Deep Reinforcement Learning als Methode zur autonomen Steuerung von Niederspannungsnetzen mit Fokus auf die Netzstabilität

Lars Quakernack, Michael Kelker, Ulrich Rückert, Jens Haubrock

Fachhochschule Bielefeld, Interaktion 1 33619 Bielefeld Deutschland, +49.521.106-70341, lars.quakernack@fh-bielefeld.de, https://www.fh-bielefeld.de/ium/forschung/agnes

Kurzfassung:

Die Bemühungen um ein emissionsarmes Stromnetz in Deutschland stellt die Verteilnetzbetreiber vor Herausforderungen. Dazu gehören der Aufbau volatiler, dezentraler Erneuerbare-Energien-Anlagen (DEA), wie z.B. Photovoltaik, und der stetige Ausbau der Elektromobilität mit zum Teil hohen Ladeleistungen in der Niederspannung. Um die Netzstabilität zu erhöhen und einen kostenintensiven Netzausbau zu vermeiden, muss ein Steuerungssystem implementiert werden. Ein neuer Weg zur Entwicklung einer solchen Steuerung ist der Einsatz von Reinforcement Learning (RL). Dies kann aufgrund aktueller Forschung kosteneffizienter und leistungsfähiger sein kann als herkömmliche Regler. Darüber hinaus zeigen J. Duan, D. Didden und M. Kelker, dass das allgemeine Prinzip von RL funktioniert, um Spannungsbandverletzungen zu reduzieren. Außerdem wird eine Verringerung der Transformatorleistung bei zunehmender Nutzung von PV-Energie festgestellt. Für diese Ergebnisse wird jedoch eine große Menge an Daten verwendet. Außerdem wurde der Algorithmus nicht im Feld getestet. Die Genauigkeit der von den Autoren vorgestellten RL- Systeme hängt von dem Detailgrad der jeweiligen Simulation ab. Der Beitrag gibt einen Überblick über den aktuellen Stand der Technik mit Kritik zu der autonomen Steuerung mit RL.

Keywords: Reinforcement Learning, Niederspannung, Netzstabilität, autonome Steuerung

1 Einleitung

Um das im Pariser Klimaabkommen festgelegte 1,5°C-Ziel zu erreichen, muss ein Wechsel von fossilen zu erneuerbaren Energiequellen stattfinden. Ein wichtiger Teil dieses Übergangs ist der Verkehrssektor. Dieser muss zunehmend elektrifiziert werden. Nach Angaben der Bundesregierung sollen in Deutschland bis 2030 7-10 Millionen Elektrofahrzeuge (EVs) zugelassen werden. Ein weiterer großer Teilbereich ist die dafür notwendige Energieversorgung. Aufgrund des geplanten Ausstiegs aus der Kohle- und Atomkraft in Deutschland wird der Anteil der erneuerbaren Energien weiter steigen. Die Bundesregierung plant bis 2025, den Anteil der erneuerbaren Energien auf 40-45% zu erhöhen. Die Durchdringung der Elektromobilität und der stetige Ausbau der dezentraler Erneuerbare-Energien-Anlagen (DEA) stellen neue Herausforderungen an das Stromnetz. So muss zum einen die vergleichsweise hohe Leistung in der Verteilnetzebene für die Elektrofahrzeuge bereitgestellt werden und zum anderen die Volatilität der DEAs berücksichtigt werden. Ungesteuert kann dies zu Einbußen in der Netzstabilität führen und die Versorgungssicherheit gefährden. Dies gilt insbesondere für die Niederspannungsebene, wo massiv Energiemengen eingespeist und verbraucht werden [1, 2].

Um Einbußen bei der Netzstabilität zu vermeiden, muss entweder ein Steuerungssystem für Verbraucher und Erzeuger eingeführt oder das Stromnetz massiv ausgebaut werden. Da letzteres kostenintensiver ist, forschen Wissenschaftler an Steuerungsalgorithmen, um diesen Problemen zu begegnen. Ein vielversprechender Ansatz ist der Einsatz von künstlicher Intelligenz (KI) - genauer gesagt durch den Einsatz von Reinforcement Learning (RL). In diesem Beitrag wird RL als Methode zur Steuerung von Niederspannungsnetzen analysiert. Dazu wird das Verfahren theoretisch erläutert und die Anpassung an das Niederspannungsnetz mit dem aktuellen Stand der Technik dargestellt. Darüber hinaus werden die Schwachstellen der Technik aufgezeigt.

2 Methodik

RL ist eine der Methoden des maschinellen Lernens. RL verwendet positive oder negative Verstärkungen, um einen Agenten auf der Grundlage von Zuständen in einer Umgebung zu trainieren, die beste Aktion zu wählen, um ein definiertes Optimum zu erreichen. Im Fall der elektrischen Netze könnte die Umgebung das Niederspannungsnetz sein, der aktuelle Zustand könnte der Lastfluss im Netz sein, die positiven und negativen Verstärkungen (Belohnungen) könnten auf der Überlastung des Transformators oder den Spannungsregelungen basieren (Überlastung des Transformators = negative Belohnung, normaler Zustand des Transformators = positive Belohnung) und der Agent könnte ein Controller sein, der die Lade- oder Entladeleistung einer Batterie ändert [3].

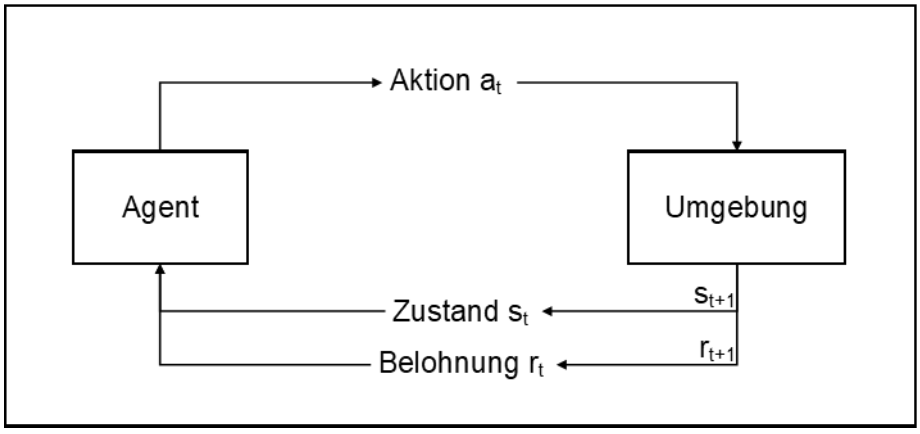

Abbildung 1: Schematische Darstellung von RL

Der allgemeine Prozess von RL ist in Abbildung 1 dargestellt. Ein Agent führt eine Aktion a_t aus, die auf seiner gelernten Strategie basiert. Auf der Grundlage des neuen Zustands s_{t+1} wird die Belohnung r_{t+1} berechnet. Der neue und der alte Zustand, die Aktion und die Belohnung werden gespeichert, und der Prozess beginnt von Neuem mit einer Aktion des Agenten. Das Lernen, die beste Aktion für den Agenten zu wählen, charakterisiert die Methode des RL. Im Folgenden werden Q-Learning und Deep Q -Learning (DQN) als die gängigsten Verfahren zur autonomen Steuerung in Niederspannungsnetzen näher erläutert [3, 4].

2.1 Q-Learning

Im Lernprozess des Q-Learnings wird in jedem (Zeit-)Schritt ein Q-Wert berechnet und in sogenannten Q-Tabellen gespeichert. Für einen Aktionsraum von zwei Batteriezuständen (Laden und Entladen) und 96 Zuständen (15min-Werte eines Tages) ist die Größe der Q-Tabellen also 2x96. Die Q-Werte werden zufällig initialisiert und iterativ während des Trainings gemäß der Bellman-Gleichung (1) aktualisiert. Der höchste Q-Wert in einer Reihe für einen Zustand in der Tabelle bestimmt die Aktion. Aufgrund der zufälligen Initialisierung führt der Agent zunächst zufällige Aktionen aus, um die Umgebung mit seinen Belohnungen zu erkunden. Langfristig passt der Agent seine Aktionen entsprechend dem oben beschriebenen Prozess an [4, 5].

$$Q_{new}(s_t,a_t) \leftarrow Q_{old}(s_t,a_t) + \alpha \times [r_t + \gamma \times \max(Q(s_{t+1},a_t) - Q_{old}(s_t,a_t)] \quad (1)$$

In Gleichung (1) berechnet sich der neue Q-Wert Q_{new} aus der gewichteten Belohnung, dem geschätzten maximalen Q-Wert und dem alten, gewichteten Q-Wert Q_{old}. Die Hyperparameter bestimmen maßgeblich das Lernverhalten. Die Learning rate α beschreibt das Ausmaß, in dem neue Erfahrungen des Agenten alte überlagern. Bei einer Learning rate von eins wird nur der letzte (Zeit-)Schritt berücksichtigt, bei einer Learning rate von null werden alle vorherigen (Zeit-)Schritte berücksichtigt. Der Discountfactor γ bestimmt die Bedeutung zukünftiger Belohnungen. Bei einem Faktor von nahezu eins wird die höchste langfristige Belohnung angestrebt. Bei einem Faktor von nahezu null werden nur die kurzfristigen Belohnungen berücksichtigt. Klassischerweise kann man sich Spiele wie Super Mario vorstellen, bei denen das Level mit wenigen Münzen und vermutlich kurzer Zeit ($\gamma \approx 0$) oder mit vielen Münzen und längerer Zeit ($\gamma \approx 1$) abgeschlossen werden kann, wenn das Ziel ist am meisten Münzen zu sammeln [4, 5].

2.2 Deep Q-Learning

DQN unterscheidet sich vom Q-Learning durch die Verwendung eines neuronalen Netzes zur Vorhersage der Q-Werte anstelle der Nutzung der Q-Tabellen. Die Verwendung von Q-Tabellen führt in komplexeren Umgebungen zu Speicherproblemen, da jede Aktion/Zustandskombination aufgezeichnet werden muss. Um einen vollständigen Überblick über das Thema zu geben, wird eine kurze Einführung in Deep Neural Networks gegeben.

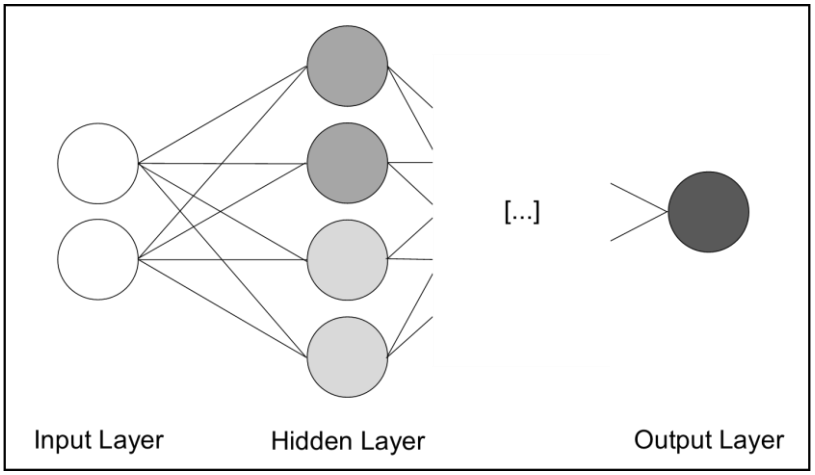

Abbildung 2 Konzept eines deep neural networks

Deep Neural Networks bestehen aus einer Eingabe- und Ausgabeschicht und mindestens einer verborgenen Schicht. In Abbildung 2 ist die Struktur von Deep Neural Networks zu sehen. Bei den Schichten handelt es sich meist um vollständig verbundene Neuronen, die ähnlich wie die biologischen Neuronen zur Identifizierung und Speicherung von Informationen dienen. In unserem Fall ist der Eingangsparameter der Zustand der Umgebung - zum Beispiel die Knotenspannungen des elektrischen Netzes. Die Ausgabeschicht liefert uns die Q-Werte für jeden Parameter im Aktionsraum. Im Fall des Ladens und Entladens der Batterie sind dies Zwei. Die Eingabedimension entspricht also unserer Zustandsdimension und die Ausgabedimension entspricht der Dimension des Aktionsraums. Die versteckten Schichten können beliebige Dimensionen haben und sind zufälligen gewichtet. Während des Lernprozesses verschieben sich die Gewichte der verborgenen Schichten. Dies hat zur Folge, dass einige Verbindungen zwischen den Schichten stärker und einige schwächer sind. Dadurch wird festgelegt, welche Merkmale der Eingabeparameter für einen entsprechenden Q-Wert der Ausgabeschicht wichtiger sind. Kurz gesagt, wird der Eingang (Zustand) und die Ausgänge (Q-Werte für jede Aktion) des Deep Neural Networks definiert und auf Basis der zufällig initiierten Aktionen des Agenten für jeden (Zeit-)Schritt im RL Algorithmus trainiert [6, 7].

Im DQN wird auch zwischen Single Agent Reinforcement Learning (SARL) und Multi Agent Reinforcement Learning (MARL) unterschieden. In Abbildung 3 ist zu sehen, dass der einzelne Agent alle steuerbaren Objekte mit derselben Aktion koordiniert (SARL), während im Fall von MARL jeder Agent ein Objekt mit verschiedenen Aktionen steuert. In diesem Fall handelt es

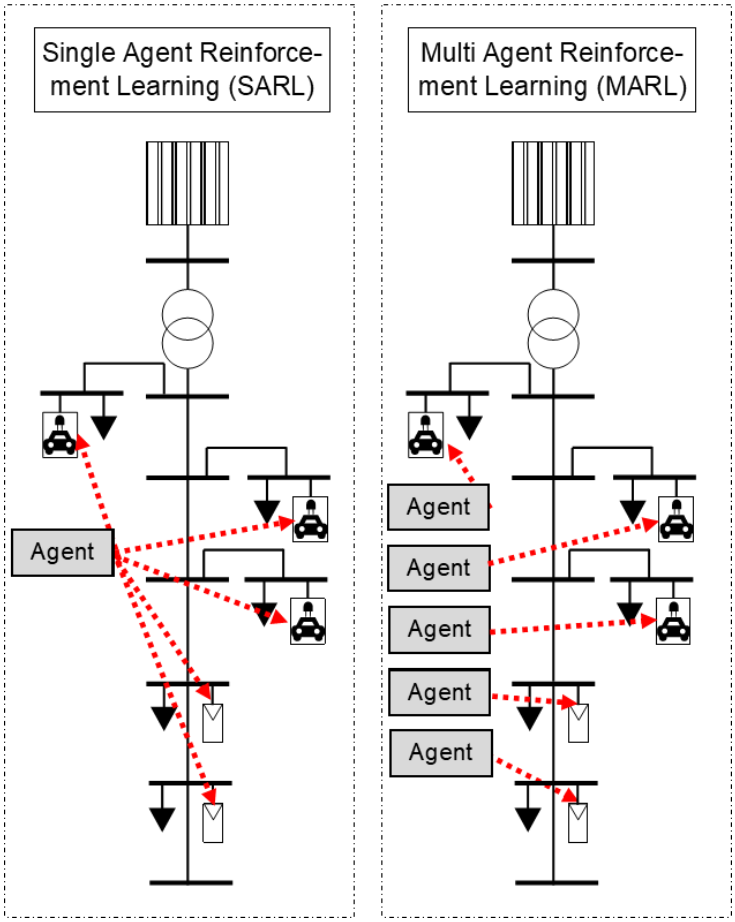

Abbildung 3 Single Agent RL vs. Multi agent RL im elektrischen Netz

sich bei den Objekten um Batterien und/oder Elektrofahrzeuge, und die Aktionen könnten das Laden oder Entladen sein [8].

3 RL in Niederspannungsnetzen

Wie in der Einleitung beschrieben, wird die Verwendung von RL als Controller in Niederspannungsnetzen analysiert. Wie können RL Controller eingesetzt werden, um Überlastungen von Betriebsmitteln zu vermeiden, DEA durch Batteriespeicher auszugleichen oder Spannungsbandgrenzen einzuhalten und wie wird RL nach dem derzeitigen Stand der Technik für die Netzstabilität eingesetzt? J. Duan [9], D. Didden [10] und M. Kelker [11] haben diese Art der Steuerung in ihren Arbeiten untersucht. Dieser Abschnitt gibt einen Überblick über ihre Methodik und ihre Ergebnisse. In Tabelle 1 sind die Parameter der Arbeiten dargestellt.

Tabelle 1: Verschiedene DQN Parameter

Parameter	**Stand der Technik**		
	J. Duan [9]	**D. Didden [10]**	**M. Kelker [11]**
Methode	DDPG, DQN	DQN	DDQN, DQN
Umgebung	200-bus System	29 Knoten, PV, Wärmepumpen	12 Knoten, 2 Stränge, PV, Elektromobilität
Zustand	Spannungsdaten	Kalenderdaten, SoC Batterie, Nutzleistungsprognose	Strom- und Spannungsdaten, Zeit, alte Aktionen jedes Agenten
Reward	$R_{voltage}$	$R_{voltage}$, R_{line}, R_{trafo}, R_{losses}	$R_{voltage}$, R_{trafo}
Agent	SARL	SARL, MARL	MARL
Aktionen	Einstellwerte von Generatoren	Batterieleistung, Einschränkung der PV-Leistung	Ladeleistung, Batterieleistung

Zunächst einmal unterscheiden sich die Rahmenbedingungen in den drei Studien voneinander. J. Duan verwendete ein 200-Bus-System mit den Knotenspannungen als Zustände. D. Didden und M. Kelker sind sich ähnlicher, da sie ein kleineres Netz mit ungefähr 29 Knoten (D. Didden) und 12 Knoten (M. Kelker) verwenden und zusätzlich zu den Daten der Lastflussanalyse, einen zeitbestimmenden Faktor und einige spezifische Daten für ihre Anwendungen wie den Ladezustand der Batterie (D. Didden) verwenden. In der Arbeit von D. Didden wurde mehr als ein Netz verwendet. Auch wurden bei D. Didden und M. Kelker ein hoher Anteil an DEA simuliert. Außerdem wurde ein hoher Anteil an Wärmepumpen (D. Didden) und ein hoher Anteil an EVs (M. Kelker) untersucht. Folglich geben die Simulationen nicht die aktuelle Situation in Niederspannungsnetzen wieder, sondern einen zukünftigen Bedarf an erneuerbaren Energien, Wärmepumpen und EVs.

Je nach betrachtetem Szenario wird ein SARL- oder MARL- Ansatz verwendet. J. Duan verwendete die Anpassung der Parameter von Generatoren als Maßnahmen zur Steuerung der Netzspannung gleichzeitig. Aus diesem Grund wurde ein SARL- Ansatz verwendet. Im Fall von D. Didden und M. Kelker wurden mehrere steuerbare Objekte wie Batterien oder EVs simuliert. Es wurde also mehrere Agenten (MARL) trainiert. Außerdem verwenden alle DQN als Hauptmethode. Darüber hinaus verwendete M. Kelker Double Deep Q-learning (DDQN)

als eine weiterentwickelte Technik von DQN und J. Duan nutzte neben DQN auch Deep Deterministic Policy Gradient (DDPG), um die Leistung zu verbessern.

Als Aktionen nutzte D. Didden einen Aktionsraum zum Laden und Entladen der Batterie, J. Duan konzentrierte sich mehr auf die Anpassung der Generatoreinstellungen und M. Kelker Agenten können die Ladeleistung von Elektrofahrzeugen verändern oder die Batterien laden und entladen.

Als Belohnungen wurden hauptsächlich die netzsicherheitskritischen Faktoren, wie die Einhaltung des Spannungsbandes und die Überlastung von Leitungen und Transformatoren verwendet. Bei Überschreitung der Begrenzungen wurde eine negative Belohnung und bei Einhaltung der Grenzen eine positive oder keine Belohnung vergeben.

Die Ergebnisse der Arbeiten sind alles in allem ähnlich. RL als autonome Steuerung für Niederspannungsnetze funktioniert und die Spannungsbandverletzungen können deutlich reduziert werden. Das wichtigste Ergebnis von J. Duans Arbeit ist, dass in 99,92 % der Fälle die richtige Aktion für den Agenten in der ersten Iteration (DDPG) gefunden werden kann. Auch mit dem normalen DQN-Agenten wurden nur 3-4 Iterationen benötigt. In D. Diddens Ergebnissen können nicht alle trainierten RL das Ziel erreichen. Offline und online trainierte SARL-Agenten mit einer zentralen Steuerung für Batterien reichen nicht aus, um besser als normale Controller zu sein. Online gelernte Agenten haben jedoch besser abgeschnitten als offline gelernte. Bei Betrachtung eines dezentralen Ansatzes erreicht auch M. Kelker gute Ergebnisse. Durch die Regelung konnte eine Reduzierung der Transformatorleistung um 24.4 % und eine Erhöhung der Energie der DEA um 10 % erreicht werden. Darüber hinaus konnte der bidirektionale Leistungsfluss von PV- Anlagen um 10 % reduziert werden. In D. Diddens Arbeit wurde das Spannungsband eingehalten. Durch die in Tabelle I dargestellten zusätzlichen Maßnahmen wurde jedoch die PV- Leistung verringert. Außerdem hat D. Didden die Leistungsstärke zwischen den Rl-Agenten und verschiedenen Reglern untersucht. Die SARL- und normalen MARL-Methoden waren nicht besser als die traditionellen Regler. Aber mit den fortgeschrittenen RL-Techniken, die in der Arbeit beschrieben werden, können RL-Agenten ein besseres Ergebnis erzielen als die meisten der betrachteten Regler [9, 10, 11].

4 Kritik an RL

RL-Modelle lernen auf der Grundlage ihrer Umgebungen. Die in den wissenschaftlichen Arbeiten veröffentlichten Umgebungen sind Simulationen des Stromnetzes. Das bedeutet, dass die vorgestellten Modelle nur auf der Grundlage der Simulation lernen. Je besser die Simulation, desto besser das Modell. Eine in der Praxis getestete Anwendung gibt es noch nicht oder konnte nicht gefunden werden [4, 6].

Wie gut eine Simulation ist, hängt von ihrem Detailgrad und der Qualität der Eingabedaten ab. In der Arbeit von D. Didden und M. Kelker wurden teilweise reale Daten von Verteilnetzbetreibern bereitgestellt. Dies führt zu realistischeren Ergebnissen, bringt aber auch Nachteile mit sich. Einer davon ist die Homogenität der Daten. Es gibt kaum negative Beispiele z.B. Kabelbruch, Messfehler) in den Daten, sodass das Modell in der Regel nur aus positiven Beispielen lernt. Dies kann zu einem fehlgesteuerten Zustand führen, bei dem der Agent keine oder die falsche Aktion durchführt.

Darüber hinaus wird für ein robustes Modell eine große Datenmenge benötigt, was auf der Niederspannungsebene eine Herausforderung darstellen kann. Vor allem in Deutschland dürfen Verteilnetzbetreiber aufgrund der Datenschutzverordnung keine Smart-Meter-Daten

verwenden. Auch D. Didden weist darauf hin, dass sein einfacher Lernprozess Dutzende von Jahren an Daten für das Training verwenden muss, um eine ähnliche, nahezu optimale Strategie für die RL-Agenten zu erreichen [3, 4, 10].

Ein weiteres Problem ist die mangelnde Erklärbarkeit. Da Deep Neural Networks, die ein wesentlicher Bestandteil von DQNs sind, Black-Box-Modelle sind, kann im Falle von Fehlern oder Fehlverhalten keine Erklärung gegeben werden, warum dieser Fehler aufgetreten ist. Außerdem ist es angesichts der Komplexität der Umgebung nicht klar, wie und warum der Agent bestimmte Regeln gelernt hat. Daher ist die Methodik nicht vollständig transparent [3, 6,7].

Aufgrund der zufälligen Aktionen zu Beginn des Lernprozesses können die Agenten in der realen Welt nicht frei interagieren. Zum Beispiel kann das Entladen aller Batterien im Netz im falschen Moment zu einer Überlastung des Netzes führen. Anstelle eines stabilen Netzes kann der RL-Steuerungsalgorithmus die Netzinstabilität erhöhen. Daraufhin muss der RL-Agent in einer sicheren Umgebung lernen, was zu den bereits erwähnten Problemen führen kann [6]. In [12] wird das Problem als „reality gap“ bezeichnet.

5 Fazit und Ausblick

Die vorgestellte Methode bietet viel Potenzial für die Netze der Zukunft. Die Ergebnisse von M. Kelker sehen eine Reduzierung der Transformatorleistung um 24,4 % und eine Steigerung der Energie der DEAs um 10 % vor. Im Fall von J. Duan benötigt der Agent in 99,92 % der Fälle nur eine Iteration, um die richtige Maßnahme zur Spannungserhaltung zu finden. Die Ergebnisse in D. Didden zeigen, dass z. B. die Spannungsband Spannungsbandverletzungen durch DQN deutlich reduziert werden können. Die Autoren kommen zu dem Schluss, dass RL ein großes Potenzial für die Netzsteuerung hat. Allerdings beruhen die bisher vorgestellten Ergebnisse jedoch nur auf Simulationen. Außerdem ist die Steuerung des RL-Systems nicht vollständig transparent. Des Weiteren ist eine große Menge an Daten erforderlich, um robuste Agenten zu trainieren. Auch kann in D. Diddens Arbeit gesehen werden, dass der beste RL Agent die meisten traditionellen Steuerungen übertrifft. Zusammenfassend lässt sich sagen, dass RL die theoretische Möglichkeit bietet, teure Netzausbau zu vermeiden und die Netzkomponenten optimal zu steuern. Ob die Methode in der Praxis funktioniert und die Qualität der Trainingsdaten und der Simulation ausreichend ist, muss noch im Feld validiert werden.

6 Danksagung

Die Forschung, die zu diesen Ergebnissen führte, wurde im Rahmen des Projekts 'KI-Grid' durch das Ministerium für Klimaschutz, Umwelt, Landwirtschaft, Natur- und Verbraucherschutz des Landes Nordrhein-Westfalen und der Europäischen Union (EFRE) Union (EFRE) im Rahmen des EFRE-Programms EFRE.NRW 2014- 2020.

7 Referenzen

[1] Kraftfahrbundesamt. “Neuzulassungen im Jahr 2019”. In: (2019).

[2] Juergen Schlabbach; Frank Fischer. Netzanschluss von EEG-Anlagen. 2nd ed. Rolf Ruediger Cichowski, 2016.

[3] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement Learning: A Survey. 1996. arXiv: cs/9605103 [cs.AI].

[4] R.S. Sutton and A.G. Barto. “Reinforcement Learning: An Introduction”. In: IEEE Transactions on Neural Networks 9.5 (1998), pp. 1054–1054. DOI: 10.1109/TNN.1998.712192.

[5] Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, oy H. Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, Ryan Sepassi, George Tucker, and Henryk Michalewski. “Model-Based Reinforcement Learning for Atari”. In: CoRR abs/1903.00374 (2019). arXiv: 1903.00374. URL: http://arxiv.org/abs/1903.00374.

[6] Vincent Franc ̧Lois-Lavet, Peter Henderson, Riashat Islam, Marc G Bellemare, and Joelle Pineau. “An Introduction to Deep Reinforcement Learning”. In: Foundations and Trends® in Machine Learning 11.3-4 (2018), pp. 219–354. ISSN: 1935-8245. DOI: 10 . 1561 / 2200000071. URL: http://dx.doi.org/10.1561/2200000071.

[7] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. “Human-level control through deep reinforcement learning”. In: Nature 518.7540 (Feb. 2015), pp. 529–533. ISSN: 00280836. URL: http://dx.doi.org/10.1038/nature14236.

[8] Arup Kumar Sadhu and Amit Konar. “Consensus Q-Learning for Multi-agent Cooperative Planning”. In: Multi-Agent Coordination: A Reinforcement Learning Approach. 2021, pp. 167–182. DOI: 10.1002/ 9781119699057.ch3.

[9] Jiajun Duan, Di Shi, Ruisheng Diao, Haifeng Li, Zhiwei Wang, Bei Zhang, Desong Bian, and Zhehan Yi. “Deep-Reinforcement-Learning- Based Autonomous Voltage Control for Power Grid Operations”. In: IEEE Transactions on Power Systems 35.1 (2020), pp. 814–817. DOI: 10.1109/TPWRS.2019.2941134.

[10] Hussain Kazmi, Davy Didden, Nadia Wiese, and Johan Driesen. “Sample efficient reinforcement learning with domain randomization for automated demand response in low-voltage grids”. In: IEEE Journal of Emerging and Selected Topics in Industrial Electronics (2021), pp. 1–1. DOI: 10.1109/JESTIE.2021.3117119.

[11] Michael Kelker, Lars Quakernack, and Jens Haubrock. “Multi agent deep Q-reinforcement learning for autonomous low voltage grid control”. In: 2021 IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe). 2021, pp. 1–6. DOI: 10.1109/ISGTEurope52324.2021. 9639897.

[12] Nick Jakobi, Phil Husbands, and Inman Harvey. “Noise and the Reality Gap: The Use of Simulation in Evolutionary Robotics,”. In: vol. 929. Jan. 1995, pp. 704–72