bzw. StarKick implementierten Verfahren zur Aktionsauswahl miteinander verglichen. Um die Leistungsfähigkeit der beiden Verfahren miteinander zu vergleichen, wurden sowohl in der Simulation als auch auf dem realen KiRo-Gerät Testspiele durchgeführt. Die Simulation wurde verwendet, da sie die Möglichkeit bietet, beide Verfahren direkt gegeneinander antreten zu lassen.

In der simulationsbasierten Evaluation wurden 20 Partien bis jeweils zu einem Spielstand von 9 Toren gespielt. Für das entscheidungstheoretische Planen wurde dabei stets eine maximale Suchtiefe von 2 verwendet. Bei einem Vergleich der durchschnittlichen Anzahl der Tore, die ein Team pro Minute erzielt hatte, erwies sich das entscheidungstheoretische Planen mit 0,7 Toren pro Minute gegenüber dem entscheidungsbaumbasierten Verfahren als deutlich überlegen, das nur 0,1 Tore pro Minute erzielte. Als weiteres Vergleichskriterium wurde die *Feldüberlegenheit* eines Teams betrachtet. Dieser Wert gibt an, wie lange sich der Ball relativ zu der Gesamtspielzeit in der gegnerischen Spielhälfte befand. Auch

lag das entscheidungstheoretische Verfahren vorne, da es bei Suchtiefe 2 mit immerhin 72% deutliche Feldüberlegenheit zeigte.

Bei einer Evaluation auf dem realen Tisch spielten mehrere, jeweils aus zwei menschliche Spielern bestehende Teams gegen KiRo. Hier konnte das entscheidungstheoretische Verfahren allerdings nicht punkten. Die reaktive Variante erzielte ein leicht höhere Torrate.

Dabei blieben aber weitere Kriterien wie die Eleganz des Spieles oder auch der Unterhaltungswert für Gegenspieler und Zuschauer bleiben unberücksichtigt. Hier hat ein planendes Verfahren deutlich mehr Potenzial, unter Einsatz komplexerer Aktionen intelligent wirkende Spielzüge zu erzeugen.

7 Verwandte Arbeiten

Nach gegenwärtigem Kenntnisstand existiert kein mit KiRo oder StarKick vergleichbares System, das in der Lage ist, gegen Menschen autonom Tischfußball zu spielen.

Allerdings wurden auch andere Sportspiele als Basis für die Grundlagenforschung verwendet. Anderson entwickelte ein Robotiksystem, das in der Lage ist, eine vereinfachte Version von Tischtennis zu spielen. Der Ball wird hierbei mittels zweier Stereo-Kamerasysteme erfasst und seine Trajektorie mittels Kurvenfitting approximiert. Zur Bewegung der Aktuatoren wählt ein Expertensystem dann aus einer Menge von Templates einen Plan aus, der kontinuierlich bezüglich neuer Sensorinformationen verfeinert wird [1].

Bentivegna *et al.* ließen einen humanoiden Roboter durch Beobachtung und Imitation menschlicher Spieler das Air-Hockey Spiel erlernen. Die für die Aktionsausführung relevanten Parameter werden unter anderem mittels einer physikalischen Simulation ermittelt [2].

Auch Bishop *et al.* schlugen das Air-Hockey-Spiel als Testdomäne für die Forschung im Bereich intelligenter Robotiksysteme vor. Sie entwickelten hierfür einen speziellen Roboterarm und verwendeten zur Erkennung des Pucks eine über dem Air-Hockey-Tisch angebrachte Overheadkamera [3].

Moore *et al.* entwickelten ein Robotiksystem für das Billardspiel und evaluierten damit verschiedene Varianten fallbasierten Lernens [7].

8 Zusammenfassung und Ausblick

Fußball kann eine wichtige Rolle bei der Forschung in der *Künstlichen Intelligenz* spielen. Wie die *RoboCup*-Symposien und Wettbewerbe zeigen, begeistert ein gemeinsame Anwendungsszenario und kann auch publikumswirksam Forschungsfragen publik machen. Allerdings sind wir hier noch sehr weit von der direkten Interaktion zwischen Menschen und Robotern entfernt.

Beim Tischfußball sind wir hier einen Schritt weiter. Das von uns mitentwickelte, kommerziell erhältliche Spielgerät StarKick ist in der Lage, auf hohem Niveau gegen menschliche Gegner zu spielen und auch meist zu gewinnen. Neben dem Unterhaltungsaspekt und dem möglichen Trainingspartner für professionelle Sportler bietet das Gerät für uns die Möglichkeit, Forschungsansätze zur physischen Interaktion zwischen Mensch und Maschine zu entwickeln und zu evaluieren. Unsere Schwerpunkte lagen dabei bisher auf den Gebieten der adaptiven Bildverarbeitung, der Verhaltenskontrolle und Verhaltensadaption. In Zukunft wollen wir vor allen Dingen auf der Seite des Lernens von Fähigkeiten durch Entdecken und Imitieren sowie der Online-Verhaltensanpassung arbeiten. Langfristig besteht natürlich die Hoffnung eines Tages auf dem Niveau von Ligaspielern zu spielen.

Literatur

- [1] R. Anderson. Dynamic Sensing in a Ping-Pong Playing Robot. *IEEE Transactions on Robotics and Automation*, 5(6):723–739, 1989.
- [2] D. Bentivegna and C. Atkeson. Learning From Observations Using Primitives. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1988–1993, Seoul, Korea, 2001.
- [3] B. Bishop, P. Shirkey, and M. Spong. An Experimental Testbed for Intelligent Control. In *American Control Conference*, pages 1328–1332, Seattle, USA, 1995.
- [4] C. Boutilier, T. Dean, and S. Hanks. Decision-Theoretic Planning: Structural Assumptions and Computational Leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.
- [5] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara. RoboCup: A Challenge Problem for AI. *AI Magazine*, 18(1):73–85, 1997.

- [6] A. Mackworth. On Seeing Robots. In A. Basu and X. Li, editors, *Computer Vision: Systems, Theory, and Applications*, pages 1–13. World Scientific Press, Singapur, 1993.
- [7] A. Moore, D. Hill, and M. Johnson. An Empirical Investigation of Brute Force to Choose Features, Smoothers and Function Approximators. In *Computational Learning Theory and Natural Learning Systems*, volume III: Selecting Good Models, pages 361–379. MIT Press, 1995.
- [8] H. A. Simon and A. Newell. Heuristic problem solving: The next advance in operations research. *Operations Research*, 6:1–10, 1958.
- [9] M. Tacke. MITRA: Aktionsauswahl im Tischfußball. Diplomarbeit, Albert-Ludwigs-Universität Freiburg, Institut für Informatik, 2003.
- [10] M. Tacke, T. Weigel, and B. Nebel. Decision-Theoretic Planning for Playing Table Soccer. In *Proceedings of the 27th German Conference on Artificial Intelligence*, pages 213–225, Ulm, Deutschland, 2004.
- [11] T. Weigel. KiRo – A Table Soccer Robot Ready for Market. *Künstliche Intelligenz*, (1):50–55, 2005.
- [12] T. Weigel. KiRo – A Table Soccer Robot Ready for Market. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4277–4282, Barcelona, Spanien, 2005.
- [13] T. Weigel and B. Nebel. KiRo – An Autonomous Table Soccer Player. In G. Kaminka, P. Lima, and R. Rojas, editors, *RoboCup-2002: Robot Soccer World Cup VI*, Lecture Notes in Artificial Intelligence, pages 119–127. Springer-Verlag, Berlin, Heidelberg, New York, 2003.
- [14] T. Weigel, K. Rechert, and B. Nebel. Behavior Recognition and Opponent Modeling for Adaptive Table Soccer Playing. In *Proceedings of the 28th German Conference on Artificial Intelligence*, pages 344–359, Koblenz, Deutschland, 2005.
- [15] T. Weigel, D. Zhang, K. Rechert, and B. Nebel. Adaptive Vision for Playing Table Soccer. In *Proceedings of the 27th German Conference on Artificial Intelligence*, pages 424–438, Ulm, Deutschland, 2004.

- [16] D. Zhang. Action Selection and Action Control for Playing Table Soccer Using Markov Decision Processes. Master Thesis, Albert-Ludwigs-Universität Freiburg, Institut für Informatik, 2005.
- [17] D. Zhang and B. Nebel. Teaching a table soccer robot a new action sequence by observing and imitating. In *Proceedings of the Third Artificial Intelligence for Interactive Digital Entertainment Conference (AIIDE 2007)*, pages 61–67.