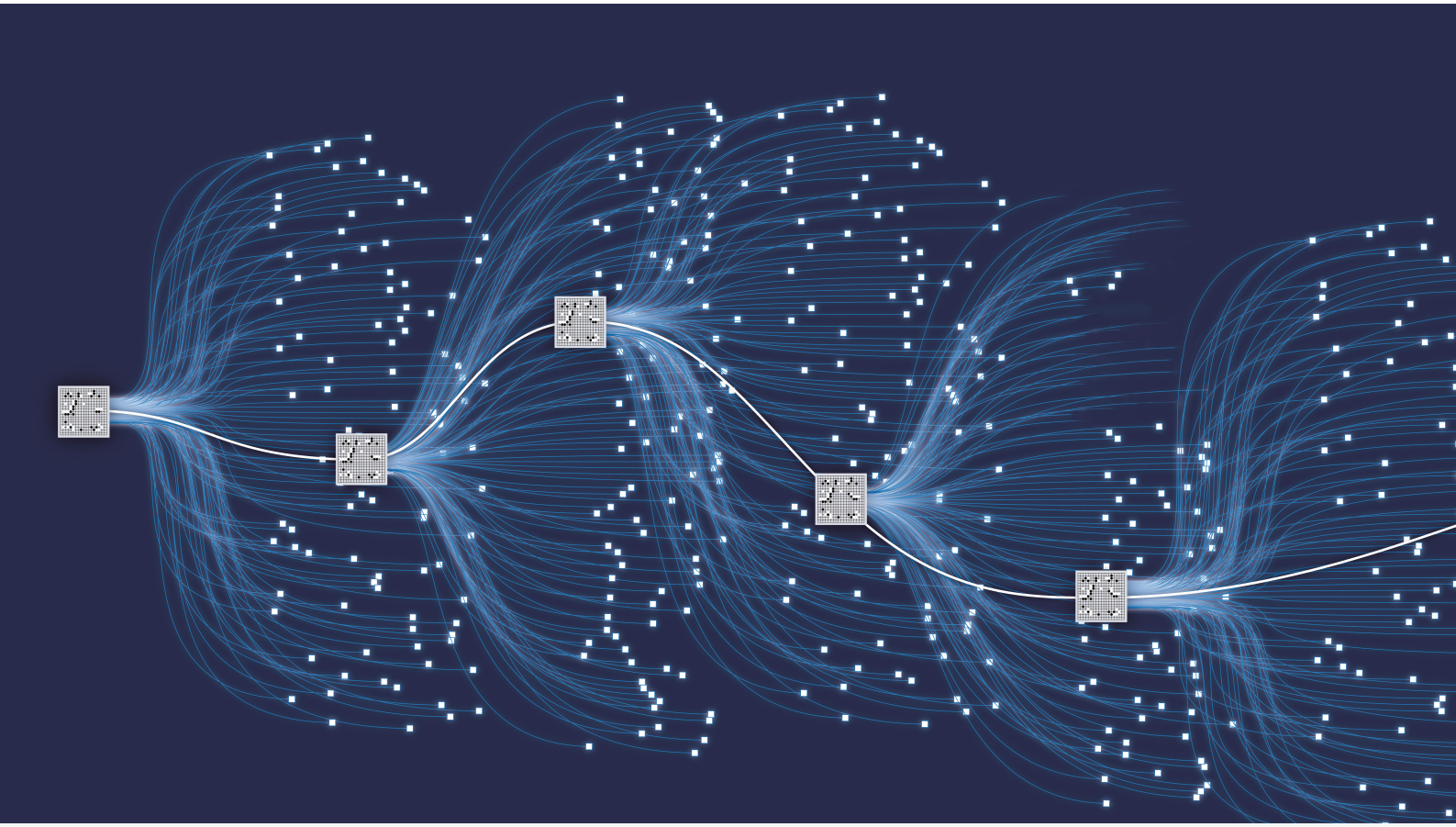

Erfolgsstrategie vom KI-Agenten

Dieser Artikel gibt den Stand zum Zeitpunkt seiner Veröffentlichung wieder. Der Inhalt kann inzwischen veraltet sein.

11/11/2021 Auf Basis von Deep Reinforcement Learing hat Porsche Engineering eine innovative Applikationsmethodik entwickelt. PERL bietet optimale Strategien für die Motorapplikation, wodurch der Zeit- und Kostenaufwand erheblich verkürzt wird.

Unter Experten gilt das Jahr 2016 als Meilenstein in der Geschichte der Künstlichen Intelligenz (KI). Von der breiten Öffentlichkeit in Europa und den USA weitgehend unbeachtet, trat damals das Computerprogramm AlphaGo gegen den südkoreanischen Weltklassespieler Lee Sedol im Brettspiel Go an und gewann vier der fünf Partien. Zum ersten Mal behielt damit ein Computer bei dem traditionellen asiatischen Strategiespiel die Oberhand. Bis dahin war es nicht gelungen, einer Software die komplexe Strategie des Brettspiels beizubringen – die erforderliche Rechenleistung und Rechenzeit wären zu groß gewesen. Den Wendepunkt brachte das Training der KI des GoComputers mithilfe des Deep Reinforcement Learning.

Eine der KI-Königsdisziplinen

Die noch recht junge Methode Deep Reinforcement Learning gilt als eine der Königsdisziplinen der KI. Erst durch neue, leistungsfähige Hardware wurde es in den vergangenen Jahren möglich, sie breiter einzusetzen und in Anwendungen praktische Erfahrungen zu sammeln. Deep Reinforcement Learning ist ein selbstlernendes KI-Verfahren, das die klassischen Methoden von Deep Learning mit denen des Reinforcement Learning verbindet. Die Grundidee: Der Algorithmus (unter Fachleuten „Agent" genannt) interagiert mit seiner Umgebung und wird für Aktionen, die zu einem guten Ergebnis führen, mit Bonuspunkten belohnt und bei Misserfolgen mit Abzügen bestraft. Ziel ist es, so viele Belohnungspunkte wie möglich zu erhalten.

Dazu entwickelt der Agent im Laufe der Trainingsphase eine eigene Strategie, an die er sich durch ein einfaches Trial-and-Error-Verfahren herantastet. Die Trainingsvorlage gibt dem System dabei für verschiedene Situationen oder Zustände Startund Zielparameter vor. Das System sucht dann nach verschiedenen Möglichkeiten, um vom Istzum Sollwert zu kommen. Für jeden Schritt approximiert es mittels eines neuronalen Netzes die Höhe der jeweiligen Belohnungen. Der Agent abstrahiert die Ergebnisse und erstellt daraus für die folgenden Rechenoperationen die Prognose, welche Maßnahme in einer bestimmten Situation oder einem Zustand das beste Ergebnis liefern wird. Daraus ergibt sich dann seine Methodik, die sogenannte „Policy", die er nach Abschluss der Lernphase auf andere Berechnungen anwendet.

Anders als bei anderen Arten der KI, wie dem Supervised Learning, bei dem aus Paaren von Eingangs- und Ausgangsdaten gelernt wird, oder dem Unsupervised Learning, das auf die Mustererkennung zielt, trainiert Deep Reinforcement Learning langfristige Strategien. Denn das System lässt auch kurzfristige Rückschläge zu, wenn sich dadurch die Chancen für den künftigen Erfolg erhöhen. Gegen das so trainierte Computerprogramm AlphaGo hatte auch ein Meister auf dem Niveau von Sedol schließlich keine Chance mehr.

Einsatz in der Motorapplikation

Die Leistung des Deep Reinforcement Learnings beim Brettspiel brachte die Experten von Porsche Engineering auf die Idee, die Methode für komplexe Applikationsaufgaben im Fahrzeugbereich einzusetzen. „Denn auch hier ist die beste Erfolgsstrategie gefragt, um eine optimale Systemabstimmung zu erzielen", sagt Matteo Skull, Ingenieur bei Porsche Engineering. Das Ergebnis ist ein gänzlich neuer Kalibrierungsansatz: Porsche Engineering Reinforcement Learning (PERL). „Mithilfe des Deep Reinforcement Learnings trainieren wir bei PERL den Algorithmus darauf, nicht nur einzelne Parameter zu optimieren, sondern sich die Strategie zu erarbeiten, mit der er ein optimales Applikationsgesamtergebnis für eine ganze Funktion erzielt", so Skull. „Vorteile sind die hohe Effizienz der Methodik, da sie selbstlernend ist, und die universelle Anwendbarkeit auf viele Entwicklungsbereiche des Fahrzeugs."

Die Anwendung der PERL-Methodik lässt sich grundsätzlich in zwei unterschiedliche Phasen unterteilen: Erst erfolgt das Training, danach kommt der eigentliche Applikationseinsatz am realen Motor. Als Beispiel führt Skull das Drehmomentmodell an, mit dem die Motorsteuerung für jeden Betriebspunkt das aktuelle Drehmoment an der Kurbelwelle berechnet. In der Lernphase benötigt PERL als einzigen Input den Messdatensatz eines alten Projekts, etwa von einem Vorgängermotor. „Dabei ist PERL höchst flexibel, denn Parameter wie Motorbauform, Hubraum oder Aufladesystem haben keinen Einfluss auf den Lernerfolg. Wichtig ist nur, dass sowohl die Trainings- als auch die spätere Zielapplikation denselben Modellansatz nutzen, damit der Algorithmus die Ergebnisse richtig umsetzt", sagt Skull.

Während des Trainings lernt das System die optimale Applikationsmethodik für die Kalibrierung des vorgegebenen Drehmomentmodells. An charakteristischen Stellen im Kennfeld vergleicht es dazu den kalibrierten Wert mit dem Wert aus dem Messdatensatz und approximiert anhand der resultierenden Belohnungen mithilfe neuronaler Netze eine Nutzfunktion. Durch das erste neuronale Netz können dabei Belohnungen für bis dato unbekannte Zustände abgeschätzt werden. Ein zweites neuronales Netz, das sogenannte Policy Network, prognostiziert dann, welche Aktion in einem bestimmten Zustand den höchsten Nutzen bringen wird.

Permanente Prüfung der Ergebnisse

Auf dieser Grundlage erarbeitet sich PERL die Strategie, mit der es am besten vom Ist- zum Sollwert gelangt. Ist das Training abgeschlossen, steht PERL für die eigentliche Applikationsaufgabe am Motor bereit. Während des Motorprüfstandversuchs berechnet der Algorithmus in Echtzeit mithilfe der antrainierten Abstimmungsstrategie die optimale Drehmomentkalibrierung, die unmittelbar wieder am Motor getestet und verifiziert wird. Im Laufe des Applikationsvorgangs überprüft das System die eigenen Ergebnisse und passt diese an, etwa wenn die Parametervariation an einer Stelle des Kennfelds Rückwirkungen auf eine andere hat.

„Zudem können wir PERL sowohl die Berechnungsgenauigkeit der Drehmomentkurve als auch einen Glättungsfaktor für die Interpolation der Werte zwischen den berechneten Stützstellen vorgeben. Damit machen wir die Applikation robust gegenüber dem Einfluss von Fertigungstoleranzen oder dem Verschleiß der Motorbauteile über die Lebensdauer", erklärt Dr. Matthias Bach, Leiter Fachdisziplin Motor Applikation und Mechanik bei Porsche Engineering.

Die Leistungsfähigkeit von PERL soll künftig dabei helfen, den stark zunehmenden Applikationsaufwand als eine der größten Herausforderungen bei der Entwicklung neuer Fahrzeuge zu bewältigen. Prof. Dr. Michael Bargende, Inhaber des Lehrstuhls Fahrzeugantriebe am Institut für Fahrzeugtechnik der Universität Stuttgart und Vorstand des Forschungsinstituts für Kraftfahrwesen und Fahrzeugmotoren Stuttgart (FKFS), erklärt das Problem am Beispiel des Antriebs: „Der Trend zur Hybridisierung und die anspruchsvolleren Abgastests haben die Zahl der Applikationsparameter weiter steigen lassen. Die Diversifizierung der Antriebsstränge und der Märkte sowie die Veränderungen im Zertifizierungsprozess haben zudem die Anzahl an Applikationen erhöht, die erstellt werden müssen." Bargende ist vom

Potenzial der neuen Methodik überzeugt: „Reinforcement Learning ist ein künftiger Schlüsselfaktor im Bereich Motor- und Antriebsstrangapplikation."

Applikationsaufwand deutlich reduzieren

Mit den heutigen konventionellen Werkzeugen wie der modellbasierten Applikation erfolgt die automatisierte Bedatung der einzelnen Parameter – etwa der Kennfelder im Motormanagement – in der Regel nicht optimal und muss vom Applikationsingenieur manuell überarbeitet werden. Zudem macht jede Hardware-Variation am Motor während der Entwicklung eine Anpassung der Applikation notwendig, obwohl sich die Software nicht geändert hat. Qualität und Dauer der Kalibrierung hängen dadurch stark vom Geschick und den Erfahrungen des Applikateurs ab.

„Der aktuelle Applikationsprozess ist mit erheblichem Zeit- und Kostenaufwandverbunden. Für die kennfeldabhängige Berechnung eines einzigen Parameters, etwa der Zylinderfüllung, muss man heute etwa vier bis sechs Wochen Entwicklungszeit veranschlagen, verbunden mit hohen Prüfstandskosten", so Bach. Für die Gesamtapplikation einer Motorvariante ergibt sich ein entsprechend hoher Zeit- und Kostenaufwand. „Mit PERL können wir diesen Aufwand signifikant reduzieren", sagt Bach mit Blick in die Zukunft.

Zusammengefasst

Die innovative PERL-Methodik von Porsche Engineering erarbeitet mithilfe des Deep Reinforcement Learnings optimale Strategien für die Motorapplikation (die sogenannte „Policy"). Experten betrachten den neuen KI-basierten Ansatz als Schlüsselfaktor, um die steigende Komplexität im Bereich der Motoren und Antriebe auch in Zukunft noch beherrschen zu können.

Info

Text: Richard Backhaus
Mitwirkende: Matteo Skull, Dr. Matthias Bach

Text erstmalig erschienen im Porsche Engineering Magazin, Nr. 1/2021

Dieser Beitrag wurde vor dem Start des Porsche Newsroom Schweiz in Deutschland erstellt. Die genannten Verbrauchs- und Emissionsangaben richten sich daher nach dem Prüfverfahren NEFZ und wurden unverändert übernommen. Alle in der Schweiz gültigen Angaben nach WLTP-Messzyklus sind unter www.porsche.ch verfügbar.

MEDIA ENQUIRIES

Sandro Kälin

Head of Communications Porsche Schweiz AG
+41 41 487 91 16
sandro.kaelin@porsche.ch

Link Collection

Link to this article
https://newsroom.porsche.com/de_CH/2021/innovation/porsche-engineering-erfolgsstrategie-ki-agent-23310.html

External Links
https://www.porscheengineering.com/peg/en/