Evolutionsstrategien und Reinforcement Learning für strategische Brettspiele

Prof. Dr. Wolfgang Konen
wolfgang.konen@fh-koeln.de, Tel. 02261/8196-6275

Prof. Dr. Thomas Bartz-Beielstein
thomas.bartz-beielstein@fh-koeln.de, Tel. 02261/8196-6391

Abstract: Das Erlernen von strategischen Brettspielen allein aus Beispielpartien repräsentiert eine wichtige Problemstellung im maschinellen Lernen. Wir vergleichen die Eignung von Reinforcement Learning (bestärkendem Lernen, TDL) und Evolutionsstrategieen (Neuroevolution, CMA-ES) für diese Aufgabe. Wir zeigen, dass wesentliche Faktoren für Erfolg oder Misserfolg sind: (a) die Wahl geeigneter Merkmale und (b) die Wahl der richtigen Fitnessfunktion für Evolutionsstrategien.

Einleitung

Computerprogramme, die gute Gegenspieler in strategischen Brettspielen sind, gehören schon seit Jahren zum Forschungsgegenstand der Künstlichen Intelligenz, und es wurden hier auch beachtliche Erfolge erzielt.[1] Allerdings sind die Programme in der Regel von menschlichen Experten mit viel Erfahrung im jeweiligen Spiel entwickelt, haben Bibliotheken erfolgreicher Spiele eingebaut oder sie verfolgen mehr oder weniger aufwändige Suchstrategien. Jedes neue Spiel bedeutet dann wieder einen völlig neuen Analyse- und Codierungsaufwand.

Ein generischerer Ansatz bestünde darin, dass ein Programm durch „try & error“ aus der Beobachtung und Durchführung zahlreicher Spielverläufe selbst lernen kann, was die besten Strategien sind. Wenn ein solcher Ansatz gelingt, so ist er viel besser auf andere strategische Situationen übertragbar. Wir verfolgen in unserem Projekt diesen zweiten Ansatz, der die Bedingungen für das Lernen an sich erforscht: Wie können wir es schaffen, dass ein Computer ohne Strategiewissen, allein durch das Spielen gegen sich selbst, zum Teil gemischt mit dem zufälligen Ausprobieren neuer Spielzüge, im Laufe der Zeit hinzu lernt, d.h. sich in einem solchen für ihn neuen Spiel sukzessiv besser zu behaupten lernt? Welche Bedingungen müssen erfüllt sein, damit Lernen hier (a) möglich ist und (b) sich möglichst schnell vollzieht?

Reinforcement Learning

Reinforcement Learning (RL, dt.: Bestärkendes Lernen) ist eine mächtige Optimierungsmethode für komplexe Probleme. Es hat besonders dann seine Vorteile, wenn nicht für jede einzelne Aktion eine Belohnung gegeben werden kann, sondern erst später, nach einer Sequenz von Aktionen. Dies ist typischerweise bei Brettspielen der Fall.

Temporal Difference (TD) Learning ist ein konkretes Reinforcement-Lernverfahren, das durch Sutton und Barto [1] entwickelt wurde und mit Tesauro’s TD-Gammon [2], einem selbstlernenden Computerprogramm, das das Spiel Backgammon auf Weltklasseniveau spielt, große Popularität erlangt hat. Trotz dieses Anfangserfolges stellte sich in späteren Anwendungen das TD Learning oft als schwierig heraus, da es für andere Spiele oder leicht andere Randbedingungen keine guten Ergebnisse erzielte.

---

[1] Beispielsweise mit dem Fritz-Schachprogramm (www.chessbase.de), das sich bereits in Turnieren gegen menschliche Schachweltmeister behauptet hat.

Wir starteten deshalb ein Forschungsprojekt, um die Bedingungen für den Erfolg oder Mißerfolg von TD-Anwendungen genauer zu studieren. Die grundlegenden Algorithmen für selbstlernende TD-Agenten in Brettspielen sind in [3] detailliert beschrieben.

Evolutionsstrategien, Neuroevolution und CMA-ES

Für Reinforcement-Learning-Probleme ist ein weiteres Verfahren, die sog. Neuroevolution in den vergangenen Jahren erfolgreich eingesetzt worden. Hierbei handelt es sich um die Optimierung eines Neuronalen Netzes mit den Mitteln der Evolutionsstrategien [4][5]. Wir verwenden in unserem Projekt die Hansen entwickelte CMA-ES (Covariance Matrix Adaptation Evolution Strategy) [6], die sich als robustes und performantes Optimierungsverfahren bei komplexen Lernaufgaben herausgestellt hat.

Die Zielsetzung des Projektes

Unser genereller Forschungsgegenstand in diesem Projekt ist das bessere Verständnis von Lernvorgängen im Rahmen komplexer (strategischer) Entscheidungsprozesse. Fortschritte, die hier erzielt werden, können die adaptive Entscheidungsfindung in komplexen Steuerungs- und Regelungsproblemen weiter voranbringen.

Konkret suchen wir im Rahmen dieses Projektes auf Antworten für folgende Fragen:

1. Welche Rahmenbedingungen trennen erfolgreiche von erfolglosen Lernversuchen im Reinforcement Learning?
2. Wie bedeutsam sind (abgeleitete) Merkmale für den Lernprozess und kann man konkret die Nützlichkeit eines Merkmals im Vorhinein abschätzen?
3. Gibt es einen signifikanten Performanzunterschied zwischenNeuroevolution mit CMA-ES und TD-Learning in Bezug auf RL für strategische Brettspiele?

Die letzte Frage nimmt Bezug auf eine kürzlich von S. Lucas vorgestellte Arbeit [7], in der er berichtet, dass für ein bestimmtes strategisches Spiel der TD-Ansatz um einen Faktor 10 schneller lernte als ein auf Evolutionsstrategien basierender Ansatz.

Als Basis für unsere Untersuchungen nutzen wir das in Abbildung 1 gezeigte Spiel TicTac-Toe. Dies gehört zu den relativ einfachen Spielen, da es nur 5.890 verschiedene Spielzustände gibt. Trotzdem ist es nicht leicht nur aus Beispielen zu erlernen.

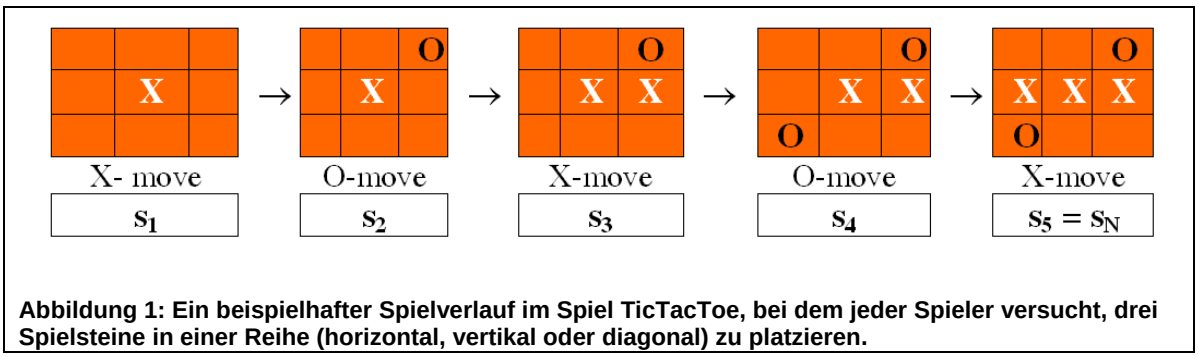

Abbildung 1: Ein beispielhafter Spielverlauf im Spiel TicTacToe, bei dem jeder Spieler versucht, drei Spielsteine in einer Reihe (horizontal, vertikal oder diagonal) zu platzieren.

Resultate

Die hier dargestellten Ergebnisse sind im Detail in einem aktuellen Beitrag der Autoren zur Konferenz GECCO'2009 [8] nachzulesen und bauen auf einer früheren Arbeit der Autoren auf [9], aus der wir das TD-Lernverfahren übernehmen konnten.

Wir gelangen zu folgenden Antworten für die im vorigen Abschnitt formulierten Fragen

1. Sowohl falsche Fitness-Funktionen als auch unzulängliche Merkmale (s. nächster Abschnitt) können den Lernerfolg stark behindern oder unmöglich machen. Wenn eine einfache „Jeder-gegen-jeden"-Fitness verwendet wird, stellt sich stets eine zu frü-

he Konvergenz auf schlechte Spieler ein. Es ist wichtig, dass die Opponenten, gegen den ein Agent seine Fitness ermittelt, über längere Phasen des Lernvorganges konstant bleiben. Mit einer Fitness-Funktion $F^{(4)}$, bei der die Opponenten ein Pool von 3 Spielern sind, die gelegentlich gewechselt werden, erreichen wir die besten Ergebnisse für das „Lernen ohne Lehrer“[2].

2. Abgeleitete Merkmale sind von ausschlaggebender Bedeutung. Wie Abbildung 2 zeigt, ist es mit der bloßen Kodierung der Brettposition (Merkmalssatz T0) in der Regel nicht getan. Dagegen findet mit abgeleiteten Merkmalen (Beispiel: 2 in horizontaler Reihe, nicht blockiert durch gegnerischen Stein), wie sie in den Merkmalssätzen T2, T3 und T4 Verwendung finden, ein mehr oder weniger schneller Lernerfolg statt. Mit der sog. *Feature Set Utility* entwickeln wir in [8] ein neues Gütemaß, mit dem sich die Nützlichkeit eines Merkmals im Vorhinein abschätzen läßt.
3. Im Gegensatz zu Lucas [7], der in einem anderen strategischen Spiel von einer 10fach langsameren Evolutionsstrategie berichtet hatte, finden wir hier, dass CMA-ES mit TD-Learning in etwa gleich auf liegt. Dies kann an den besseren Konvergenzeigenschaften von CMA-ES liegen. Es gilt aber in jedem Fall nur, wenn eine geeignete Fitnessfunktion $F^{(4)}$ verwendet wird!

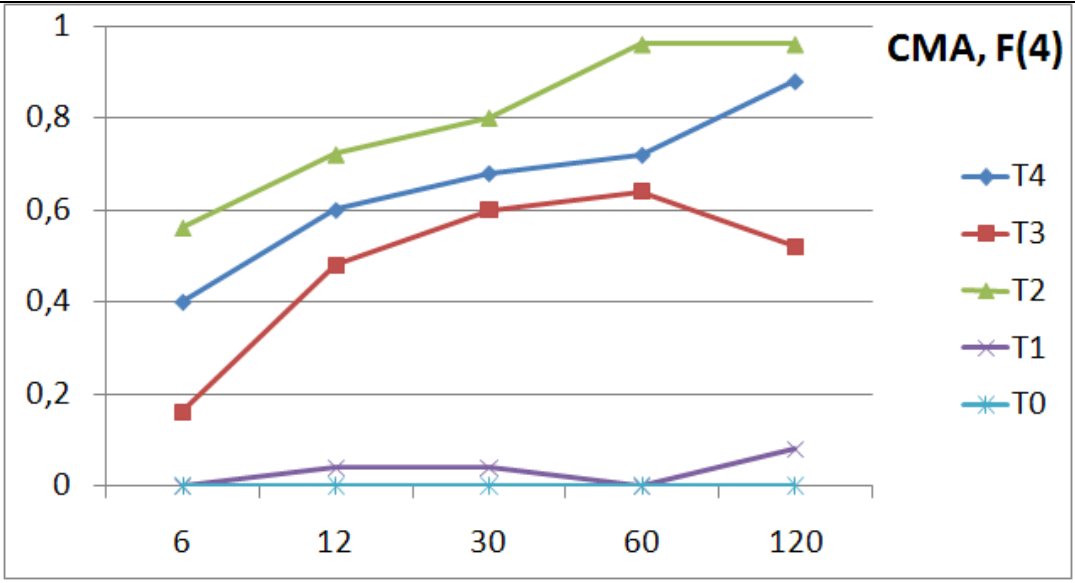

Abbildung 2: Lernerfolg des CMA-ES-Agenten im Spiel TicTacToe. Die horizontale Achse zeigt die Anzahl der Trainingsspiele (in Tausend), die vertikale Achse die Erfolgsrate der trainierten Agenten. Jeder Punkt basiert auf der Mittelung über 25 trainierte Agenten. Die bloße Brettstellung T0) ist als Merkmalsatz für RL nicht hinreichend. Dagegen lernen die Merkmalssätze T2, T3 und T4 vergleichsweise schnell eine gute Performance. In allen Fällen fand die im Text beschriebene Fitnessfunktion $F^{(4)}$ Verwendung [8].

Ausblick

Inzwischen arbeiten wir an dem anspruchsvolleren Brettspiel „Vier gewinnt“ (Connect-4), bei dem die Zahl der Spielzustände bei über 10^{14} liegt. Dieses Brettspiel war schon Gegenstand verschiedener Diplomarbeiten an der FH Köln [10][11][12], die das Problem ohne TD-Learning bearbeiteten, sowie einer aktuellen Diplomarbeit [13], die erstmalig einen TD-Agenten einsetzt. Der TD-Agent kann bereits bestimmte Endspiele (Abbildung 3) erlernen, es fehlt allerdings noch der Einsatz von Merkmalen. Das Spielverhalten des TD-Agenten ist

[2] Mit „Lernen ohne Lehrer“ bezeichnen wir den Umstand, dass während der Lernphase kein vorab vorgegebener perfekter oder guter Spieler als Vergleich (auch nicht als Opponent) zur Verfügung steht.

daher noch lange nicht optimal, wenngleich er gegen einen zufällig ziehenden Agenten sicher gewinnt.

Eine weitere Ausgestaltung des TD-Learning-Agenten für Connect-4 ist geplant. Es sollen insbesondere (abgeleitete) Merkmale in den TD-Lernvorgang mit eingebracht werden. Die in dieser Arbeit entwickelte *Feature Set Utility* kann hierbei von wesentlicher Hilfe sein für die Evaluation von Merkmalen.

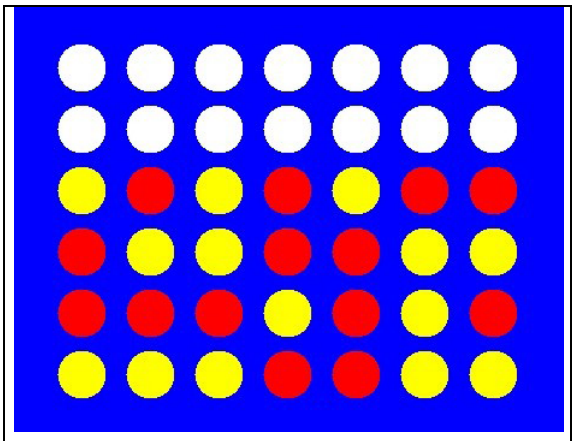

Abbildung 3: Endspielsituation aus dem Spiel Connect-4.

Für die allgemeine Herangehensweise an strategische Lernsituationen mit einer hohen kombinatorischen Vielfalt von Zuständen ist es interessant, ein generisches Vorgehen zur Gewinnung von Merkmalen zu besitzen. Hierfür erscheinen uns N-Tupel-Systeme [14] besonders geeignet. Deren Nutzen für TD-Learning möchten wir im Forschungsprojekt SOMA (Systematische Optimierung von Modellen in Informatik und Automatisierungstechnik)[15] weiter untersuchen.

Danksagung: Dieses Projekt wurde von der FH Köln im Rahmen des anerkannten Foschungsschwerpunktes COSA gefördert. Es wird derzeit weitergeführt im Kontext des vom Bundesministerium für Forschung und Bildung (BMBF) geförderten Projektes SOMA (AIF FKZ 17N1009, Förderlinie "Ingenieurnachwuchs")

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Literatur

[1] Richard S. Sutton, Andrew G. Barto: *Reinforcement Learning - An Introduction*. MIT Press, Cambridge, 1998.
[2] Gerald Tesauro: *Temporal Difference Learning and TD-Gammon*, *Communications of the ACM,* March 1995 / Vol. 38, No. 3.
[3] W. Konen: *Reinforcement Learning für Brettspiele: Der Temporal Difference Algorithmus*, Techn. Report, Institut für Informatik, FH Köln, Okt. 2008. (PDF)
[4] C. Igel. *Neuroevolution for reinforcement learning using evolution strategies*. In R. S. et al., editor, Proc. Congress on Evolutionary Computation (CEC 2003), pages 2588–2595, 2003.
[5] D. Whitley, S. Dominic, R. Das, and C. Anderson. *Genetic reinforcement learning for neurocontrol problems*. Machine Learning, 13:259–284, 1993.
[6] N. Hansen and A. Ostermeier. *Completely derandomized self-adaptation in evolution strategies*. Evolutionary Computation, 9:159–195, 2001.
[7] S. Lucas. *Investigating learning rates for evolution and temporal difference learning*. In Proc. IEEE Symposium on Computational Intelligence and Games CIG2008, Perth, Australia, December 2008. IEEE Press.
[8] W. Konen, T. Bartz-Beielstein, *Reinforcement Learning for Games: Failures and Successes – CMA-ES and TDL in comparision*, GECCO 2009, Montreal, July 2009. (PDF)
[9] W. Konen, T. Bartz-Beielstein: *Reinforcement Learning: Insights from Interesting Failures in Parameter Selection*. In: G. Rudolph et al. (ed.), 10th International Conference on Parallel Problem Solving From Nature (PPSN2008), Dortmund, September 2008, p. 478-487, Lecture Notes in Computer Science, LNCS 5199, Springer, Berlin, 2008. (PDF)
[10] T. Wende: *Entwurf und Anwendung künstlicher neuronaler Netze zum Lernen strategischer Brettspiele*, Diplomarbeit, FH Köln, Okt 2003.
[11] T. Rudolph: *Konzeption einer Entwicklungsumgebung lernender KNN für strategische Spiele*, Diplomarbeit, FH Köln, Sept 2003.
[12] A. Klassen: *Evaluation der Einsetzbarkeit lernfähiger neuronaler Netze für das strategische Brettspiel „4-Gewinnt“*, Bachelorarbeit, FH Köln, Feb 2005.
[13] J. Schwenck, *Einsatz von Reinforcement Learning für strategische Brettspiele am Beispiel von "4-Gewinnt“*, Diplomarbeit, FH Köln, Okt 2008.
[14] W. Bledsoe, I. Browning: *Pattern recognition and reading by machine*. In: Proceedings of the EJCC, pp. 225 232, 1959.
[15] Förderprojekt SOMA, www.gm.fh-koeln.de/go/soma, Okt. 2009.