Andeas Dörr

Prädiktive Modelle für daten-effizientes Reinforcement Learning

In aktuellen industriellen Anwendungen und Produkten begegnen uns eine immer größere Systemkomplexität und ein Zusammenwirken von vielen verschiedenen Teilsystemen. Gleichzeitig erwarten wir eine möglichst effiziente Nutzung aller Ressourcen bei optimaler Systemperformance. Bei der Regelung und Ansteuerung vieler dieser Systeme ist allerdings unklar, wie eine optimale Lösung aussehen könnte und gegebenenfalls steht der Aufwand um eine solche Lösung zu finden in keinem Verhältnis zum erwarteten Gewinn. Klassischerweise werden Regelungskonzepte für solche Systeme von Experten anhand von (physikalischem) Modellverständnis oder anwendungsspezifischen Heuristiken abgeleitet und analysiert. Die Modellierung dieser Systeme stößt jedoch oft an Grenzen, sei es durch die Komplexität der physikalischen Wirkmechanismen, die Größe und Komplexität des Problems oder Begrenzungen in Budget oder Zeit.

Aktuelle Methoden des *Maschine Learnings (ML)* und der *Künstlichen Intelligenz (KI)* eröffnen einen neuen und abweichenden Lösungsansatz. Dabei wird nicht mehr explizit Wissen über das System und den Weg zur möglichen Lösung vorausgesetzt, sondern die Maschine selbst muss Charakteristika des Problems und ihrer Umgebung erfassen und die optimale Lösung zur gestellten Aufgabe selbst finden. Vorwissen über das System kann zwar eingebracht werden, die wesentliche Charakterisierung erfolgt jedoch über eine abstrakte Beschreibung des gewünschten Verhaltens. Die Maschine selbst muss nun identifizieren, welche konkreten Abläufe erforderlich sind um die abstrakten Ziele zu erreichen.

Autonomes Lernen von neuem Verhalten

Beim *Reinforcement Learning (RL)*, oder auch bestärkendem Lernen, geht man idealisiert von einer Interaktion eines Agenten mit einer zunächst unbekannten Umgebung aus. Der Agent kann durch Aktionen Einfluss auf seine Umgebung ausüben und mittels seiner Sinne gewisse Auswirkungen seiner

Kurzzusammenfassung:

Eines der großen Versprechen von Machine Learning (ML) und Künstlicher Intelligenz (KI) ist es, einer Maschine die Fähigkeit zu verleihen, in komplexen und unbekannten Umgebungen autonom neues Verhalten zu erlernen. Ergebnisse dieser Forschungsrichtung, im speziellen der Disziplin des Reinforcement Learnings (RL), sind allerdings nach wie vor überwiegend beschränkt auf sehr spezialisierte, simulierbare und gut verstandene Problemfelder. Die Arbeit von Andreas Dörr adressiert dagegen Probleme wie: Daten-Effizienz, Robustheit und Lernen in Gegenwart von fehlerhafter und unvollständiger Wahrnehmung der Umgebung. Diese Probleme treten typischerweise auf, sobald maschinelle Systeme in unseren unstrukturierten, alltäglichen Umgebungen operieren sollen. Kernthemen seiner Arbeiten sind das Erlernen probabilistischer, prädiktiver Modelle, die es der Maschine erlauben aus vergangenen Erfahrungen optimale Entscheidungen für zukünftige Handlungen abzuleiten.

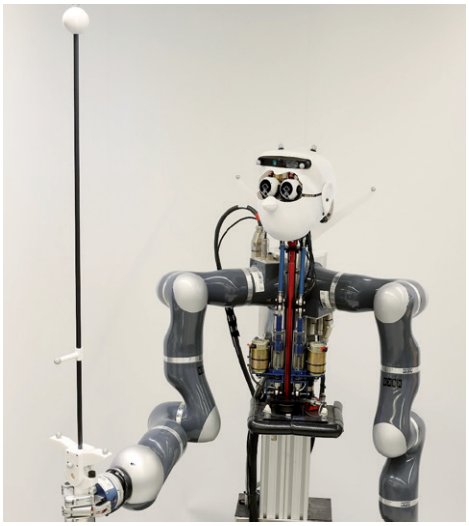

Abbildung 1:
Methoden des Reinforcement Learnings ermöglichen es dem Roboter Apollo das invertierte Pendel in seiner Hand zu balancieren. Im Gegensatz zu Fertigungsrobotern in Industrieanlagen hat er zunächst kein Wissen über sich und seine Umgebung. Er ist nicht vorprogrammiert und muss selbst herausfinden, wie er z. B. seinen Arm optimal ansteuern kann.

Aktionen auf die Umgebung beobachten. Gleichzeitig erhält der Agent von Zeit zu Zeit Rückmeldung, ob er sein Ziel erreicht oder erfüllt hat oder diesem zumindest näher gekommen ist. Diese Rückmeldung, auch Reward genannt, versucht der Agent langfristig zu maximieren. Diese Methode des bestärkenden Lernens, mit positiver Belohnung oder negativer Bestrafung bei erwünschtem oder unerwünschtem Verhalten ist aus dem Tierreich bestens bekannt und untersucht.

Mit Methoden des Maschinellen Lernens, können die allgemeinen Prinzipien des Reinforcement Learnings auch auf die eingangs erwähnten, automatisierte und maschinelle Systeme übertragen werden. Ein Beispiel ist der Roboter Apollo in *Abbildung 1*. Dieser Roboter bildet einen humanoiden Oberkörper mit den Freiheitsgraden zweier menschlicher Arme und Hände nach. In diesem Fall, ist der Agent der Computer bzw. das Computerprogramm, das die Motoren in den Gelenken des Roboters als „Aktionen“ ansteuern kann um mit der Umgebung zu interagieren. Über Sensoren in den Gelenken können gleichzeitig die aktuelle Position der Arme und über Kameras die Umgebung erfasst werden. Eine Aufgabe könnte es nun sein, den im *Abbildung 1* gezeigten Stab auf der Handfläche zu balancieren. Diese Aufgabe des invertierten Pendels ist ein klassisches Problem der Regelungstechnik. Die Physik kann einfach modelliert werden und eine Vielzahl an Lösungsansätzen existiert. In diesem Fall, hat der Agent bzw. der Computer jedoch kein Vorwissen über die Physik des Problems bzw. seine Umgebung und das invertierte Pendel. Lediglich die Beobachtungen aus der Interaktion zwischen Roboterarm und

Pendel stehen zur Verfügung um nach und nach zu erlernen, wie die Motoren anzusteuern sind, um letztendlich das Pendel zu balancieren. Eine Methode um dieses Problem direkt auf dem realen System zu lösen ist in der Arbeit [1] dargestellt.

Im Reinforcement Learning wird diese Interaktion zwischen Agent und Umgebung üblicherweise im Formalismus des Markovschen Entscheidungsprozesses (engl. *Markov Descision Process (MDP)*) beschrieben. Hierbei verfügt der Agent zu jedem Zeitpunkt über eine vollständige Beschreibung der Umgebung (den sogenannten System-Zustand), mit Hilfe derer vollständig bestimmbar ist, wie die Umgebung auf eine Aktion des Agenten reagieren wird. In der Realität sind unsere Wahrnehmungen, ebenso die Sensorinformationen einer Maschine stark limitiert. Selbst mit den besten Kameras, Radar und Lidar Sensoren, kann zum Beispiel ein autonomes Fahrzeug einen Menschen hinter einem Hindernis zeitweilig nicht mehr direkt wahrnehmen. Dieses Problem wird als *Partially Observabe Markov Descision Process (POMDP)* abstrahiert. Obwohl ein System-Zustand existiert, muss der Agent Entscheidungen allein anhand seines verfügbaren Wissens und seiner verfügbaren Beobachtungen treffen. Diese Situation ist bei weitem der Normalfall in realen Anwendungen. Gleichzeitig sind viele der akademischen Methoden jedoch auf den MDP Fall zugeschnitten. Bereits rauschbehaftete oder fehlerhafte Wahrnehmungen machen es unmöglich präzise die weitere Entwicklung der Umgebung nur anhand einer Messung zu prognostizieren. Gravierender werden die Probleme, sobald manche Größen in der Umgebung (z. B. die Position oder Geschwindigkeit des Fußgängers) überhaupt nicht mit Sensoren zu erfassen sind.

> *„Die Fähigkeit, die eigene Unsicherheit und Unsicherheiten in der Umgebung abzuschätzen und zu berücksichtigen ist eine essentielle Voraussetzung für eine optimale Problemlösung."*
>
> *Andreas Dörr*

Ausgehend von dem allgemeinen, idealisierten MDP oder POMDP Szenario sind verschiedene Klassen von Methoden bekannt um aus den Interaktionsdaten zwischen Agent und Umgebung eine Regelvorschrift abzuleiten bzw. zu erlernen. Forschung in diesem Bereich hat sich zunächst auf Fälle spezialisiert, in denen aus einer endlichen Anzahl von Aktionen ausgewählt werden muss. Ein Beispiel wären Spiele wie Schach oder Go, wo zu jedem Zeitpunkt nur eine endliche Anzahl von Handlungsmöglichkeiten besteht. In einigen Spezialfällen

können so optimale Lösungen und Regelvorschriften gefunden werden. Oft aber übersteigt die Komplexität der verschiedenen möglichen Handlungsstränge bereits in diesen Problemen die Grenzen der Berechenbarkeit. In typischen Problemen in der realen Welt kann der Agent allerdings zu jedem Zeitpunkt eine Vielzahl von typischerweise kontinuierlichen Aktionen treffen. Beim Autofahren könnte der Mensch z. B. eine beliebigen Lenkradeinschlag und Gaspedalstellung bewirken, sodass der Raum der zukünftigen Entwicklung der Umgebung immer komplexer wird.

Wesentliche Fragestellungen im Reinforcement Learning sind daher, wie man mit möglichst wenig Interaktionszeit, möglichst viel über seine Umgebung erlernen kann und wie man mit der zur Verfügung stehenden Erfahrung möglichst gute Lösungsansätze entwickelt um seinen Reward zu maximieren und somit sein Ziel zu erreichen. Diese Frage gewinnt an Bedeutung, sobald man anstelle von simulierten, d. h. virtuellen Umgebungen z. B. Spiele, mit realen, physischen Umgebungen interagiert. Dies könnten zum Beispiel Roboter in Fabriken, selbstfahrende Autos oder autonome Rasenmäher sein. All diese Systeme könnten durch eine fehlerhafte und unvorsichtige Bedienung Schaden nehmen. Durch anhaltende und gegebenenfalls fehlerhafte Benutzung würden diese System zusätzlich verschleißen.

Modell-basiertes Reinforcement Learning

Eine Klasse der Reinforcement Learning Algorithmen bildet die Gruppe der modellbasierten Verfahren. Bei diesen Verfahren wird aus der bisher gesammelten Erfahrung ein Modell abgeleitet, mit dessen Hilfe das zukünftige Verhalten des Systems für die gewählten Aktionen des Agenten vorhergesagt werden kann. Dieses Modell ist sozusagen ein interner Simulator, mit dem der Agent verschiedene Verhaltensmuster durchprobieren kann und deren Resultate und Erfolgsaussichten bewerten kann, ohne wirklich mit dem realen System zu interagieren. Da der Agent über kein Vorwissen über seine Umgebung verfügt, muss er mit zufälligen Aktionen explorieren. Mit jeder ausgeführten Aktion sammelt er allerdings Erfahrung, die sein Modell verbessert und es ihm somit ermöglicht in Zukunft bessere Aktionen zu wählen. Die modell-basierten Verfahren sind daher im Kern iterative Methoden aus einem Schritt des Modell Lernens und einem Schritt der modell-basierten Optimierung des Verhaltens des Agenten. Mit jeder Iteration lernt der Agent neue Bereiche in seiner Umgebung kennen und nähert sich seiner eigentlichen Zielsetzung, sodass eine Anpassung des Modells erforderlich wird. Ein prädiktives Modell bietet dabei den Vorteil, dass unabhängig von der eigentlich zu erfüllenden Aufgabe, die Aktionen und ihre Auswirkungen auf die Umgebung beschrie-

ben werden können. Somit kann Erfahrung aus verschiedenen Aufgaben kombiniert werden.

Wesentliche Teile dieser Algorithmen bilden einerseits die gewählten Modellstrukturen und ihre Trainings- oder auch Lernmethode. Andererseits muss anhand des Modelles bestimmt werden, in welche Richtung innerhalb der Umgebung weiter exploriert werden muss um den Ziel näher zu kommen, aber auch wo bereits gute Verhaltensmuster gefunden wurden. Dieser Trade-off zwischen Exploration um neue Bereiche der Umgebung zu erkunden und Exploitation, d. h. Ausnutzung der bereits gefundenen Erkenntnisse ist inhärent im Reinforcement Learning Problem.

Typischerweise gibt es bei realen Umgebungen jedoch eine Reihe von Problemen, die das Erlernen von prädiktiven Modellen und gute Prädiktionen weit in die Zukunft erschweren. Mit den Verfügbaren Sinnen sind kann der wahre Zustand der Umgebung nur eingeschränkt erfasst werden. Beim autonomen Fahren könnte ein Fußgänger z. B. zeitweise durch ein Hindernis verdeckt sein und somit nichtmehr direkt wahrnehmbar sein. Dennoch muss der Agent intern eine Erinnerung an die Anwesenheit dieses Fußgängers haben um korrekt vorhersagen zu können, wann der Fußgänger hinter dem Hindernis auftauchen könnte und wie darauf zu reagieren ist. Neben der unvollständigen und streckenweise fehlerhaften Wahrnehmung der Umgebung, spielt Unsicherheit in der Modellierung eine große Rolle. Unsicherheit ergibt sich einerseits auf Grund mangelnder Erfahrung. Gewisse Situationen sind bisher unbekannt und können somit nicht treffend vorausgesagt werden. Andererseits ist Unsicherheit inhärent in der Umgebung. Der eben erwähnte Fußgänger könnte sich spontan schneller bewegen oder in eine andere Richtung gehen und somit völlig unterschiedliche zukünftige Systemreaktionen bewirken. Unsicherheit und Fehler in der Wahrnehmung und Prädiktion wirken sich entsprechend auf alle folgende Prädiktionen aus und können katastrophale Fehler in Langzeitprädiktionen bewirken.

> *„Methoden des Machine Learning ermöglichen es mit skalierbaren Algorithmen anhand großer Datensätze bessere Heuristiken für komplexe Probleme zu finden.“*
>
> *Andreas Dörr*

Gerade im geschlossenen Kreislauf der Interaktion zwischen dem Agent und der Umgebung, kann die Unsicherheit im prädiktiven, inneren Modell des Agenten jedoch auch eine wichtige Rolle übernehmen. Anstatt sich zu 100% auf die Korrektheit seiner Prädiktionen zu verlassen, kann der Agent erkennen, in welchen Situationen zu wenig Erfahrung vorhanden ist und entsprechend vorsichtig handeln oder diese Situationen vermeiden. Ähnlich zur Diskussion über endliche, diskrete Entscheidungen vs. un-

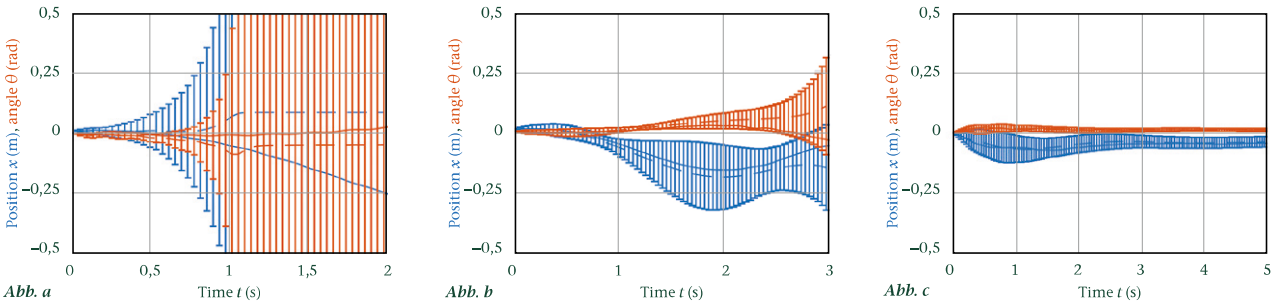

Abbildung 2:
Am Beispiel des Roboters Apollo, der das invertierte Pendeln balancieren soll, visualisieren wir den Lernfortschritt des internen Models. Zu Beginn ***(Abb. a)*** *ist wenig Wissen über die Umgebung bekannt und die Prädiktion über die zukünftige Hand- und Pendel Position wird sehr schnell sehr unsicher. Im Verlaufe des Lernprozesses werden die Prädiktionen akkurater* ***(Abb. b)****. Schließlich lernt der Roboter erfolgreich das Pendel in der Mitte zu balancieren* ***(Abb. c)****. Neben der Prädiktion (schraffierter Bereich) ist die echte Position des Pendels und der Hand in dem jeweiligen Lernfortschritt gezeigt (durchgezogene Linie).*

endliche, kontinuierliche Entscheidungen, eröffnet Unsicherheit und deren Beschreibung eine weitere Problemdimension.

Lernen von prädiktiven Modellen der Umgebung

Aufgabe des Agenten ist es nun ein Modell seiner Umgebung zu erlernen, mit Hilfe dessen er bewerten kann ob eine Sequenz von zukünftigen Aktionen ihn möglichst nahe (d. h. gut oder schnell) an sein Ziel führt. In den einfachsten Modellen wird dazu anhand des aktuellen Zustands des Systems und einer aktuellen Aktion eine Abbildung auf den nächsten Zustand des Systems erlernt. Diese Abbildung kann z. B. durch sogenannte neuronale Netze oder zum Beispiel auch durch Gauss Prozesse realisiert werden. Beides sind Möglichkeiten um datenbasiert Funktionen zu erlernen, die gegebene Input (in diesem Fall: aktueller Systemzustand und Agenten Aktion) und Output (in diesem Fall: nächster Systemzustand) erklären. Innerhalb der Modellierung gibt es mehrere Aspekte, die speziell für das Reinforcement Learning Problem und dabei speziell für reale Systeme hohe Relevanz haben. Im Rahmen dieser Arbeit lag das Hauptaugenmerk dabei auf: i) Langzeit Prädiktionen über das zukünftige Systemverhalten, ii) Repräsentation von Unsicherheit in der Modellprädiktion, iii) Ermittlung von Umgebungsvariablen, die nicht direkt über einen Sensor wahrgenommen werden können und iv) dem Einfluss des Agentenverhaltens auf die beobachteten Umgebungseinflüsse. Unter Berücksichtigung dieser Elemente, lassen sich schließlich Lernverfahren für prädiktive Modelle ableiten, die ideal auf das eigentliche Problem, ideales Verhalten in einer unbekannten Umgebung zu erlernen, zugeschnitten sind.

Modelle für langfristige Prädiktionen

Anhand der gesammelten Interaktionsdaten, werden in den einfachsten Modellen lediglich ausgehend von der aktuellen Umgebung und einer gewählten Aktion, der Zustand der Umgebung im nächsten Moment prädiziert. Um langfristigere Prädiktionen zu ermöglichen, muss dabei ausgehend von der ersten Prädiktion, eine weitere Prädiktion erfolgen.

Durch dieses aneinanderreihen von Prädiktionen, die alle mit Modellfehlern behaftet sind, kann es zum exponentiellen Ansammeln von Fehlern und katastrophalen Prädiktionen kommen. Gerade bei Reinforcement Learning Problemen ist jedoch eine akkurate, langfristige Prädiktion erforderlich, da bei vielen Problemen überhaupt erst nach einer gewissen Zeit entschieden werden kann, ob das ausgeführte Verhalten zielführend war. Bei Spielen wie Schach oder Go ist z. B. erst im letzten Zug endgültig geklärt, wer das Spiel gewinnt.

Modellierungsverfahren, die im Rahmen dieser Arbeit weiterentwickelt wurden, sind daher dafür ausgelegt direkt längere Sequenzen in die Zukunft zu prädizieren [2, 3]. Diese Modelle werden trainiert um möglichst akkurat die Verteilung von bisher beobachteten Systemverläufen über den gesamten betrachteten Horizont abzubilden.

Repräsentation von Unsicherheit

Im Reinforcement Learning gibt es zwei grundsätzliche Quellen der Unsicherheit. Da der Agent nach und nach über die Umgebung lernt, ist sein Wissen zunächst notwendigerweise lückenhaft. Die üblichen, daten-basierten Modelle haben jedoch keine Möglichkeit zu erkennen, ob eine bestimmte Vorhersage in einer Region getroffen wurde, in der Erfahrungen, d. h. Trainingsdaten vorliegen, oder ob die Vorhersage allein auf der Extrapolation anderer Erfahrungen beruht. Für den Agenten ist es jedoch essentiell für die Optimierung seines Verhaltens Informationen über die Güte des Modelles zu haben. Diese Information erlaubt es explizit Region mit ungenügendem Vorwissen zu vermeiden, wenn ein vorsichtiges und sicheres Handeln erforderlich ist. Andererseits können diese Bereiche aber auch explizit angesteuert werden, etwa um neugieriges, neues entdeckendes Verhalten zu fördern.

Die zweite Quelle von Unsicherheit ist in der Umgebung intrinsische Stochastizität. Eine Vielzahl von Prozessen in echten Systemen kann nur näherungsweise deterministisch beschrieben werden, sodass stets eine gewisse Unsicherheit verbleibt. Auch hier ist es für den Agenten wesentlich nicht pauschal einer bestimmten Ausprägung der Umgebung zu vertrauen, sondern eine Vorstellung über verschiedene mögliche zukünftige Entwicklungen zu haben.

Der Roboter Apollo mit der Aufgabe das invertierte Pendeln zu balancieren ist wiederrum ein Beispiel für beide Arten von Unsicherheit. Wie in *Abbildung 2* gezeigt, ist zu Beginn der Lernphase wenig Erfahrung über das Verhalten des Pendels vorhanden. Das interne Modell zeigt daher große Unsicherheit über die zukünftige Position und Winkel des Pendels. Mit fortschreitendem Lernen wird das Modell immer sicherer und kann präziser die Bewegungen und auch die Stabilisierung des Pendels prädizieren. Gleichzeitig verbleibt eine Restunsicherheit, die stochastischen und nicht-modellierten Prozessen geschuldet ist. Diese könnte z. B. von Reibungsprozessen in den Motorgetrieben stammen.

Erlernen von nicht beobachteten Umgebungseinflüssen

In realen Systemen mit ihren unzähligen, komplexen Wirkzusammenhängen, können in der Regel nicht alle Einflüsse und Größen direkt gemessen werden. Trotzdem können aus den verfügbaren Messungen oft Rückschlüsse auf andere, sozusagen indirekt beobachtete Systemzustände, getroffen werden. Am Beispiel des Fußgängers, der von einem Hindernis verdeckt wird, wäre der Aufenthaltsort oder seine Bewegungsrichtung und Geschwindigkeit ein solcher nicht direkt beobachtbarer Zustand. Anhand der zuletzt beobachteten Bewegungsrichtung und Geschwindigkeit, kann man Prognosen über das zukünftige Verhalten des Fußgängers erstellen. Dies erfordert jedoch ein Modell über das Verhalten des Fußgängers, ein sogenanntes Modell der latenten Systemdynamik. Zusätzlich sollte dieses Modell wiederspiegeln, dass mit Prognosen in die weitere Zukunft, die Unsicherheit über das wahre Verhalten des Fußgängers immer größer wird.

Für den lernenden Agenten ergeben sich nun gleich mehrere Probleme. Die Anzahl als auch die Bedeutung dieser verdeckten, oder auch latenten Systemzustände ist in der Regel unbekannt und muss aus den Daten erschlossen werden. Zusätzlich ist die weitere zeitliche Entwicklung dieser Zustände zwangsläufig ebenso unbekannt. Schließlich muss in geeigneter Weise die Unsicherheit über die wahre Natur und die wirkliche Entwicklung dieser latenten Zustände ausgedrückt werden.

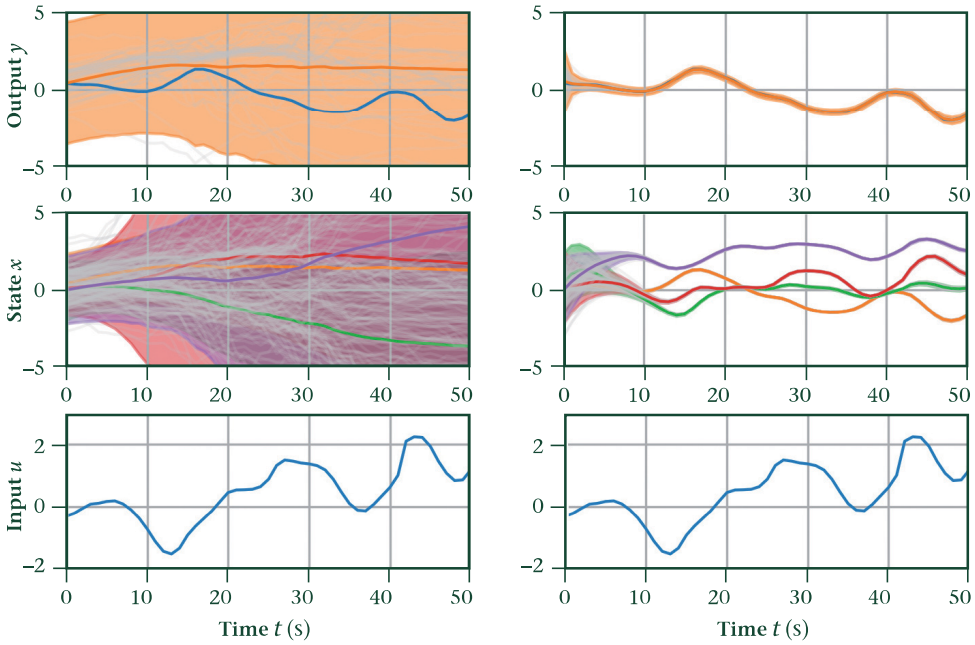

Abbildung 3:
Visualisierung des Lernfortschritts für prädiktive Modelle bei Systemen mit unbeobachteten Zustands-größen. Obwohl nur der ***Input*** *(Aktion) zum System und eine Messgröße* ***(Output)*** *gemessen werden können, gelingt es dem lernenden Agenten trotz der anfänglichen Unsicherheit, die Bedeutung und die dynamische Entwicklung der verdeckten Zustandsgrößen* ***(State)*** *im System zu erlenen.*

Einflüsse des eigenen Verhaltens auf die Umgebung

In den allermeisten, realen Problemen ist die Anzahl an Handlungsmöglichkeiten derartig groß, dass keine optimale Lösung mehr berechnet werden kann. Selbst für wenige Stellgrößen wäre der Lösungsraum viel zu groß, sobald der Agent wirklich zufällige Aktionen in jedem Zeitschritt wählen könnte. In der Regel ist „sinnvolles“ Verhalten jedoch nur ein sehr eingeschränkter Bereich in diesem Raum aller zufällig generierbarer Verhaltensmuster. Im Reinforcement Learning können wir diese Einschränkung des Suchraumes direkt nutzen. In der Regel wird Verhalten durch eine parametrisierte Funktion, eine sogenannte Policy, beschrieben.

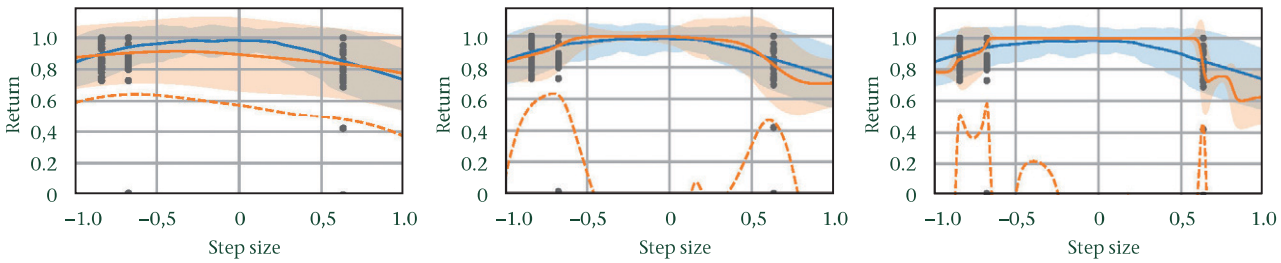

Abbildung 4:
Für verschiedene Handlungsstrategien (auf der x-Achse von -1 bis 1) prognostiziert der Agent den erwarteten Erfolg (orange), im Vergleich zum realen Erfolg der jeweiligen Strategie (blau). Obwohl der Agent nur wenige Strategien ausprobiert hat (graue Punkte) kann er mittels seiner Modellannahme Prognosen für unbekannte Strategien erstellen. Je nach Modellannahmen reichen diese Prognosen weit in unbekannte Regionen oder enden kurz nach den bisherigen Strategien.

Im Lernprozess werden optimale Parameter gesucht, auf Basis des erhaltenen Rewards. Sowohl im Erlernen des Modells der Umgebung, als auch in der Optimierung des Verhaltens kann das Wissen über die Klasse oder Art dieser Verhaltensregeln direkt benutzt werden. In dieser Arbeit wurde eine Methode abgeleitet, die direkt die Verhaltensklasse im Modelllernen integriert. Auf diese Weise werden Modelle bevorzugt, die unter dem zu erwartenden Verhalten sinnvoll die Realität wiederspiegeln. Gegenüber allgemeinen Modellen, die auf beliebigen Verhaltensmustern korrekte Prädiktionen liefern müssen, können diese Modelle somit mit weniger Daten und somit deutlich schneller zu sinnvollen Prädiktionen kommen.

Gute Modelle und Modellannahmen sind essentieller Bestandteil um effizient aus der gesammelten Erfahrung zu lernen und Rückschlüsse für besseres zukünftiges Verhalten zu ziehen. Wesentlich ist dabei zu erkennen, in welchem Raum für ein bestimmtes Problem und für eine bisher gesammelte Menge von Erfahrungen, die besten bzw. stärksten Modellannahmen gerechtfertigt und möglich sind.

Modellannahmen hinterfragen

Ohne einige Annahmen zu treffen kann kein System aus seiner Erfahrung Rückschlüsse auf zukünftige Entwicklungen treffen. Für das maschinelle Lernen ist es daher unbedingt erforderlich sinnvolle Modellannahmen zu treffen, die es dem System ermöglichen in unbekannte Bereiche zu extrapolieren. Bildlich gesprochen könnte dies zum Beispiel bedeuten, dass die lernende Maschine weiß, dass das Auto, wenn in der Vergangenheit ein Lenkeinschlag nach rechts ein Rechtsabbiegen bewirkt hat, in der Zukunft ein etwas größerer Einschlag nach rechts ein etwas schärferes Rechtsabbiegen bedeuten könnte. Diese Annahmen werden üblicherweise über die Glattheit oder Kontinuität der entsprechenden Modelle abgebildet. Die dis-

kutierten, Modell-Basierten, Reinforcement Learning Verfahren haben somit einen großen Vorteil, wenn die realen Systeme diese Glattheitsannahmen tatsächlich erfüllen, sodass bereits mit wenigen Datenpunkten Aussagen über weit entfernte Bereiche im Verhalten des Systems möglich sind.

In der Tat gibt es jedoch zahlreiche Probleme, in denen die Systemdynamik stellenweise diskontinuierlich ist. Ein typisches Beispiel wäre ein laufender Roboter, dessen Dynamik der Beine sich schlagartig ändert wenn das zunächst frei schwingende mit dem Boden in Kontakt kommt. Eine wesentliche Forschungsfrage ist somit, welche Annahmen in welchem Modell gerechtfertigt sind. Diese Annahmen können von Problem zu Problem unterschiedlich und von der Menge an bisher gesammelter Erfahrung abhängig sein. Statt die System-Dynamik zu modellieren, gibt es andere Klassen von RL Methoden, die andere Modelle erlernen und somit andere Annahmen erfordern. In einer weiteren Veröffentlichung [4], wurde daher untersucht, wie daten-effizientes Lernen in einer dieser Klassen möglich ist und vor allem, wie sich Modellannahmen von einem Bereich übersetzen lassen in einen anderen Bereich.

In diesem Fall werden Annahmen lediglich übe die elementaren Aktionen, d. h. Handlungsmöglichkeiten getroffen. In vielen realen Problemen sind die Handlungsmöglichkeiten gut verstanden und mittels Vorwissen über das konkrete Problem können Abschätzungen getroffen werden, was ähnliche oder verwandte Handlungsmöglichkeiten sind. Basierend allein auf dieser Modellannahme kann mit den vorgeschlagenen Methoden abgeschätzt werden, wie erfolgsversprechend unterschiedliche langfristige Vorgehensweisen sind. In *Abbildung 4* ist für verschiedene Strategien (x-Achse) der langfristige akkumulierte Reward (blau) dargestellt. Basierend auf wenigen Versuchen (graue Datenpunkte) kann der Agent mittels seiner Modellannahmen Abschätzungen für bisher unversuchte Strategien erhalten. Je glatter, d. h. verwandter die einzelnen Handlungen miteinander sind, umso weiter kann der Agent aus seinem bisherigen Wissen extrapolieren. Falls sehr unterschiedliche Aktionen betrachtet werden, verliert die bisherige Erfahrung relativ schnell an Bedeutung und die Prädiktion (orange) wird deutlicher schneller unsicher, wenn sich der Agent von seiner bisherigen Erfahrung entfernt.

> *„No-free-lunch: Ohne Modell-Annahmen sind keinerlei generalisierenden Aussagen auf Basis der bisherigen Erfahrung möglich.“*
>
> *Andreas Dörr*

Fazit

Lernende und daten-basierte Verfahren haben in den letzten Jahren eine Reihe von schwierigen Problemen, wie z. B. die

Computer-basierte Bild- und Videoanalyse oder das automatisierte Text- und Sprachverständnis, revolutioniert. In diesen Bereichen wurden unter Verwendung großer Datenmengen und mit Hilfe von massiver Rechnerleistung und skalierbaren Methoden neue Lösungen gefunden, deren Performance die der bisherigen, manuell entworfenen Lösungen weit übersteigt. Dies ist möglich da für diese Probleme durch die Digitalisierung große Datensätze einfach verfügbar sind. Zusätzlich wurden einige wenige anwendungsspezifische Invarianten und Modellannahmen gefunden, die das Lernproblem für diese Klasse an Problemen erheblich vereinfachen.

Anders stellt sich die Lage für die hier beschriebene, sequentielle Interaktion eines Systems mit seiner zunächst unbekannten Umgebung dar. Für viele dieser Entscheidungs- oder Regelungsproblemen ist die Datenverfügbarkeit deutlich schlechter. Zusätzlich eröffnet sich mit jeder neuen Entscheidung ein potentiell neuer Bereich der Umgebung mit bisher unbekanntem Verhalten. Obwohl lernende Verfahren für einige simulierte Umgebungen bereits sehr gute Lösungen finden (z. B. sind sie in der Lage, bei Spielen wie Go oder Schach selbst die besten menschlichen Spieler zu schlagen), sind diese Methoden nicht direkt auf reale Problemstellungen übertragbar.

In der vorliegenden Arbeit wurden daher Ansätze und Methoden entwickelt, um lernende Verfahren in realen, industriellen Anwendungen einzusetzen. Dies erfordert schnelles, d. h. daten-effizientes, und gleichzeitig robustes, d. h. in allen Situationen zuverlässiges, Erlernen von neuen Verhaltensmustern. Die entwickelten Methoden ermöglichen es prädiktive Modelle zu erlernen, die langfristige Entwicklungen vorhersagen können. Diese Modelle können dann eingesetzt werden um langfristig optimales Vorhalten vorherzusagen und umzusetzen.

Referenzen

[1] Doerr et al., Model-Based Policy Search for Automatic Tuning of Multivariate PID Controllers, ICRA 2017

[2] Doerr et al., Optimizing Long-Term Predictions for Model-Based Policy Search, CORL 2017

[3] Doerr et al., Probabilistic Recurrent State-Space Models, ICML 2018

[4] Doerr et al., Trajectory-Based Off-Policy Deep Reinforcement Learning, submitted to ICML 2019