

Künstliche Intelligenz für Computerspiele

Historische Entwicklung und aktuelle Trends

Christian Bauckhage · Kristian Kersting
Christian Thureau

Einleitung

Computer- und Videospiele haben sich als ein fester Bestandteil unserer Alltagskultur etabliert. Innerhalb weniger Jahrzehnte ist eine globale Industrie entstanden, deren jährliche Milliardenumsätze mittlerweile die Ergebnisse der Filmbranche übertreffen und erwartbar weiter wachsen werden [15]. Bedenkt man, dass Computerspiele ein vergleichsweise neues Medium darstellen, lässt sich ihr Erfolg nur dadurch erklären, dass sie tiefe Bedürfnisse vieler Menschen ansprechen und faszinierend genug sind, Zeit und Geld dafür zu investieren.

Tatsächlich erlaubt es moderne Hardware, atmosphärisch dichte und immersive Spielwelten zu programmieren, die derart komplex sind, dass sie von Spielern als alternative Realitäten wahrgenommen werden, die es zu erkunden und zu meistern gilt. In populären Genres wie Action-Spiele, Massively Multiplayer Online Role Playing Games (MMORPGs) oder Sportsimulationen übernehmen Spieler die Rolle virtueller Charaktere, die alleine oder im Team Aufgaben lösen müssen, die mit zunehmender Erfahrung immer anspruchsvoller werden. Zudem bieten mobile Spiele, die auf Smartphones oder Tablet-Computern gespielt werden, oder Spiele in Web-basierten sozialen Netzwerken neue Möglichkeiten, das Spielgeschehen in den Alltag zu integrieren und neue Formen gemeinsamen Spielens zu erleben.

Neben technischen Herausforderungen in Bezug auf die Computergrafik oder die Simulation physikalischer Gesetze spielt die Künstliche Intelligenz (KI) bei der Entwicklung moderner Computerspiele eine besondere Rolle. Dies betrifft in erster Linie Fragen der Modellierung und Erkennung mensch-

lichen Verhaltens. Um kommerziell erfolgreich zu sein, müssen Computerspiele Spaß machen. Dabei hängt Spielspaß nicht nur von realistischer Grafik oder überzeugend simulierter Physik ab, sondern geht ganz wesentlich mit der Schwierigkeit der zu lösenden Aufgaben einher. Für interaktive Spiele, die alleine oder im Team gegen den Computer gespielt werden, bedeutet dies, dass computergesteuerte Gegenspieler weder durchschaubar oder vorhersehbar agieren sollen, noch über übermenschliche Spielstärke verfügen dürfen, da beides bei menschlichen Spielern schnell zu Langeweile oder Frustration führt [7]. Gleiches gilt für Spiele, in denen menschliche Spieler über das Internet gegeneinander antreten. Hier stellt sich für die KI die Herausforderung, automatisch zu ermitteln, wie gut einzelne Spieler sind und dafür zu sorgen, dass sie automatisch mit Servern verbunden werden, auf denen Spieler mit vergleichbaren Fähigkeiten spielen.

Dieser Artikel zeichnet die Entwicklung der KI für Computerspiele nach und benennt aktuelle Herausforderungen für Forschung und Entwicklung. Die Darstellung gliedert sich dabei in zwei Teile, in denen traditionelle und neue Fragestellungen betrachtet werden.

KI für klassische Spielszenarien

Lange Jahre konzentrierte sich die Entwicklung von KI-Technologien für Computerspiele vor allem

DOI 10.1007/s00287-014-0822-4
© Springer-Verlag Berlin Heidelberg 2014

Christian Bauckhage · Kristian Kersting · Christian Thureau
Universität Bonn, Bonn; Fraunhofer IAIS, Sankt Augustin;
TU-Dortmund, Dortmund; Gameanalytics, Kopenhagen
E-Mail: {christian.bauckhage, kristian.kersting,
christian.thureau}@iais.fraunhofer.de

Zusammenfassung

Die technische Entwicklung von Computerspielen und die Entwicklung von Methoden der Künstlichen Intelligenz (KI) gehen seit Jahrzehnten Hand in Hand. Spektakuläre Erfolge der KI in Spieleszenarien sind etwa der Sieg des Schachcomputers Deep Blue über den damaligen Weltmeister Gary Kasparow im Jahr 1997 oder der Gewinn der Quizshow Jeopardy durch das Programm Watson im Jahr 2010. Standen lange Zeit Fragen zur Implementierung möglichst intelligenter und glaubwürdiger künstlicher Spieler im Vordergrund, ergeben sich durch aktuelle Entwicklungen in den Bereichen mobile- und social gaming neue Problemstellungen für die KI. Dieser Artikel beleuchtet die historische Entwicklung der KI in Computerspielen und diskutiert die Herausforderungen, die sich in modernen Spieleszenarien ergeben.

darauf, überzeugende Verhaltensweisen für computergesteuerte Spieler (engl. *Non-Player Characters*, NPCs) zu realisieren. Dieser Abschnitt fasst die Entwicklung zusammen, die das Feld dabei nahm.

KI für Spiele mit vollkommener Information

Spiele, bei denen Spieler über sämtliche Aktionen oder Züge aller Mitspieler informiert sind, heißen Spiele mit vollkommener Information (*perfect information games*). Gängige Beispiele sind klassische 2-Personen-Spiele wie Schach, Dame, Othello oder Go. Da sie sich leicht formalisieren lassen, standen derartige Spiele lange im Fokus der KI Forschung. Ihr Zustandsraum, d. h. die Menge aller möglichen Spielzustände, die sich aus Spielzügen ergeben können, kann mittels diskreter Strukturen, insbesondere in Form von Suchbäumen, modelliert werden. Dies wurde erstmals von Claude E. Shannon beschrieben, der bereits 1950 einen Artikel veröffentlichte, worin er Ansätze zur Programmierung eines Schachcomputers vorschlug [18].

Shannon schlug im Wesentlichen zwei Strategien zur Planung intelligenter Züge vor. Sein erster Vorschlag bestand darin, zu einem gegebenen Spielstand alle Konfigurationen zu berechnen, die nach drei abwechselnden Zügen denkbar sind. Da es im Schach zu jedem Spielstand nur abzählbar viele erlaubte Folgezüge gibt, lässt sich ein solcher *look-ahead* in Form eines Suchbaums repräsentieren (siehe Abb. 1). Ein mittelfristig erfolgsversprechender Zug ergibt sich dann aus einer Betrachtung der

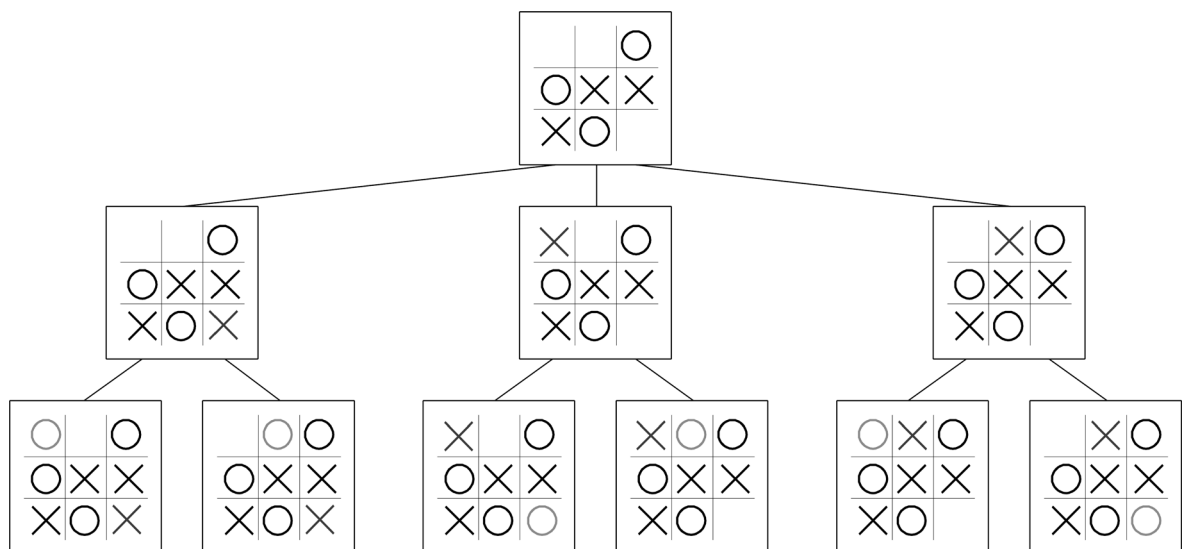


Abb. 1 Expansion eines Suchbaums am Beispiel des Spiels tic-tac-toe. Die Wurzel des Baumes charakterisiert den aktuellen Spielzustand; die Knoten der nächsten Ebene des Baumes modellieren, welche Zustände sich aus den drei Zügen ergäben, die für Spieler X in der aktuellen Situation möglich sind. Im nächsten Zug verbleiben für Spieler O jeweils zwei Möglichkeiten, deren Konsequenzen in der untersten Ebene des Baumes dargestellt sind. Das Problem besteht für Spieler X nun darin, einen Zweig des Suchbaums zu identifizieren, der zu einer günstigen Konfiguration führt, und entsprechend zu ziehen

Kosten der erzeugten Suchbaumknoten. Dazu werden Kostenfunktionen definiert, die einerseits auf die Minimierung möglicher Verluste und andererseits die Maximierung möglicher Gewinne abzielen. Shannon erkannte allerdings sofort, dass dieser *min-max lookahead* Algorithmus aufgrund des erwartbar exponentiellen Wachstums des Suchbaums und des daraus resultierenden Rechenaufwandes nicht wirklich praktikabel ist. Alternativ schlug er daher vor, Heuristiken zu entwickeln, die garantieren, dass nur solche Suchbaumknoten expandiert werden, die nicht zu völlig unbrauchbaren Konfigurationen führen. Da dadurch in jeder Ebene des Baumes nur noch wenige Nachfolgeknoten entstehen, wird es gleichzeitig möglich, deutlich mehr als drei Folgezüge zu explorieren.

Mit Shannons Arbeit war der wesentliche Grundstein zur Entwicklung von KI-Techniken für Spiele mit vollkommener Information gelegt. Praktisch alle folgenden Arbeiten und Systeme widmeten sich der Entwicklung geeigneter Kostenfunktionen oder Suchheuristiken. Zu den bekannten und erfolgreichen Ansätzen zählen etwa das *alpha-beta-pruning* oder der *A** Algorithmus, die heute zum Kanon der KI-Methoden zählen [13, 16]. DeepBlue, der Schachcomputer von IBM, der im Jahr 1997 schließlich den damaligen Weltmeister Gary Kasparow schlug, nutzte eine Kombination aus purer Rechenkraft und ausgeklügelten Suchheuristiken. Das System war in der Lage, pro Sekunde etwa 100 Millionen Stellungen zu bewerten und konnte damit Zugfolgen von bis zu elf Zügen testen.

Ein Beispiel dafür, dass es sich lohnt, nicht nur Suchheuristiken sondern auch Bewertungsfunktionen zu optimieren, findet sich beim Spiel Othello. Das System Logistello [6], das 1997 erstmals gegen den amtierenden menschlichen Weltmeister gewann, ist auch deshalb interessant, weil es ein frühes Beispiel für den erfolgreichen Einsatz von Techniken des Maschinellen Lernens (ML) in Computerspielen darstellt: Durch wiederholte Spiele gegen sich selbst, lernte das System, welche Bewertungsfunktionen die zuverlässigste Analyse eines Spielzustandes erlauben.

Insgesamt lässt sich nach 60 Jahren Forschung und Entwicklung festhalten, dass die Programmierung intelligenter Gegenspieler für Spiele mit vollständiger Information eine Erfolgsgeschichte der KI darstellt und als ein weitestgehend gelöstes Problem betrachtet werden kann. Eine Herausforderung

bildet jedoch nach wie vor das Spiel Go. Während KI-Algorithmen für Dame, Othello oder Schach mittlerweile nicht mehr auf Spezial-Hardware angewiesen sind, um gegen menschliche Weltmeister zu gewinnen, gelangen Siege gegen Go Großmeister bisher nur, wenn etwa das Spielfeld deutlich verkleinert wird. Aufgrund der Dimension des eigentlichen Spielfeldes (19×19 Felder) und der kaum vorhandenen Beschränkung möglicher Züge ist der Suchraum bei Go immer noch zu groß, als dass er mit gängigen Computern (Speicher und Prozessor) ausreichend tief exploriert werden könnte. In den letzten Jahren hat sich jedoch gezeigt, dass sowohl probabilistische Suchverfahren, in denen Monte-Carlo-Methoden eingesetzt werden, die Zustandsräume nur in erwartbar aussichtreiche Richtungen durchsuchen, als auch ML-Algorithmen, die sinnvolle Zugfolgen anhand von Beispielen lernen, vielversprechende Ansätze für Computer Go bieten.

KI für Spiele mit unvollständiger Information

Ebenso wie Go, dessen Suchraum extrem groß ist, stellen auch Spiele mit unvollständiger Information (*imperfect information games*) noch immer eine Herausforderung für die KI dar [17]. Als Spiele mit unvollständiger Information bezeichnet man dabei Spiele, deren aktueller Spielzustand nicht sämtlichen Spielern eindeutig bekannt ist, etwa wie im Falle von Poker oder Bridge, wo mit verdeckten Karten gespielt wird. Spiele wie z. B. Backgammon, bei denen Spielzüge inhärent vom Zufall abhängen, heißen zufallsgesteuerte Spiele (*stochastic games*) und stellen die KI vor ähnliche Probleme.

Im Unterschied zu Spielen mit vollständiger Information hängen Spielzüge bei zufallsgesteuerten Spielen wie auch bei Spielen mit unvollständiger Information von Wahrscheinlichkeitsannahmen ab. Wird die Entwicklung eines solchen Spiels durch Übergänge in einem diskreten Zustandsraum modelliert, müssen Suchalgorithmen zur Planung geeigneter Züge um probabilistische Komponenten erweitert werden. Besonders häufig werden dazu Bayes'sche Modelle bedingter Wahrscheinlichkeiten herangezogen, da sie konsistente Inferenzmechanismen bereitstellen, um Bewertungen anhand partieller Informationen vorzunehmen [19].

Insbesondere bei Spielen, in denen mögliche Züge vom Zufall abhängen, haben sich auch regelbasierte Systeme oder Expertensysteme als geeignet

Ansätze zur Programmierung künstlicher Spieler erwiesen. Bereits 1980 präsentierte Hans Berliner ein entsprechendes Programm, das in der Lage war, den damaligen Backgammon-Weltmeister zu schlagen [3]. Die entscheidende Idee bestand darin, Bewertungsfunktionen mit Fuzzy Regeln zu kombinieren, um verschiedenen Phasen des Spiels besser abzubilden. Ebenso haben Ansätze, die auf Algorithmen zur statistischen Mustererkennung beruhen, gute Leistungen in zufallsgesteuerten Spielen gezeigt. Beispielsweise verwendet das Programm TD-Gammon Neuronale Netze, um aus aktuellen Spielständen und Würfelergebnissen geeignete Backgammon Züge zu generieren [20].

KI für interaktive Spiele

Moderne Computerspiele gehen in ihrer Komplexität deutlich über die eingeschränkten Szenarien von Brett- oder Kartenspielen hinaus. Sie realisieren komplexe virtuelle Welten, in denen die Spieler mehr oder weniger kontinuierlich frei und unvorhergesehen agieren können, sodass sich Spielverläufe nur schwerlich in diskrete Zustandsräume einbetten lassen. Künstliche Agenten in solchen Welten müssen über ein Repertoire von Handlungen verfügen, das in der Interaktion mit menschlichen Spielern den Eindruck einer glaubhaften Persönlichkeit erzeugt, die eigene Ziele und eigene Strategien zur Verfolgung dieser Ziele hat. Derartige Spiele werden als interaktive Spiele (*interactive games*) bezeichnet.

Interaktive Computerspiele haben sich in den letzten 30 Jahren zu einem der wichtigsten Anwendungsfelder für Methoden der KI entwickelt. Nahezu jedes moderne Computerspiel, angefangen bei den populären Action-Spielen über persistente Onlinewelten wie *World of Warcraft* bis hin zu Browserspielen wie *Farmville* greift in Teilbereichen auf Modelle und Algorithmen der KI zurück. Aufgrund der spezifischen Anforderungen interaktiver Spiele haben sich dabei weitestgehend andere Verfahren etabliert, als diejenigen, die bei klassischen Brettspielen zum Einsatz kommen.

In vielen Genres kommt der Figur des autonomen Agenten (engl. *Gamebot*) in einer virtuellen zwei- oder dreidimensionalen Umgebung eine besondere Bedeutung zu. Dieser tritt zumeist als Gegner oder Mitstreiter innerhalb der Spielwelt auf und sollte, um realistisch zu wirken, über ähnliche kognitive Eigenschaften und Fähigkeiten verfügen wie menschliche Spieler. Übernatürliche Reflexe,

globales Wissen oder Supersensoren, wie etwa ein Röntgenblick durch virtuelle Wände hindurch, sind daher ebenso zu vermeiden wie offensichtlich dummes Verhalten, wie zum Beispiel unkoordinierte Bewegungen gegen Hindernisse oder häufige Wiederholungen bestimmter Aktionen. Einer der wichtigsten Unterschiede zur klassischen KI ist daher das Ziel, ein Verhalten zu simulieren, das intelligent erscheint, aber nicht notwendigerweise intelligent sein muss. Es ist durchaus üblich, zu betrügen, etwa indem auf alle Variablen zugegriffen wird, die den Spielzustand des Spiels beschreiben, oder aber indem auch Bewegungen jenseits der Gesetze der Spielphysik ausgeführt werden, natürlich nur, wenn dies dem menschlichen Spieler verborgen bleibt. Künstlich erzeugtes Verhalten sollte darüber hinaus nicht optimal an die jeweilige Situation angepasst sein, sondern vielmehr ein herausforderndes und unterhaltsames Spielerlebnis erzeugen.

Auch wenn sich die Fähigkeiten computergesteuerter Spielfiguren in den letzten Jahren stetig erweitert haben, werden nach wie vor eher einfache Methoden zur Modellierung und Implementierung von Verhalten eingesetzt. Klassischerweise erfolgt die Verhaltenssteuerung, z. B. in Action-Spielen, durch Regelsysteme, endliche Automaten oder auch Fuzzy Regler. Dabei wird der aktuelle Zustand der Spielfigur durch eine Reihe abstrakter Merkmale charakterisiert (z. B. Aufenthaltsort in der Spielwelt, verfügbare Gegenstände, Nähe zu anderen Spielern, ...) und sinnvolle Aktionen werden je nach Zustand entweder deterministisch oder zufallsgetrieben generiert [10, 14]. Derartige Modelle verfügen zum Teil über eine beachtliche Komplexität, wodurch sich das mittlerweile sehr umfangreiche Verhaltensrepertoire von Gamebots erklärt, das neben einfachen reaktiven Verhaltensweisen auch komplexe Teamfähigkeiten beinhaltet. Um bekannten Schwierigkeiten bei der Skalierbarkeit der genannten Verfahren zu begegnen, wurden in den letzten Jahren eine Vielzahl heuristischer Erweiterungen vorgeschlagen, die oftmals an ein bestimmtes Genre oder Spiel adaptiert sind. So wurden z. B. im Spiel *Black and White* (Lionhead Studios, 2001) neuronale Netze und statistische Entscheidungsbäume implementiert, die es ermöglichen, dass virtuelle Kreaturen im Laufe der Zeit dazulernen und ihr Verhalten an die Vorgaben menschlicher Spieler anpassen können. Im Spiel *F.E.A.R.* (Monolith Productions, 2005) optimieren

computergesteuerten Gegner ihr Verhalten über eine Planungskomponente.

Neben der Steuerung künstlicher Spielfiguren haben Methoden der KI in den letzten Jahren auch in anderen Bereichen der Computerspielentwicklung Verbreitung gefunden. Eines der interessantesten Beispiele ist der *AI-Director* in *Left for Dead* (Valve, 2008), der Teile der gesamten Spielewelt steuert. Das heißt hier werden die oben genannten Techniken dazu verwandt, um den Aufbau von Levels oder die Abfolge von Aufgaben innerhalb des Spiels dynamisch zu gestalten, um ein weniger vorhersagbares Spielerlebnis zu ermöglichen. Die Steuerung in *Left for Dead* geht dementsprechend deutlich über reine Zufallsmechanismen hinaus und erzeugt so glaubhafte Spannungsbögen.

Generell ist die Steuerung künstlicher Agenten in komplexen Spielumgebungen ein nicht triviales Problem. Eine spezielle Herausforderung ergibt sich durch die fortwährende Kombination von reaktiven Verhaltensweisen (kurzfristige Aktionen bezogen auf den aktuellen Spielzustand), taktischen Verhaltensweisen (mittelfristige Aktionsfolgen bezogen auf die aktuelle Spielsituation) und strategische Verhaltensweisen (langfristige Aktionsfolgen zur Verfolgung eines langfristigen Zieles). Die Notwendigkeit zur Modellierung unterschiedlich granularer Verhaltensweisen ergibt sich aus den immer realistischeren Spielumgebungen und der damit verbunden wachsenden Komplexität von Aktionen: Ein Geist im Spiel Pac-Man hatte deutlich weniger taktische Optionen als ein Teammitglied eines modernen Action-Spiels.

Neben der expliziten Modellierung von Verhaltensweisen haben sich Verfahren des maschinellen Lernens als geeignet erwiesen, um anhand von Beobachtungen menschlicher Spieler entsprechende Verhaltensweisen zu generieren. Der gängige Ansatz besteht hier darin, Spiele aufzuzeichnen und die so gewonnenen Daten zu nutzen, um Klassifikationsalgorithmen zu trainieren, die Zusammenhänge zwischen abstrakten Spielzuständen und Aktionsmustern lernen. Da die aufgezeichneten Daten sowohl Informationen über die Zustände der von den Spielern gesteuerten Figuren als auch über deren Aktionen enthalten, wird es möglich, statistisch zu ermitteln, in welchen Zuständen welche Aktionen zu günstigen Ergebnissen führen. So wurden neuronale Netze für reaktives Verhalten [1], probabilistische Bayes'sche Netzwerke für komplexere Aktionen [11]

und Bewegungen [21] und Clusterverfahren zum Erlernen taktischer Verhaltensweisen [22] erfolgreich eingesetzt. Eine besondere Herausforderung besteht jedoch noch immer in der Integration, d. h. der gleichzeitigen Betrachtung unterschiedlicher Verhaltensebenen [23].

Der Bedeutung interaktiver Computerspiele für die weitere Entwicklung der KI lässt sich momentan nur schwer abschätzen. Das Problem der Verhaltenssimulation für einfachere Spiele, insbesondere Geschicklichkeits- und Denkspiele, kann man als gelöst betrachten. Für das beliebte *jump-and-run* Spiel *Super Mario*, zum Beispiel, kann plausibles und bei Bedarf unschlagbares Verhalten durch einen einfachen A* Algorithmus erzeugt werden. Für komplexere Spiele gibt es in der Regel jedoch keine einfachen Lösungen: Der Abstraktionsgrad im Vergleich zur realen Welt ist gering und erfordert eine Kombination unterschiedlichster Verhaltensweisen. Es ist unwahrscheinlich, dass z. B. ein einfacher Suchbaum Verhalten dieser Komplexität abbilden kann, sodass sich hier weiterer Forschungsbedarf ergibt. Die Idee, komplexe Computerspiele in der akademischen KI-Forschung zu betrachten, ist dabei nicht neu und wurde insbesondere in Bezug auf die Programmierung quasi-menschlichen Verhaltens schon häufiger diskutiert [10]. Neu ist jedoch, dass das Problem in der letzten Zeit vermehrt in Form von Wettbewerben angegangen wird, die sich am Vorbild des RoboCup orientieren und zum Ziel haben, möglichst menschlich wirkende Computerspieler zu entwickeln [4]. Inwieweit der Wettbewerbsgedanke dazu beitragen wird, substantielle Fortschritte auf diesem Gebiet zu machen, bleibt vorerst abzuwarten.

KI zur Analyse von Spielerzufriedenheit

Das MMORPG *World of Warcraft* (Blizzard, 2004) erschien im Dezember 2004 und entwickelte sich schnell, wenn auch nur für kurze Zeit, zum erfolgreichsten Spiel aller Zeiten. Anfang 2010 hatte es mehr als 12.000.000 Spieler, die sich trotz eines monatlichen Beitrags von mindesten 13 Dollar und zusätzlicher Kosten innerhalb des Spieles mindestens einmal pro Woche auf einen entsprechenden Server einloggten. Eine solche Breitenwirkung war bis dahin einmalig und Industrie und Experten waren sich darin einig, dass der Erfolg, d. h. der anhaltende Spielspaß, von *World of Warcraft* darauf zurückzuführen sei, dass es der Firma Blizzard gelungen war, das Spiel so zu programmieren, dass

es sich automatisch und beinahe unmerklich an die individuellen Fähigkeiten seiner Spieler anpasst. Das heißt der Spielverlauf wurde derart flexibilisiert, dass die Aufgaben, mit denen Spieler innerhalb des Spieles konfrontiert sind, abhängig von ihren mit der Zeit wachsenden Fähigkeiten generiert werden. Dadurch erscheint das aktuelle Spielgeschehen weder als zu leicht noch als zu schwierig, Langeweile oder Frustration werden verhindert, und das Spielerlebnis bleibt über lange Zeit spannend und herausfordernd.

Auch wenn der Erfolg von *World of Warcraft* erstaunlich erscheint, wurde er in jüngster Zeit noch übertroffen. Das Spiel *Farmville* (Zynga, 2009) wurde im Juni 2009 als eine Facebook Applikation veröffentlicht und erreichte innerhalb weniger Monate eine Fangemeinschaft von mehr als 80.000.000 regelmäßigen Spielern. Im Gegensatz zu *World of Warcraft* kann *Farmville* kostenlos erworben werden, denn das zugrundeliegende Geschäftsmodell besteht darin, dass die Spieler ihre Charaktere weiterentwickeln, indem sie innerhalb des Spieles virtuelle Gegenstände erwerben, die sie mit echtem Geld bezahlen.

Infolge des immensen kommerziellen Erfolges von *World of Warcraft* oder *Farmville* lässt sich momentan ein Paradigmenwechsel in der Spieleindustrie beobachten. Immer mehr Hersteller gehen dazu über, ihre lange Jahre erfolgreichen Titel nun umsonst anzubieten und durch Werbung innerhalb der Spiels oder durch den Verkauf begehrter Gegenstände Umsätze zu generieren. Diese Umstellung auf neue Geschäftsmodelle geht mit neuen Anforderungen an Methoden der KI einher. Stand bisher die Frage nach überzeugenden Computergegnern im Vordergrund, ergeben sich heute neue Fragen zur Modellierung, Erkennung und Vorhersage menschlichen Verhaltens.

Diese Fragen werden in den Disziplinen *Game Mining* und *Game Metrics* aufgegriffen, mit denen sich in den letzten Jahren weitere wichtige Anwendungsfälle für Methoden der KI und der computergestützten Statistik in Computerspielen ergeben haben. Da sich die Geschäftsmodelle der Spieleindustrie immer mehr zugunsten freier Browser Spiele und Spielen mit monatlichen Beiträgen verschieben, besteht eine der Herausforderungen für die Hersteller nun in der dauerhaften Kundenbindung. Es gilt, auf Kundenwünsche einzugehen, Kundenverhalten zu antizipieren und ganz

allgemein zu verstehen, wie sich Spieler in Spielen verhalten. Den vielversprechendsten Ansatz hierzu stellt die statistische Analyse aufgezeichneter Spieldaten dar. Diese *data mining* Aufgabe gleicht der Suche nach der sprichwörtlichen Nadel im Heuhaufen, nur dass es sich bei diesem Heuhaufen um Petabyte-große Logdateien menschlicher Aktivitäten handelt, die nach prominenten oder wiederkehrenden Mustern durchsucht werden müssen, die auf allgemeine Vorlieben oder Abneigungen der Spieler schließen lassen.

Ein konkretes Beispiel für den Einsatz statistischer Analysen findet sich im Zusammenhang mit dem Problem der automatischen Zulosung geeigneter Mitspieler in Spielen, die über das Internet gespielt werden. Da erfolgreiche Spiele heutzutage von mehreren Millionen Spielern gespielt werden, stellt sich die Frage, wer gegen wen spielen soll, um für alle Beteiligten den größtmöglichen Spielspaß zu erreichen. Eine bereits etablierte Lösung stellt das TrueSkill Verfahren von Microsoft dar [9], das zu einem festen Bestandteil der Paarung bei Xbox-live Spielen geworden ist. Der Ansatz ähnelt der vom Schach bekannten Elo-Wert-Methode, ergänzt diese um ausgeklügelte Bayes'sche Wahrscheinlichkeitsmodelle und liefert somit bereits nach sehr wenigen beobachteten Spielen konkrete Schätzungen über die Spielstärke der Spieler.

Einen weiteren großen Nutzen von *Game Mining* Methoden verspricht man sich momentan in Bezug auf mögliche Verbesserungen des Spieldesigns. Datengetriebene Analysen des Spieldesigns können die Arbeit menschlicher Entwickler sicher nicht ersetzen, wohl aber Hinweise auf mögliche Designfehler oder kritische Situationen innerhalb der Spielewelt liefern. Traditionell wurden Spiele ohne weitere Analysemöglichkeiten an den Kunden ausgeliefert; eine nachträgliche Änderung der Inhalte war dabei ebenso wenig möglich wie eine genaue Betrachtung des Nutzerverhaltens. Es blieb somit weitestgehend unklar, ob die zur Verfügung gestellten Spielwelten entsprechend ihrer Ausrichtung genutzt wurden oder wann und warum Spieler aufhörten zu spielen. Moderne Spiele hingegen werden weitestgehend online gespielt und selbst wenn ein Spieler allein gegen computergesteuerte Gegner antritt, besteht heutzutage typischerweise eine Verbindung zum Internet, sodass sich Aktivitäten innerhalb eines Spieles protokollieren lassen. Zur Beantwortung der oben genannten Fragen werden daher mittlerweile

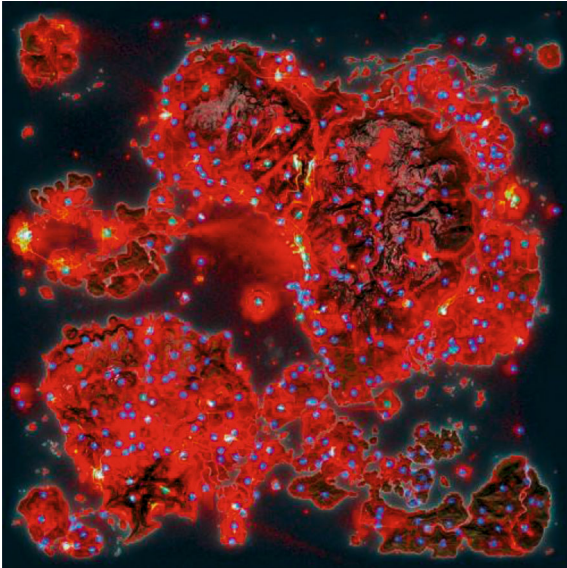


Abb. 2 Beispiel einer heatmap für das Spiel *Just Cause 2* (Square Enix, 2010). Das Luftbild der dreidimensionalen Spielwelt ist so eingefärbt, dass es Aufenthalts-Häufigkeiten nach 22,3 Millionen beobachteten Spieleraktionen widerspiegelt. Anhand solcher Daten lässt sich z. B. die Platzierung von Gegenständen innerhalb einer Spielwelt so optimieren, dass der Spielverlauf für alle Spieler möglichst fair und flüssig wird

einfache statistische Auswertung aufgezeichneter Spielerdaten durchgeführt, wie etwa die Analyse der räumlichen Nutzung der Spielwelt über sogenannte *heatmaps* (Abb. 2). Trotz aussichtsreicher Fortschritte in den letzten Jahren, z. B. bei der automatischen Vorhersage von Spielerverhalten in *Tomb Raider Underworld* (Eidos Interactive, 2008) [12] oder bei der Analyse der zeitlichen Entwicklung sozialer Gruppen in *World of Warcraft* [5, 24] und allgemeiner Nutzungsmuster [2], besteht in den Bereichen *Game Mining* und *Game Metrics* nach wie vor großer Forschungsbedarf [8].

Fazit

Die Entwicklung von Computerspielen und die Erforschung von Methoden der Künstlichen Intelligenz beeinflussen sich schon seit Jahrzehnten gegenseitig. Aus Sicht der KI standen und stehen dabei vor allem Fragen zur Modellierung menschlichen Verhaltens im Vordergrund. In den Anfängen ihrer gemeinsamen Entwicklung waren KI Methoden zunächst auf 2-Personen-Spiele mit vollkommener Information ausgerichtet. Insbesondere am Beispiel des Schachspiels wurde dabei eine Reihe von Verfahren entwickelt, die heute als klassische KI-

Algorithmen gelten und überzeugende Lösungen für viele Brettspiele liefern. Bedarf an Forschung und Entwicklung besteht jedoch weiterhin für Spiele wie Go, deren immenser Zustandsraum nach wie vor nur unzureichend exploriert werden kann.

Zur Programmierung künstlicher Spieler in Spielen mit unvollständiger Information oder in modernen interaktiven Spielen haben sich Kombinationen aus zustandsbasierten Modellen und Methoden des maschinellen Lernens als Standardlösungen etabliert. Auch wenn es hier in den letzten Jahren enorme Fortschritte in Richtung glaubwürdigen menschlichen Verhaltens gegeben hat, stellt die integrierte Betrachtung von reaktiven, taktischen und strategischen Verhaltensweisen weiterhin ein Problem dar, das bisher nicht überzeugend, d. h. verallgemeinerbar gelöst werden konnte.

Infolge neuester Entwicklungen bei Spielen, die etwa innerhalb webbasierter sozialer Netzwerke gespielt werden, treten in jüngster Zeit Fragen zur Erkennung menschlichen Verhaltens in den Vordergrund. *Game Mining* und *Game Metrics* sind vergleichsweise neue Disziplinen, die versuchen, anhand der riesigen Datenmengen, die in modernen, internetbasierten Spielen anfallen, zu untersuchen und zu modellieren, wie sich menschliche Spieler wann verhalten. Diese Fragen haben beim aktuellen Stand der Spieleindustrie große wirtschaftliche Bedeutung, da man sich Antworten darauf erhofft, was ein Spiel attraktiv macht bzw. wie ein Spiel über längere Zeit attraktiv gehalten werden kann, um eine möglichst große Nutzercommunity möglichst lange an das Spiel zu binden. Zur Beantwortung dieser Fragen gilt es insbesondere, Methoden zu entwickeln, die einerseits in der Lage sind, riesige Datenmengen effizient zu verarbeiten und andererseits Analyseergebnisse zu liefern, die nicht nur Statistikern verständlich sind, sondern vor allem Programmierer, Designer und Marketing Experten in der Industrie bei ihrer Arbeit unterstützen.

Literatur

1. Bauckhage C, Thureau C, Sagerer G (2003) Learning Human-like Opponent Behavior for Interactive Computer Games. In: Michaelis B, Krell G (eds) Pattern Recognition. LNCS, vol 2781. Springer, pp 148–155
2. Bauckhage C, Kersting K, Sifa R, Thureau C, Drachen A, Canossa A (2012) How Players Lose Interest in Playing a Game – An Empirical Evaluation Based on Distributions of Total Playing Times. In: Proc. Int. Conf. on Computational Intelligence and Games, IEEE
3. Berliner H (1980) Backgammon Program Beats World Champ. ACM SIGART Newslett 69:6–9
4. Bohannon J (2010) Killer Bots Are Getting Human. Science 330:30–31
5. Bohannon J (2010) Game-Miners Grapple with Massive Data. Science 330:30–31

6. Buro M (1997) The Othello Match of the Year: Takeshi Murakami vs. Logistello. *Int Comput Chess Assoc J* 20(3):189–193
7. Cass S (2002) Mind Games. *IEEE Spectrum* 39(12):40–44
8. El-Nasr M, Drachen A, Canossa A (eds) (2013) *Game Analytics*. Springer
9. Herbrich R, Minka T, Graepel T (2007) TrueSkill(TM): A Bayesian Skill Rating System. In: *Proc. Advances in Neural Information Processing Systems*
10. Laird J (2001) Using a Computer Game to develop advanced AI. *IEEE Computer* 34(7):70–75
11. Le Hy R, Arrigioni A, Bessiere P, Lebeltel O (2004) Teaching Bayesian Behaviours to Video Game Characters. *Robot Auton Syst* 2–3(47):177–185
12. Mahlman T, Drachen A, Canossa A, Togelius J, Yannakakis G (2010) Predicting player behavior in Tomb Raider: Underworld. In: *Proc. Int. Conf. on Computational Intelligence and Games*, IEEE
13. Nilsson N (1998) *Artificial Intelligence: A new Synthesis*. Morgan Kaufmann
14. Nareyek A (2004) Artificial Intelligence in Computer Games – State of the Art and Future Directions. *ACM Queue* 1(10):58–65
15. Reuters (2011) A Look at the \$65 Billion Video Games Industry, June 2011
16. Russel S, Norwig P (2010) *Artificial Intelligence: A Modern Approach*, 3rd edn. Prentice Hall
17. Schaeffer J, van den Herik H (2001) Games, Computers, and Artificial Intelligence. *Artif Intell* 1–2(134):1–7
18. Shannon C (1950) Programming a Computer for Playing Chess. *Philos Mag* 41(314):256–275
19. Southey F, Bowling M, Larson B, Piccione C, Burch N, Billings D, Rayner C (2005) Bayes’ Bluff: Opponent Modeling in Poker. In: *Proc. Conf. on Uncertainty in Artificial Intelligence*
20. Tesauro G (1995) Temporal Difference Learning and TD-Gammon. *Commun ACM* 38(3):58–68
21. Thureau C, Bauckhage C, Sagerer G (2004) Synthesizing Movements for Computer Game Characters. In: Rasmussen C, Bühlhoff H, Giese M, Schölkopf B (eds) *Pattern Recognition. LNCS*, vol 3175, Springer, pp 179–186
22. Thureau C, Bauckhage C (2005) Tactical Waypoint Maps: Towards Imitating Tactics in FPS Games. In: *Proc. Int. Conf. Computer Game Design and Technology*. ACM
23. Thureau C, Bauckhage C, Sagerer G (2004) Imitation Learning at all Levels of Game-AI. In: *Proc. Int. Conf. Computer Games, Artificial Intelligence, Design and Education*
24. Thureau C, Bauckhage C (2010) Analyzing the Evolution of Social Groups in World of Warcraft. In: *Proc. Int. Conf. on Computational Intelligence and Games*, IEEE