FKM
Forschung im VDMA

InPulS

Intelligente und selbstlernende Produktionsprozesse

FKM-Vorhaben Nr. 440
Heft 336 | 2020

Abschlussbericht

www.fkm-net.de

InPulS

Vorhaben Nr. 7044000

Intelligente und selbstlernende Produktionsprozesse

Abschlussbericht

Kurzfassung:

Für KMU ist die Integration von Automatisierungslösungen aufgrund der hohen Investitionskosten häufig wirtschaftlich nicht tragbar. Insbesondere in der Automatisierungs- bzw. in der Steuerungstechnik von Maschinen und Anlagen liegen über Jahre gewachsene, starre Systeme vor, deren technologische Fähigkeiten stark auf den jeweiligen Einsatzzweck ausgerichtet sind. Ein aussichtsreicher Lösungsansatz zur Integration neuester Technologien trotz wechselnder Randbedingungen, Prozessstörungen und geringer Losgrößen ist der Einsatz von Verfahren aus dem Forschungsfeld der künstlichen Intelligenz (KI). Besonders vielversprechend im Bereich der Steuerungstechnik ist dabei der Bereich Reinforcement Learning.

Trotz des großen Potentials des Reinforcement Learning ist diese Methodik in der Industrie bisher nicht verbreitet. Dies lässt sich zum einen auf die industriellen Anforderungen hinsichtlich Robustheit, Sicherheit und Dateneffizienz zurückführen. Zum anderen sind bisher wenig konkrete Anwendungsbeispiele bekannt, die als „Leuchttürme“ dienen. Im Rahmen des Projektes InPulS wurden daher zunächst in Kooperation mit Projektpartnern aus der Industrie Anforderungen an neue industriell anwendbare Lernverfahren ermittelt. Unter Berücksichtigung dieser wurden Anwendungsszenarien identifiziert und ein neues Reinforcement Learning Verfahren entwickelt. Das neu entwickelte Verfahren zeichnet sich durch besonders hohe Dateneffizienz, Robustheit und Generalisierungsfähigkeit aus und ist damit als erstes Verfahren den industriellen Anforderungen gewachsen. Die industrielle Anwendbarkeit konnte in der Steuerung eines pneumatischen Schüttgutförderers erstmalig gezeigt werden: Gegenüber einer initial gewählten Steuerungsstrategie wurde eine Effizienzsteigerung um 20% erreicht. Dabei wurde die Steuerungsstrategie mit wenigen Trainingsdaten erlernt und konnte anschließend von einem Fördergut (Trainingsmaterial) auf ein anderes (Testmaterial) übertragen werden.

Damit wurde einer der ersten industriellen Anwendungsfälle von Reinforcement Learning realisiert und so die Möglichkeit der Integration dieser hochkomplexen Verfahren auch für KMU demonstriert. Die in Rahmen von InPulS formulierten Handlungskonzepte erleichtern es KMU, eine geeignete Einführungsstrategie für Reinforcement Learning zu ermitteln und so die Anwendbarkeit in KMU sicherzustellen. So kann das Potential dieser Technologie genutzt und basierend darauf neue Geschäftsmodelle entwickelt werden.

Das Ziel des Forschungsvorhabens ist erreicht worden.

Berichtsumfang:	60 S., 41 Abb., 7 Tab., 44 Lit.
Laufzeit:	01.10.2017 - 30.09.2019
Zuschussgeber:	VDMA/FKM-Eigenmittel
Forschungsstelle(n):	Institut für Unternehmenskybernetik e.V. (IfU) Leiter: Dr. phil. Daniela Janssen

Bearbeiter und Verfasser:	Emma Pabich, M. Sc. Dr. phil. Daniela Janssen Dr. rer. nat. Frank Hees

Vorsitzende(r) projekt-begleitender Ausschuss:	Dipl. Inform. med. Dipl. Ing. (FH) Dieter Herzig (AZO GmbH + Co. KG)
Vorstandsvorsitzender FKM:	Dipl.-Ing. Hartmut Rauen (VDMA)
Weitere Berichte zum Forschungsvorhaben:	-

Danksagung

Dieser Bericht ist das wissenschaftliche Ergebnis einer Forschungsaufgabe, die von dem VDMA Forum Industrie 4.0 und von dem Forschungskuratorium Maschinenbau (FKM) e.V. gestellt und am Institut für Unternehmenskybernetik e. V. (IfU) unter der Leitung von Dr. Daniela Janssen bearbeitet wurde. Der VDMA und das FKM dankt der Professorin Dr. rer. nat. Sabina Jeschke und den wissenschaftlichen Bearbeitern Philipp Ennen, Emma Pabich und Yun Zheng (IfU) für die Durchführung des Vorhabens. Für die Leitung des Projekts seitens des VDMA-Forum Industrie 4.0 danken wir Judith Binzer. Das Vorhaben wurde von einem Arbeitskreis des VDMA unter der Leitung von Dipl. Inform. med. Dipl. Ing. (FH) Dieter Herzig (AZO GmbH + Co. KG) begleitet. Diesem projektbegleitenden Ausschuss gebührt unser Dank für die große Unterstützung. Insbesondere danken wir:

AZO GmbH + Co. KG

Festo AG & Co. KG

FIBRO GmbH

Hans Weber Maschinenfabrik GmbH

Karl Mayer Textilmaschinenfabrik GmbH

Geschäftsbereich Komponentenfertigung

Lenze SE

MAHLE Behr GmbH & Co. KG

Oskar Frech GmbH + Co. KG

Schaeffler Technologies AG & Co. KG

SchuF-Armaturen und Apparatebau GmbH

SMC Deutschland

TE Connectivity Germany GmbH a

TE Connectivity Ltd. Company

THEEGARTEN-PACTEC GmbH & Co.KG

Voith GmbH & Co. KGaA

Volkswagen AG

Weidmüller Interface GmbH & Co. KG

ZIMMER GmbH

Die Arbeit wurde durch den VDMA und das Forschungskuratorium Maschinenbau (FKM) e.V. aus Eigenmitteln (FKM Fördernummer: 7044000) finanziell gefördert.

Inhaltsverzeichnis

1 Executive Summary

In Germany, the sector of automation and robotics comprises approximately 500 companies. In average, they have 65 employees and have a volume of sales of 11 billion Euro per year [1]. This sector therefore heavily contributes to the competitiveness of the German industry.

Nevertheless, these small or middle-sized companies face difficulties when trying to incorporate state-of-the-art solutions for automation. This is mainly due to high investment costs, small batch sizes and alternating constraints [2]. Especially within the field of automation and control of machines and plants, existing solutions are often systems which were grown over years and hence lack flexibility. Their technology is constrained to a narrow field of application. This condition can be explained with the strict requirements in industrial production regarding robustness and safety. However, it leads to a reduced adaptability of production systems.

Nowadays, automation systems are equipped with a growing number of sensors, which are increasingly interconnected. The data they deliver assists in assessing the operating state of a system. At the same time, the obtained data points have the potential to enable new concepts and improvements. To exploit these new possibilities, intelligent and self-learning systems are gaining more and more significance.

In order to detect, understand and use these potentials, small and middle-sized companies within Germany require concepts of action. They have to be enabled to assess fields of implementation, develop new business models and implement the underlying technologies. Therefore, the project InPulS focused on developing and analyzing such concepts for modeling and implementation of self-learning production processes for small and middle-sized companies. These concepts provide guidance for companies from within the sector of automation and robotics, but also from the production industry in order to

- assess the potential and possible benefits of intelligent production processes within their industrial context and enable them to
- enhance existing processes to self-learning control circuits.

The main research goal of this project was the formulation of context-based learning methods in a self-learning production process as well as the evaluation of the application of these methods on concrete use cases in SMEs. The extension and adaption of these procedures thus enables the integration of intelligent learning methods in the production process which meet the requirements of SMEs.

Industrial Reinforcement Learning

Machine learning is a subfield of artificial intelligence comprising of a variety of different concepts and methods. The common factor between the methods is the reliance on a pool of data to train a model for a desired task. In machine learning three classes are distinguished: supervised learning, unsupervised learning and reinforcement learning. These methods are suitable for different tasks, depending on the use case and its complexity. In an industrial context, machine learning methods show great potential in the area of process monitoring, optimization and control. While in process monitoring and optimization often supervised or unsupervised learning methods are used, in the context of control the use of reinforcement learning is particularly promising as they are based upon direct interaction with the environment.

Reinforcement learning is based upon trial and error. For industrial applications this gives limitations and necessary conditions for the possible applications. Thus, in this project a new kind of reinforcement learning has been developed, industrial reinforcement learning. Industrial reinforcement learning is characterised by special requirements for the algorithms regarding robustness, training safety and data efficiency. While current reinforcement learning applications often focus on the domain of robotics or computer games, where reliable simulations exist, industrial applications often require training on the real system. This requires additional care when testing unknown parameter settings in order to ensure safety of employees and machines at all times. For this purpose, the real system should have a high fault tolerance and

occurring faults should be easy to detect and eliminate. The fault tolerance of the system also plays an important role in the robustness against uncertainties. Uncertainties occur in various form in a production process, e.g. environmental conditions such as the temperature or variances in natural products. The reinforcement learning approach needs to be chosen and designed such that it reacts robustly against uncertainties in the environment. These special requirements of industrial reinforcement learning show that the development of adequate algorithms and their implementation in practice is highly complex.

Project Outline

For this project, the development process has been divided into five phases: analysis, conceptualization, implementation, validation and documentation.

In the analysation and formalization phase, the requirements of SMEs for industrially suitable models applicable for self-learning control of production processes were collected. Within workshops with participants of the PA the research objective was compared and evaluated to industrial needs. In conformity to the necessary requirements possible application scenarios were discussed. The industrial needs and the possible application scenarios then enabled the development of requirements for the intelligent systems allowing the integration of self-learning systems into the production environment.

A new reinforcement learning algorithm suitable for these specific environments was developed. For this, different learning methods were compared and their suitability was evaluated. In order to ensure the best possible adaptation to the needs of SMEs, three use cases were developed. One scenario comprised a position- and force-controlled assembly process with continuous state and action space. As a representative part of an assembly process a joining-task based on a peg-in-hole-task was considered. A second task, consisted of the self-learning control of a pneumatic bulk material conveyor. This application enabled the direct testing in industry. The last scenario, a use case with a discrete state and action space, considered the planning of a production process.

Based on these scenarios, system architectures, target functions and policies were developed, which were implemented for the various use cases. By using reinforcement learning algorithms in these practical scenarios, the feasibility of a robust and data-efficient reinforcement learning could be demonstrated. The implementation of these pilot projects shows the great potential of reinforcement learning in particular for the adaptive control of plants and processes. The implemented control is able to react to disturbances and changed environmental conditions.

Integration Strategy for SMEs

Within the project a guideline for SMEs was developed, facilitating the transfer of the research results into industry. The guideline shows an implementation strategy for the application of reinforcement learning in industrial automation. The reader is enabled to recognize the potentials as well as the necessary framework conditions for the application of reinforcement learning for the industrial application. With the help of a set of questions and an associated toolbox, the guide provides a tool to facilitate the introduction of reinforcement learning. The guide is addressed to companies, wanting to make their production system more efficient with the help of machine learning and looking for orientation on risks and potentials as well as an implementation strategy. The guide explains the differences between conventional control and self-learning control using reinforcement learning and introduces the necessary terms and principles in the field of reinforcement learning. Afterwards, the questions and the appropriate toolbox for answering these questions are provided in order to facilitate the selection of a suitable pilot project. This is followed by a sketch of the current algorithmic approaches in the field of reinforcement learning, which should serve as a starting point for more in-depth research. The guide gives insight into what kind of personnel as well as hardware resources are necessary for the successful integration of reinforcement learning in production processes. Here three kinds of expertise are identified:

- a reinforcement learning expert for the development of industrially suitable methods
- a system expert who has knowledge of the underlying process and

- a software developer for the development of fast and efficient machine learning code.

These three different skill sets are vital for the success of industrial reinforcement learning. Finally, the integration strategy is illustrated by the two industrial use cases from the InPulS project, the autonomous assembly process and a self-learning control system for a pneumatic bulk material conveyor.

Applicability in Industry

In recent years, industry has placed great expectations on Big Data. Today there is a trend away from Big Data towards Smart Data. Contrary to the widespread assessment that large amounts of data are available in the industry, there are still many scenarios, for example in the area of special mechanical engineering, in which comprehensive data acquisition is not possible. Especially data-efficient processes are required for these applications. Reinforcement learning fits into this trend, as only selected data tailored to the specific application must be collected. The potential of reinforcement learning as part of machine learning is only slowly being discovered. With the help of reinforcement learning, an efficient adaptive control can be learned in scenarios where modelling with conventional methods would be too complex. This opens up the potential for an efficiency gain with regard to conventional control systems the reduces the need for time-consuming manual adjustment of the plant parameters.

The application on the bulk material conveyor has confirmed the enormous potential of industrial reinforcement learning. A particularly robust variant of a Guided Policy Search algorithm has been developed for use on bulk material conveyors and with this method an efficiency gain of 20% compared to initial control parameters was achieved. Even more striking is the transfer of a control policy trained with one material (plastic granulate) to a new material with different characteristics (mustard flour). The control policy was able to control the conveying process of the new material and achieved the same accuracy as on the training material. However, the question remains how such an algorithm can be transferred to other scenarios. Further application-oriented fundamental research is necessary to develop new application areas and corresponding algorithms for reinforcement learning.

2 Wissenschaftlich-technische und -wirtschaftliche Problemstellung

2.1 Ausgangssituation und Anlass für den Forschungsantrag

In der deutschen Robotik- und Automatisierungsbranche beschäftigen sich ca. 500 Unternehmen mit der Herstellung und Integration von Automatisierungslösungen. Diese Unternehmen haben im Durchschnitt ca. 65 Mitarbeiter und erzielen einen jährlichen Gesamtumsatz von rund 11 Mrd. Euro [3]. Diese Branche trägt maßgeblich dazu bei, dass das produzierende Gewerbe auch zukünftig in Deutschland zu wettbewerbsfähigen Bedingungen produzieren kann.

Die Integration von Automatisierungslösungen ist aufgrund der hohen Investitionskosten, bei geringen Losgrößen oder wechselnden Randbedingungen, für KMU jedoch häufig nicht wirtschaftlich tragbar [4]. Insbesondere in der Automatisierungs- bzw. in der Steuerungstechnik von Maschinen und Anlagen liegen über die Jahre gewachsene, starre Systeme vor, deren technologische Fähigkeiten stark auf den jeweiligen Einsatzzweck ausgerichtet sind. Grundsätzlich ist dies auf die strengen Anforderungen in der industriellen Produktion bezüglich Robustheit (deterministisches Verhalten in jedem Betriebszustand) und Sicherheit (definierte Einschränkungen während des Betriebs) zurückzuführen. Dies führt in der Regel zu einer stark eingeschränkten Anpassungsfähigkeit der Produktion. Ein aussichtsreicher Lösungsansatz dazu ist es, die Maschinen und Anlagen mit einer eigenen Intelligenz und Lernfähigkeit auszustatten.

Künstliche Intelligenz (KI) beschäftigt sich mit der Automatisierung intelligenten Verhaltens [5]. In den vergangenen Jahren wurden mit Verfahren der KI bereits bemerkenswerte Fortschritte in der Objekt- und Spracherkennung [6], [7] erzielt. Vieles davon basiert auf einer Variante des Deep Learning [8], worin große, mehrschichtige neuronale Netze antrainiert werden. Ein weiteres Einsatzfeld ist die Regelung von Systemen, üblicherweise durch eine Kombination verschiedener KI-Verfahren. Hervorzuheben sind selbstgelernte Hubschrauberflüge [5], autonome fahrerlose Transportsysteme [9], selbstoptimierende Montageprozesse [10], sowie treffsichere Vorhersagen von Kundenverhalten [11]. Obwohl sich diese neu entstandenen Technologien an vielen Stellen durch unkonventionelle und schwer nachvollziehbare Lösungen auszeichnen, führen sie vielerorts zu effizientem und robustem Systemverhalten. Dieser Erfolg stellt auch für die Produktion eine große Chance dar. Der Einsatz von Verfahren aus dem Forschungsfeld der künstlichen Intelligenz bergen Potentiale zur Entfaltung neuer Funktionalitäten in Produktionssystemen und stellen somit einen Beitrag zur Entlastung der Benutzer dar. Somit besteht die Möglichkeit, das Engineering, die Wartung sowie das Lebenszyklusmanagement zu verbessern und Zuverlässigkeit, Sicherheit und Verfügbarkeit zu erhöhen. Darüber hinaus ermöglicht der Einsatz von Verfahren aus dem Forschungsfeld der künstlichen Intelligenz, Ressourcen wie Energie und Material effizienter einzusetzen und so äußerst flexible und einfach wandelbare Produktionsprozesse zu ermöglichen. Diese Verfahren zeichnen sich durch einen breiten Einsatzbereich aus, wobei der konkrete Nutzen, insbesondere für KMU, in der Regel meist schwer zu quantifizieren ist. Um dies zu konkretisieren, ist ein genaues Verständnis über selbstlernende Systeme notwendig. Es bedarf der Entwicklung von Handlungskonzepte, die es KMU ermöglichen, Einsatzfelder und Potentiale von selbstlernenden Produktionsprozessen abzuschätzen und zu implementieren. Im Rahmen von InPuls werden daher Handlungskonzepte zur Modellierung und Implementierung von selbstlernenden Produktionsprozessen für KMU entwickelt und untersucht. Ziel ist es, erfolgreiche Handlungskonzepte für den praktischen Einsatz kontextbasierter Lernverfahren in Prozessregelkreisen zu erarbeiten, zu demonstrieren und zu dokumentieren. Diese Handlungskonzepte ermöglichen es KMUs aus der Robotik- und Automatisierungsbranche, als auch der produzierenden Industrie die

- Potentiale und den Nutzen selbstlernender Produktionsprozesse in ihrem speziellen Kontext abschätzbar zu machen und versetzen sie in die Lage,
- vorhandene Prozesse zu selbstlernenden Prozessregelkreisen zu erweitern.

Betrachtungsgegenstand stellen dabei intelligente Produktionsprozesse dar (vgl. ***Abbildung 1***) die über ein Verständnis von Zielgrößen verfügen, jedoch über kein Systemmodell, welches die physikalisch-technische Dynamik der Produktionseinheit beschreibt.

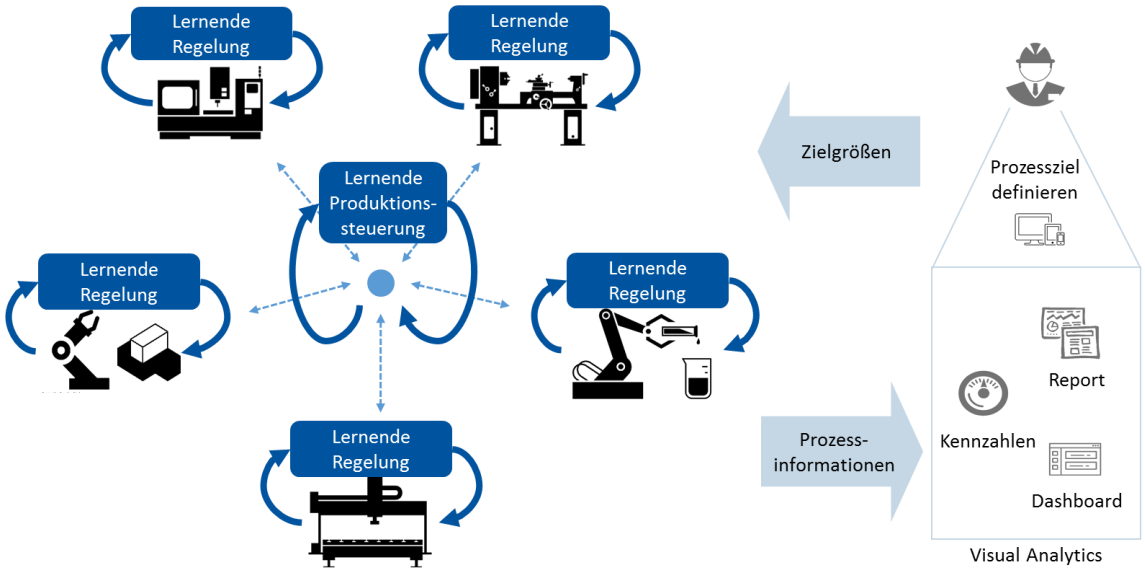

Abbildung 1: Zielbild von InPulS für selbstlernende Produktionsprozesse

Während der Produktion besteht die Herausforderung, dynamische Effekte sowie Toleranzen der zugeführten Rohstoffe und Bauteile auszugleichen und auf wechselnde Produktvarianten reagieren zu müssen. Dies erfordert entweder intensive Einblicke in die Systemdynamik oder ein zeit- und kostenintensiver Trial-and-Error beim Einprogrammieren der individuellen Steuerungsstrategien für verschiedenen Varianten, häufig durchgeführt durch Experten. Durch den Einsatz von Verfahren der künstlichen Intelligenz werden die Prozesse in die Lage versetzt, robuste Strategien für wechselnde Varianten zu erlernen, ohne dass die Systemdynamik in ihrer genauen Beschaffenheit bekannt sein muss.

Im Rahmen des Projektes InPulS wurden dazu Einsatzbereiche vielversprechender KI-Verfahren in Bezug auf industrierelevante Problemstellungen in der Prozessregelung identifiziert und deren Potential in einem Demonstrator für diskrete und kontinuierliche Prozesse dargestellt. Anschließend wurden die Erkenntnisse in einen web- und printbasierten Handlungsleitfaden überführt und in industriellen Anwendungsbeispielen innerhalb des projektbegleitenden Ausschusses (PA) validiert. In den industriellen Anwendungsbeispielen werden verschiedene Varianten der lernenden Prozessregelung in einem Unternehmen umgesetzt. In der Prozessindustrie wurde das System zudem exemplarisch an einem pneumatischen Schüttgutförderer, der auf Jahrzehnte ausgelegt ist und dabei Güter mit unterschiedlichsten Eigenschaften im jeweils effizientesten Betriebspunkt fördert und somit ein kontinuierliches Adaptieren der Steuerungsstrategie erfordert, validiert. Die durchgeführten Anwendungsfälle sind von ihrer Beschaffenheit (kontinuierliche Prozessindustrie (Szenario A), diskontinuierliche Montage (Szenario B)) bewusst unterschiedlich gewählt. Sie zeichnen sich dadurch aus, dass es sich bei ihnen um komplexe nichtlineare Systeme mit nichtdeterministischen Randbedingungen handelt, die nicht oder nur begrenzt in Beschreibungsmodellen abgebildet überführbar sind.

In diesem Kontext wird der breite Einsatzbereich der Verfahren der künstlichen Intelligenz und ihre generische Eignung für eine Vielzahl von Problemstellungen demonstriert. Zur ganzheitlichen Betrachtung der Lernverfahren für ein breites Anwendungsspektrum werden diese nach der Beschaffenheit der Aktions- und Zustandsräume erarbeitet und untersucht (vgl. **Tabelle *1***). Der Forschungsdemonstrator basiert auf einem bestehenden System, welches um mehrere simulierte und einer physischen Montagezelle erweitert wird. Darin werden in einem kontinuierlichen, kraftgeführten Prozess, Komponenten einer Getriebebox montiert. Über die simulierten Montagezellen werden vorgelagerte und nachgelagerte Prozessschritte und deren Abhängigkeiten untereinander modelliert. Der Forschungsdemonstrator stellt Lernen auf der Ebene

der diskreten Ablaufplanung und -steuerung, sowie auf der Ebene der kontinuierlichen Fügeprozesse dar.

	Diskrete Aktionen	**Kontinuierliche Aktionen**
Diskrete Zustände	Produktionsablauf mit simulierten Montagezellen (Szenario A)	*Kein Untersuchungsgegenstand*
Kontinuierliche Zustände	*Kein Untersuchungsgegenstand*	• Positions- und kraftgeregelte Montageprozesse an physischer Hardware (Szenario A) • pneumatischer Schüttgutförderer (Szenario B)

Tabelle 1: Anwendungsfälle im Industrie- und Forschungsdemonstrator

Innovativer Beitrag

Das Vorhaben InPuls versetzt KMU in die Lage, für existierende Maschinen und Anlagen Anforderungen, Methoden und Implementierungsstrategien abzuleiten, um diese Systeme erfolgreich mit einer eigenen Lernfähigkeit auszustatten. Somit werden die Systeme befähigt, den sie umgebenden Systemkontext zu verstehen und sich flexibel an geänderte Randbedingungen anzupassen. Der innovative Beitrag des Projektes liegt insbesondere in der Befähigung von Systemen zur eigenständigen Optimierung ihres Verhaltens ohne, dass dazu eine zeit- und kostenintensive Modellierung aller Teilkomponenten notwendig ist. Die betrachteten KI-Verfahren lernen auf den Umgebungskontext zu reagieren, so dass sich Systeme um einen Betriebspunkt herum an neue, nicht modellierte Randbedingungen anpassen können.

Nutzung der erzielten Forschungsergebnisse

Die Entwicklung hin zu selbstlernenden Produkten und Systemen ist fester Bestandteil der deutschen Hightech-Strategie Industrie 4.0 [12]. In diesem Kontext proklamierte Komponenten existieren zum Teil bereits heute und werden in Zukunft als cyber-physische Produktionssysteme (CPPS) die Produktion prägen. Während große Unternehmen intelligente und selbstlernende Komponenten als Teil vollständiger Produktionssysteme mehr und mehr einbinden, setzen KMU derartige Komponenten bisher nur vereinzelt ein, wobei die Potentiale zum aktuellen Zeitpunkt noch nicht voll ausgeschöpft sind. Die verteilt und heterogen agierenden Systeme werden meist nicht im Verbund eingesetzt, da es an einer hierzu notwendigen Informations- und Kommunikationstechnologie (IKT) mangelt. Die Ergebnisse von InPuls zielen darauf ab, KMU zu befähigen, die richtigen Komponenten für ihre Produktionsbedürfnisse auszuwählen und gewinnbringend einzusetzen. Die im Rahmen des Projekts umgesetzten Anwendungsfälle dienen als Vorbild verschiedener Produktionsarten (Prozess- / Fertigungsindustrie).

Beitrag zur Steigerung der Wettbewerbsfähigkeit

Eine Steigerung der Wettbewerbsfähigkeit von KMU setzt kostengünstigere Produktionsverfahren oder neue Produkte voraus, die auf einen entsprechenden Bedarf treffen oder mit denen Unternehmen befähigt werden, neue Marktpotentiale zu erschließen. Das Projekt InPuls verfolgt eine weitreichende Strategie, in der Prozesse durch Lernverfahren neue Funktionalitäten entfalten und den Anwender entlasten. Es werden bislang nicht erreichbare Potentiale für KMU erschlossen, indem Produktionssysteme eigenständig und transparent optimale Betriebspunkte finden und folglich flexibel auf wechselnde Produktionsbedingungen reagieren können. Dadurch werden Ressourcen effizienter eingesetzt sowie die Zuverlässigkeit und Verfügbarkeit der Systeme erhöht. Durch die lernenden Komponenten werden eine Verkürzung und Stabilisierung bei der Programmierung von automatisierten Prozessen erreicht, sodass Produktwechsel und -änderungen schneller realisiert werden und Produkte somit schneller auf den Markt verfügbar sind. Weiterhin ermöglichen die Projektergebnisse, wandelbare automatisierte Produktionsprozesse abzuleiten, so dass KMU ihre Produktionssysteme flexibel für ein

breites Produktspektrum einsetzen können. KMU sind so in der Lage, adaptive Prozesse stärker zu automatisieren und die Produktionskosten zu senken.

2.2 Stand der Forschung

Intelligente lernfähige Systeme evaluieren ihre Performance anhand einer Benchmark mit der Idealperformance und leiten daraus Strategien zur Verbesserung des eigenen Systemverhaltens ab. Zusätzlich werden Verfahren des maschinellen Lernens eingesetzt, die es intelligenten Systemen auch in unerwarteten Situationen ermöglichen, Aufgaben zielgerichtet zu erfüllen [13]. Um die gewünschte Flexibilität in komplexen Systemen zu ermöglichen, werden im Forschungsumfeld häufig verteilte agentenbasierte Steuerungssysteme eingesetzt, bei denen jeder Agent über eine eigene Intelligenz verfügt.

Die Anwendung von agentenbasierten Ansätzen zur Steuerung von Fertigungsabläufen und des Enterprise-Resource Planning (ERP) wurde bereits intensiv untersucht [14]–[16]. Auf Feldebene ist die agentenbasierte Steuerung hingegen ein vergleichsweise neues Forschungsfeld [17], [18]. Erste Ansätze wurden jedoch bereits validiert [17]. Geeignete Ansätze für intelligentes Lernverhalten in Produktionssystemen sind im Bereich des maschinellen Lernens zu finden. Maschinelles Lernen ist ein Teilaspekt des Forschungsfeldes künstlicher Intelligenz [KI], der viel Aufmerksamkeit auf sich gezogen hat und immer noch zieht. Daraus ist eine große Menge an Lern-Methoden entstanden, die in Supervised, Unsupervised und Reinforcement Learning unterteilt werden [16], [19], [20].

Reinforcement Learning (kurz: RL) gilt als einer der generischsten Ansätze für lernfähige Steuerungssysteme, mit dem bspw. bereits komplexe motorische Montageprozesse erlernt wurden [16], [21]. Im Reinforcement Learning exploriert ein Agent sein optimales Verhalten basierend auf Trial-and-Error Interaktionen mit seiner Umgebung. Nach jedem Versuch evaluiert ein numerisches Bewertungssystem die Performance des Agenten und quantifiziert dies mit einem Belohnungswert. Das Ziel von Reinforcement Learning ist es, auf Grundlage einer beobachteten Historie von Zustands-Aktions-Belohnungs-Sequenzen eine Steuerungsstrategie (Policy) zu lernen, die für den Agenten in jedem Zustand eine zielführende Aktion ableitet [16]. Steuerungsstrategien werden dabei ganz unterschiedlich abgebildet: als neuronales Netz, als modellprädiktiver Regler, als dynamisches System oder als Liste. In der Praxis etablierte Lernverfahren für Steuerungsstrategien sind unter anderem: Deep Reinforcement Learning [22], Relative Entropy Policy Search [23] und Guided Policy Search [24]. Das Verfahren Deep Reinforcement Learning erlernt eine Bewertung des Nutzens von Systemzuständen und leitet daraus eine optimale Strategie ab. In den übrigen Verfahren werden unmittelbar eine optimale Steuerungsstrategie erlernt. Die Verfahren fanden bereits erfolgreich Anwendung in einer Vielzahl von Szenarien, wie beispielsweise in GO-Spielen (Deep Reinforcement Learning) [25], dem Erlernen von Tischtennisbewegungen mit einem Roboter (Relative Entropy Policy Search) [26] und der Steuerung von komplexen, robusten Handhabungsaufgaben mit einem Roboter [27].

Diese traditionellen Reinforcement Learning Methoden sind in realen Produktionssystemen bisweilen nicht ohne weiteres anwendbar. Die Anforderungen an sichere und robuste Prozesse sowie einen effizienten Ressourceneinsatz verhindern den Einsatz aufwendiger Reinforcement Learning Verfahren. Es bedarf einer anwendungsorientierten Selektion und Kombination der Verfahren, die zielgerichtet geführt ein effizientes Erlernen neuer Aufgaben ermöglicht. Hierbei ist eine Kombination aus RL mit modellgetriebenen Ansätzen (z.B. modellprädiktive Regler) denkbar [28].
Grundsätzlich wurde im Projekt InPuls die Anwendbarkeit unterschiedlicher Lernverfahren für das Erlernen von Steuerungsstrategien in unterschiedlichen Produktionsszenarien untersucht.

2.3 Institut für Unternehmenskybernetik e. V. (IfU)

Das Institut für Unternehmenskybernetik e.V. (IfU) ist ein gemeinnütziges, unabhängiges, branchenübergreifendes und interdisziplinäres Forschungs- und Entwicklungsinstitut an der RWTH Aachen auf dem Gebiet der Technischen Kybernetik, der Wirtschafts- und Sozialkybernetik sowie der Mobilen Robotik.

Das IfU erforscht u.a. zukünftige Produktionssysteme. Dazu adressiert es die Integration intelligenter Planungs- und Steuerungssysteme für die Produktion unter Verwendung von Verfahren aus den Forschungsgebieten der künstlichen Intelligenz, wissensbasierter Systeme und Kognition [9], [10], [21], [29], [30]. Diese werden adaptiert und angepasst für das Anwendungsfeld der Produktionstechnik. Von besonderem Interesse sind die Anwendungsfelder der automatisierten Bewegungsplanung für Montageaufgaben und der Produktionsablaufplanung [21]. In diesem Kontext betrachtet das IfU zentralistische Planungsansätze, sowie dezentrale, agentenbasierte Systeme [27, 7, 28]. Diese zeichnen sich durch Lernfähigkeit aus, wodurch Systeme auf wechselnde und unbekannte Prozessrandbedingungen reagieren [21], bspw. die Windschutzscheibenmontage an bewegten LKW-Kabinen bei unbekannten Bauteilschwingungen (AiF-Projekt Fasim_XL, IGF-Nr. 18425 N). Ein weiterer wissenschaftlicher Fokus liegt in der Anwendung von Teams autonomer, mobiler Roboter in der Intra-Logistik [9], [30]. Neben der akademischen Forschung nimmt das Institut am internationalen Wettbewerb RoboCup Logistic League teil, die als Testumgebung für mobile Roboter in CPPS dient. Das Team konnte sowohl die German Open als auch die Weltmeisterschaften 2014, 2015, 2016 und 2017 für sich entscheiden. Durch zahlreiche Forschungsprojekte verfügt das IfU zudem über ein fundiertes Fachwissen in den Gebieten der Wirtschaftlichkeitsbetrachtung und IT-Anwendungen sowie über Erfahrungen bezüglich spezifischer Anforderungen von KMU. Zur Erfassung von langfristigen Effekten und Wirkungen von Investitionen wird am IfU das entwickelte und etablierte Verfahren der erweiterten Wirtschaftlichkeit (NOWS-Verfahren) eingesetzt. Im Rahmen der Durchführung der erweiterten Wirtschaftlichkeitsbewertung verfügt das IfU über langjährige Erfahrung [31]–[33].

3 Forschungsziel und Lösungsweg

3.1 Forschungsziel

Im Rahmen von InPuls wurden Konzepte zur Modellierung und zur Implementierung von selbstlernenden Produktionsprozessen für KMU entwickelt. Ziel war es, Konzepte für den praktischen Einsatz kontextbasierter Lernverfahren in Prozessregelkreisen zu erarbeiten, zu demonstrieren und zu dokumentieren. KMU wurden auf die Ausrichtung und Erweiterung ihrer Prozesse auf selbstlernende Systeme vorbereitet. Somit können bislang nicht erreichbare Potentiale aus Bereichen wie Engineering, Zuverlässigkeit und Verfügbarkeit erschlossen werden. Darüber hinaus bieten selbstlernende Prozesse die Möglichkeit, Ressourcen effizienter einzusetzen und so äußerst flexible und einfach wandelbare Produktionsprozesse zu gestalten. Um diese Ziele zu erreichen, ist ein exaktes Verständnis selbstlernender Methoden erforderlich. Die erforschten Erkenntnisse wurden in einem Demonstrator an der Forschungsstelle sowie einem Handlungsleitfaden zusammengeführt und in unterschiedlichen industriellen Anwendungen validiert.

Beschreibung der Arbeitshypothese

Das wesentliche Forschungsziel ist die Aufarbeitung von kontextbasierten Lernverfahren in einem selbstlernenden Produktionsprozess sowie die Bewertung dieser Verfahren für konkrete Einsatzmöglichkeiten in KMU. Die Erweiterung und Anpassung der Modelle und Verfahren ermöglicht somit eine Integration des Produktionskontextes in intelligente Prozesse, die den Anforderungen von KMU genügen. Intelligente Prozesse umfassen Maschinen, Anlagen oder Teilkomponenten und verfügen über spezielle Fähigkeiten mit denen sie Zustände von Objekten in ihrer Umgebung (physische Welt) wahrnehmen und verändern (vgl. ***Abbildung 2***). So werden in einem Produktionsprozess beispielsweise Baugruppen montiert, einzelne Bauteile gefertigt oder Fluide transportiert. Dazu wird eine Regelungsstrategie verwendet, die ausgehend von einem wahrgenommenen Zustand eine Aktion für eine spezifische Aufgabe ableitet. Durch Adaption der Regelungsparameter ist der Prozess befähigt, selbstlernend auf wechselnde Randbedingungen und Aufgaben zu reagieren. Durch Lernverfahren (bspw. Reinforcement Learning) werden die Parameter dabei eigenständig hinsichtlich einer Zielfunktion optimiert.

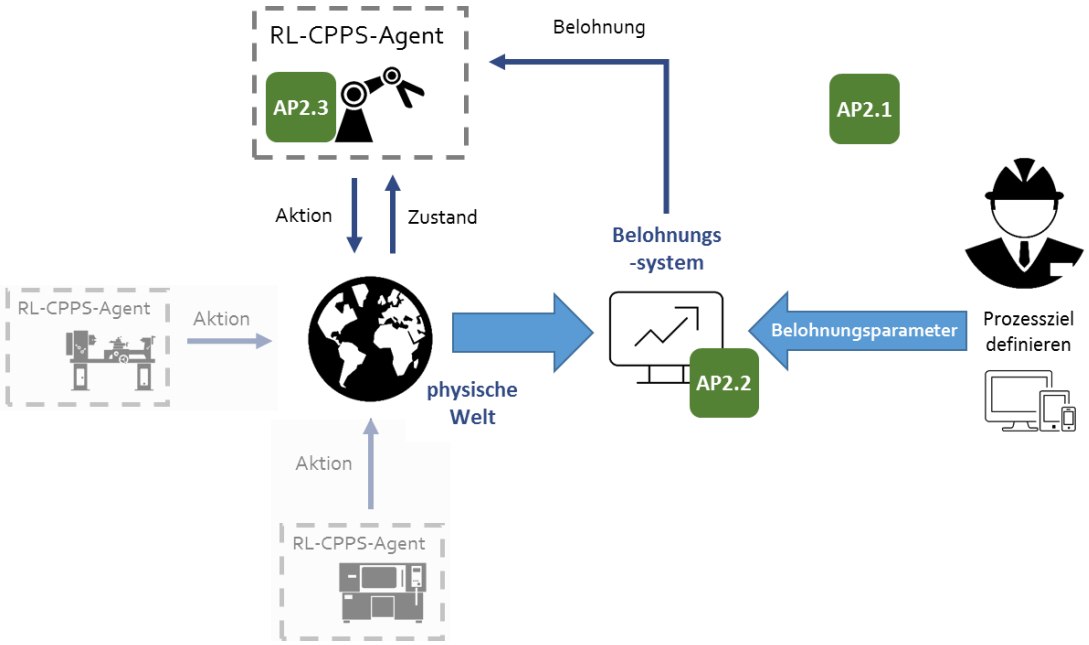

Abbildung 2: Erweitertes Lernmodell am Beispiel von Reinforcement Learning

3.2 Lösungsweg

Das Vorgehen zur Erreichung der Zielsetzung gliedert sich im Folgenden in die fünf Arbeitsschritte Analyse (AP1), Konzeption (AP2), Realisierung (AP3), Validierung (AP4) und Dokumentation (AP5) (vgl. **Abbildung 3**). Innerhalb der Analyse und Konzeption werden Produktionsprozesse betrachtet und individuelle Modelle entwickelt. Als Funktionsmuster wird im Rahmen von InPuls ein Forschungsdemonstrator aufgebaut, der anhand eines konkreten Anwendungsfalls die Applikation der entwickelten Modelle aufzeigt sowie ein webbasierter Leitfaden zur Integration intelligenter, lernfähiger Systeme in bestehende Produktionsumgebungen erarbeitet. Die Firmen aus dem PA umfassen produzierende Unternehmen, Endanwender und Systemintegratoren, die in den einzelnen Arbeitsphasen integriert werden. Im Vordergrund stehen KMU, die anhand des Leitfadens eine dezidierte Vorgehensweise für die Integration intelligenter Prozesse erlangen. Parallel zum Forschungsdemonstrator zeigt eine Evaluation innerhalb von industriellen Anwendungsfällen bei Unternehmen des PA die Anwendbarkeit in bestehenden Produktionssystemen auf.

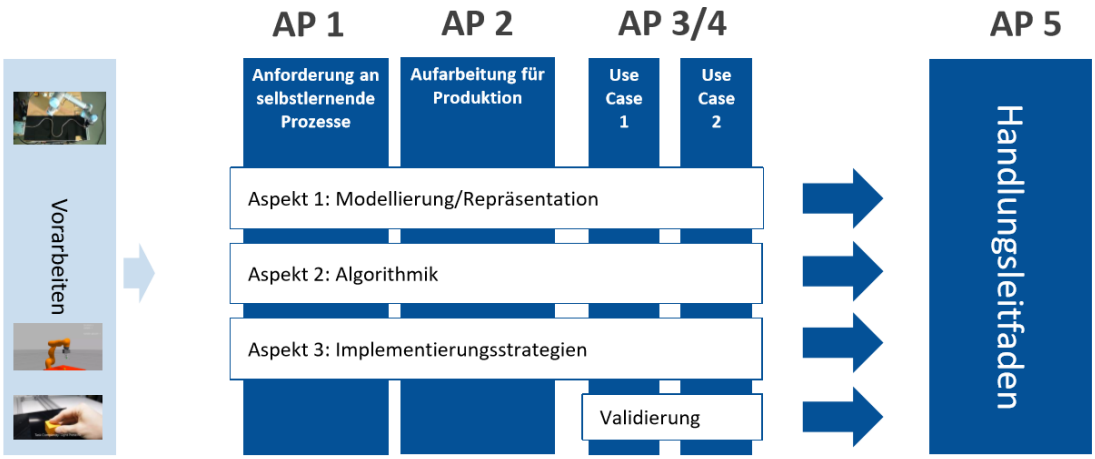

Abbildung 3: Arbeitsplan InPulS

4 Forschungsergebnisse

4.1 Arbeitspaket 1: Analyse

In der Analyse- und Definitionsphase wurden zunächst die Anforderungen der Industrieunternehmen an industrietaugliche Modelle für lernfähige Regelungen für Produktionsprozesse erhoben. Im Rahmen von Workshops und Befragungen von Akteuren aus dem PA und weiterer Industrieunternehmen erfolgte ein Abgleich des Forschungsgegenstandes mit den industriellen Bedürfnissen. Weiterhin wurden Anwendungsszenarien gesammelt, die den Einsatz von lernfähigen Systemen ermöglichen. Damit wurden potentielle Antworten gefunden, inwiefern sich eine Teil- oder Vollautonomie in Produktionsumgebungen erreichen lässt. Auf Grundlage dessen wurden die Anforderungen an die Gestaltung der Aspekte Intelligenz und Lernfähigkeit zusammengetragen, sodass diese sich in die Produktionsumgebung integrieren lassen.

4.1.1 Arbeitspaket 1.1: Anforderungen an selbstlernende Produktionsprozesse

4.1.1.1 Ziel des Arbeitspakets

Ziel des AP 1.1 war die Befragung der Akteure aus dem PA, um die Erwartungen an selbstlernende Produktionssysteme von Seiten der Industrie zu erarbeiten. Hier sollte insbesondere der erwartete Mehrwert eines solchen Systems untersucht werden. Ein weiteres Ziel war die Analyse der Anforderungen an selbstlernende Produktionsprozesse. Im gemeinsamen Dialog sollten mithilfe der Anforderungen mögliche Anwendungsszenarien für intelligente, selbstlernende Produktionssysteme entwickelt werden. Auf Basis der Anwendungsszenarien sollten dann erste Anforderungen an die Gestaltung der Systeme aufgestellt und mit dem PA diskutiert werden.

4.1.1.2 Durchgeführte Arbeiten

In diesem Arbeitspaket wurden Anforderungen an selbstlernende Produktionsprozesse erhoben. Dazu wurde mit dem projektbegleitenden Industriearbeitskreis ein Workshop zur Identifizierung von Anwendungsszenarien durchgeführt und daraus Anforderungen an selbstlernende Produktionsprozesse abgeleitet. Ausgehend vom pneumatischen Schüttgutförderer und einer Gießmaschine wurden die Anforderungen in funktionale (FA) und nichtfunktionale (NFA) Anforderungen gegliedert. Funktionale Anforderungen betrachten die direkte Funktionserfüllung und nichtfunktionale Anforderungen beschreiben zusätzliche Rahmenbedingungen.

4.1.1.3 Erzielte Ergebnisse

Ergebnis des Arbeitspakets sind die funktionalen und nichtfunktionalen Anforderungen an selbstlernende Produktionsprozesse. Folgende Auflistung gibt einen Überblick über die Anforderungen:

FA-1: Der Produktionsprozess soll stets im optimalen Betriebspunkt betrieben werden.
Durch unvorhersehbare Änderungen der Produktionsrandbedingungen, sowie der Prozesseigenschaften ändert sich das Zielsystem und die Systemdynamik. Der selbstlernende Produktionsprozess soll sich eigenständigen adaptieren, um unter gegebenen Zielsystem und Systemdynamik den Prozess im optimalen Betriebspunkt zu betreiben.

FA-2: Der Produktionsprozess soll mit möglichst wenig Trainingsdaten gelernt werden.
Die benötigten Trainingsdaten werden bei einem selbstlernenden Produktionsprozess aktiv durch Ausprobieren gewonnen. Durch das Ausprobieren werden Inbetriebnahme-Kosten verursacht, die möglichst minimal gehalten werden sollen.

FA-3: Die Trainingsdaten sollen vorsichtig gewonnen werden. Für einen effizienten, intelligenten Produktionsprozess sind Trainingsdaten aus möglichst unterschiedlichen Betriebsarten erforderlich. Die Trainingsdaten sollen jedoch so gewonnen werden, dass ein sicheres, d.h. vorsichtiges Ausprobieren, gewährleistet ist, ohne dabei die Produktionsmaschine zu gefährden.

FA-4: Der resultierende Prozess soll robust gegenüber unerwarteten Störungen sein. Das Ergebnis des Trainingsprozesses ist eine intelligente Steuerungsstrategie. Die Steuerungsstrategie soll dabei robust sein gegenüber unerwarteten äußeren Einflüssen.

FA-5: Die intelligente Steuerungsstrategie soll generalisieren. Die gelernte Steuerungsstrategie soll eigenständig in der Lage sein, in unbekannten, neuen Situationen einen optimalen Betriebspunkt zu finden. Das heißt, der Produktionsprozess soll auch neue, unbekannte Produktionsaufgaben realisieren.

FA-6: Die Bewertung des Produktionsprozesses findet nachgelagert statt. Die Bewertung der Qualität des Produktionsprozesses findet häufig erst mit einiger Verzögerung statt. Der selbstlernende Produktionsprozess soll diese verzögerte Bewertung auswerten und zur Verbesserung der intelligenten Steuerungsstrategie heranziehen.

FA-7: Der Lernalgorithmus soll selbstlernend sein. Der selbstlernende Produktionsprozess soll eigenständig die nötigen Trainingsdaten sammeln und auswerten. Er soll ohne extern zugeführte und gelabelte Trainingsdaten funktionieren.

FA-8: Der Aktionsraum kann auch kontinuierliche Aktionen umfassen. In der Produktion werden häufig kontinuierliche Aktionen benötigt (z.B. ein beliebiges Drehmoment zwischen 0 und 4 Nm). Der selbstlernende Produktionsprozess soll über die Fähigkeit verfügen, auch mit kontinuierlichen Aktionsräumen zu funktionieren.

FA-9: Die intelligente Steuerungsstrategie soll komplexe Aufgaben behandeln können. Heutige Produktionsprozesse erfüllen üblicherweise komplexe Produktionsaufgaben. Die gelernte Steuerungsstrategie soll über die Fähigkeit verfügen, die Komplexität der Aufgabenstellung hinreichend zu bewältigen.

NFA-1: Kommunikationsschnittstelle zur SPS. Die Software für den selbstlernenden Produktionsprozess soll so in die Produktion eingebunden werden, dass eine gängige Kommunikationsschnittstelle zu einer speicherprogrammierbaren Steuerung bedient werden kann.

NFA-2: Dokumentation der Software und der Methodik. Die Software für den selbstlernenden Produktionsprozess wird hinreichend dokumentiert, um eine Installation auf einem weiteren Computer eigenständig durchzuführen. Weiterhin wird die dahinterliegende Methodik in einem Handlungsleitfaden festgehalten, um das Vorgehen der Algorithmik nachvollziehen zu können.

4.1.2 Arbeitspaket 1.2: Analyse geeigneter Lernverfahren für die Produktion

4.1.2.1 Ziel des Arbeitspakets

Auf Basis von AP 1.1 sollten in AP 1.2 die Eignung existierender Lernverfahren für das Produktionsumfeld untersucht werden. Dazu erfolgt eine Betrachtung existierender Verfahren aus den Bereichen Supervised Learning, Unsupervised Learning und Reinforcement Learning mit Fokus auf eine intelligente, selbstlernende Prozessregelung und Produktionssteuerung. Zur Sicherstellung der Übertragbarkeit wurden die Lernverfahren in die Kategorien diskrete/kontinuierliche Aktionen, sowie diskrete/kontinuierliche Zustände eingeordnet. Im Rahmen des Projekts sollten insbesondere Verfahren für diskrete Aktions- und Zustandsräume, sowie kontinu-

ierliche Aktions- und Zustandsräume tiefgehend untersucht werden. Eingang dieses Arbeitspaketes war das Anforderungskonzept des vorangegangenen Arbeitspaketes. Ausgangsziel des Arbeitspaketes war eine Übersicht über in der Produktion einsetzbare Lernverfahren.

4.1.2.2 Durchgeführte Arbeiten

In diesem Arbeitspaket wurden Verfahren recherchiert und auf ihre Eignung für die Produktion untersucht. Die Eignung für die Produktion ergibt sich hierbei aus dem Erfüllungsgrad der funktionalen Anforderungen. Dazu wurde zunächst eine ausgiebige Recherche der Lernverfahren durchgeführt. Die Verfahren wurden in modellbasiertes Reinforcement Learning, modellfreies Reinforcement Learning, Policy Search, Imitation Learning und inverses Reinforcement Learning (vgl. ***Abbildung 4***) kategorisiert. In einem nächsten Schritt wurden die möglichen Verfahren auf ihre Eignung für die industrielle Anwendung geprüft. Hierfür wurde jedes Lernverfahren bezüglicher der in AP1.1 entwickelten funktionalen und nicht funktionalen Anforderungen geprüft. Im Rahmen eines Webinars wurden die Ergebnisse des Arbeitspaketes dem PA vorgestellt und mit diesem ausführlich diskutiert. Darüber hinaus wurden in enger Zusammenarbeit mit dem PA die drei im Projektantrag beschriebenen Anwendungsszenarien, zwei Forschungsszenarien und ein industrielles Szenario tiefergehend untersucht und diskutiert.

4.1.2.3 Erzielte Ergebnisse

Ergebnis des Arbeitspakets ist zum einen die Übersicht über die möglichen Lernverfahren für die Produktion und eine Bewertung dieser Verfahren bezüglich der funktionalen und nicht funktionalen Anforderungen. Die Verfahren unterscheiden sich einerseits in den zugrundeliegenden Daten und andererseits im Vorgehen zum Trainieren des Machine Learning Modells (vgl. ***Abbildung 4***).

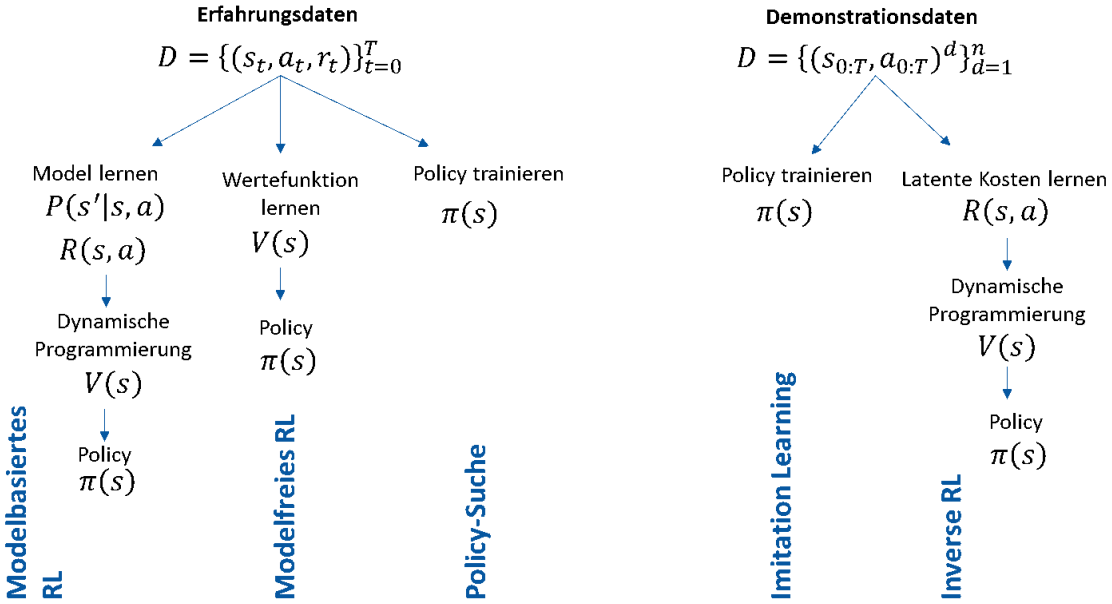

Abbildung 4: Lernverfahren für die Produktion

Wenn Daten vorliegen, die aus dem Zustand (s_t), der Aktion (a_t) und einer Qualitätsbewertung (r_t) bestehen, können modellbasierte und modellfreie Reinforcement Learning Verfahren, sowie Policy Search verwendet werden. Im modellbasierten Reinforcement Learning wird zunächst ein Modell gelernt, welches die Systemdynamik beschreibt (die Zustandsübergänge). Daraus wird im Anschluss die optimale Steuerungsstrategie abgeleitet. Im modellfreien Reinforcement Learning wird zunächst eine Bewertungsfunktion der Zustände

gelernt und anschließend die optimale Steuerungsstrategie abgeleitet. Im Gegensatz dazu wird bei Policy Search direkt die Steuerungsstrategie aus den Daten gewonnen. Grundsätzlich unterscheiden sich die drei Verfahren hinsichtlich der benötigten Trainingsdaten, Robustheit, Generalisierungsfähigkeit und Komplexität der Aufgabenstellung.

Sofern die Daten aus Demonstrationen von optimalen Zustands- und Aktionsabfolgen bestehen, lassen sich Imitation Learning und Inverse Reinforcement Learning durchführen. Im Imitation Learning wird aus den Demonstrationen direkt eine Steuerungsstrategie abgeleitet. Beim inversen Reinforcement Learning wird aus den optimalen Demonstrationen die zugrundeliegende Qualitätsbewertungsfunktion abgeleitet. Dadurch wird das Übertragen des Gelernten auch auf andere technische Systeme möglich.

Tabelle 2 gibt einen Überblick über den Erfüllungsgrad der Anforderungen der einzelnen Lernverfahren.

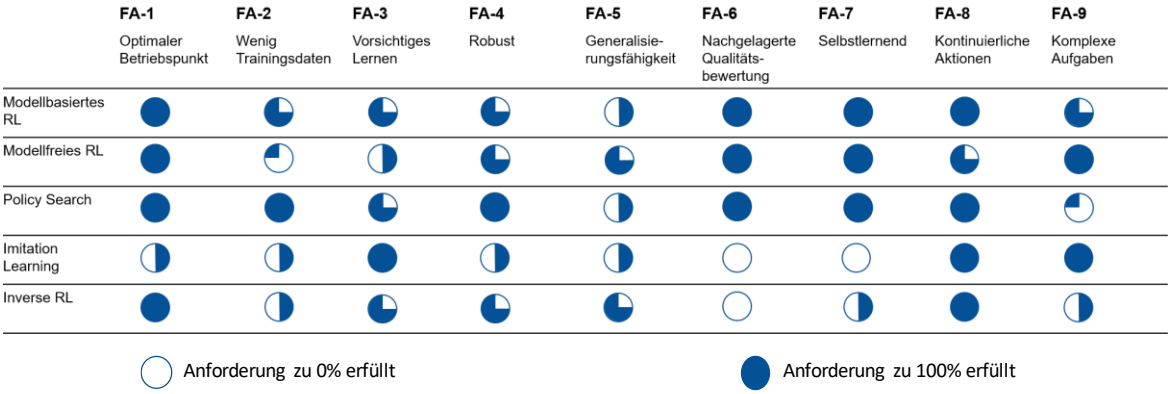

Tabelle 2: Vergleich der Lernverfahren

Zur Anwendung und Validierung der im Projekt entwickelten Methoden wurden drei Use Cases definiert. Es wurden zwei Szenarien an einem Forschungsdemonstrator, ein autonomer Fertigungsablauf (**Szenario A1**) und ein autonomer Montageprozess (**Szenario A2**) ausgewählt. Neben den beiden Forschungsszenarien wurde ein industrielles Szenario (**Szenario B**), die selbstlernende Steuerung eines pneumatischen Schüttgutförderers der AZO GmbH und Co. KG, ausgearbeitet.

Das Szenario A1 betrachtet eine modular aufgebaute Produktion. Eine wachsende Variantenvielfalt des Fertigungsablaufs erfordert, die Produktionsprozesse schnell und flexibel anpassen zu können. Klassische Zielkriterien der Produktionsplanung, wie die Qualität, werden im Zeitalter von Industrie 4.0 um weitere Informationen, wie z.B. der Effizienz oder dem Energieverbrauch ergänzt. Das Ziel dieses Anwendungsszenarios war es, den Ablauf so zu planen, dass die höchste Produktionseffizienz und Qualität der Produkte erreicht wird. Hierfür wurde der Fertigungsablauf im Hinblick auf die Bewertungskriterien Produktionszeit, -qualität und maximale Maschinenbelegung optimiert. Das Szenario betrachtet einen Maschinenverbund, indem jede Maschine nicht nur eine, sondern mehrere Funktionsweisen hat. Die Produktionszeit

und -qualität ist für jede Maschine und Funktionsweise unterschiedlich und unbekannt. Eine Aufgabe besteht dann aus mehreren Schritten. Die autonome Fertigungsplanung bestimmt nun für jeden Schritt die ausführende Maschine. Der Ablauf einer solchen Fertigung ist in ***Abbildung 5*** dargestellt. Das Szenario A1 behandelte den Fall eines diskreten Zustands- und Aktionsraum.

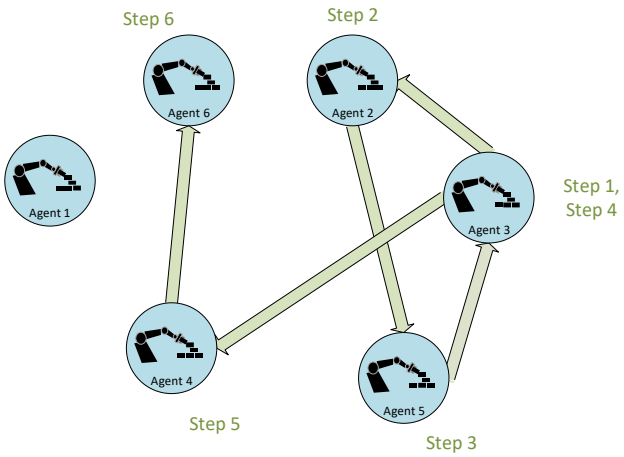

Abbildung 5: Autonomer Fertigungsablauf (Szenario A1)

Das Szenario A2 umfasste einen positions- und kraftgeregelten Montageprozess mit kontinuierlichem Zustands- und Aktionsraum. Als Montageaufgabe wurde eine Fügeaufgabe anhand einer Stift-in-Loch Aufgabe betrachtet, wie in ***Abbildung 6*** dargestellt. Stift-in-Loch Aufgaben erfordern immer noch einen aufwändigen Programmierprozess am Roboter. Zudem sind die resultierenden Roboterbewegungen nur geringfügig fehlertolerant.

Das dritte Szenario (Szenario B) umfasste die selbstlernende Steuerung eines pneumatischen Schüttgutförderers. Dieser Anwendungsfall, ebenfalls mit kontinuierlichem Zustands- und Aktionsraum, ermöglichte die direkte Erprobung in der Industrie. Pneumatische Förderer werden in der Industrie immer da angewendet, wo ein Produkt, genauer gesagt ein Schüttgut, von einem Produktionsort zu einem nächsten transportiert werden muss. Schüttgüter sind beispielsweise Mehl, Sand aber auch Kunststoffpulver.

Mit dem Abschluss des AP1.2 wurde die Anforderungsspezifikation abgeschlossen und somit der **erste Meilenstein** erreicht.

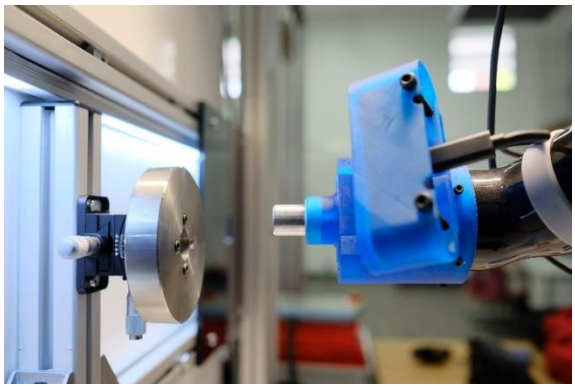

Abbildung 6: Stift-in-Loch Aufgabe (Szenario A2)

4.2 Arbeitspaket 2: Konzeptionierung von Lernverfahren

Ziel des AP2 war die Konzeptionierung der Integration einer Prozessregelung in KMU, die sich durch Intelligenz und Lernfähigkeit auszeichnet und den Anforderungen von KMU genügt. Von zentraler Bedeutung war hierbei die Befähigung von Maschinen, Anlagen und Teilkomponenten zu intelligentem, selbstlernendem Verhalten. Dazu sollten auf Basis der erhobenen Anforderungen die relevanten Betrachtungsaspekte in Bezug auf eine lernfähige Regelung ausgestaltet werden. Kernziel war die Untersuchung und Konzeptionierung von Lernstrategien für verschiedenste Produktionsszenarien.

4.2.1 Theoretische Grundlagen

4.2.1.1 Reinforcement Learning

Reinforcement Learning ermöglicht es einer Maschine bzw. einer Anlagensteuerung, einen komplexen Zusammenhang selbstständig zu erlernen. Dafür muss nicht der gesamte Prozess bekannt sein. Stattdessen wird der Lösungsweg Schritt für Schritt durch Ausprobieren gefunden und optimiert. Im Folgenden werden das Prinzip und die notwendigen Begriffe definiert.

Die formale Darstellung des Prinzips ist in ***Abbildung 7*** dargestellt. Ein Agent wirkt über einen oder mehrere Aktuatoren auf seine Umgebung ein. Anhand der Kostenfunktion wird diese Aktion bewertet. Der Agent erhält als Feedback den neuen Folgezustand und als Bewertung einen Kostenwert auf Basis der Kostenfunktion. Auf dieser Grundlage führt er im nächsten Iterationsschritt erneut eine Aktion aus. Dieses Vorgehen wird solange iteriert bis ein hinreichend gutes Ergebnis erzielt wurde.

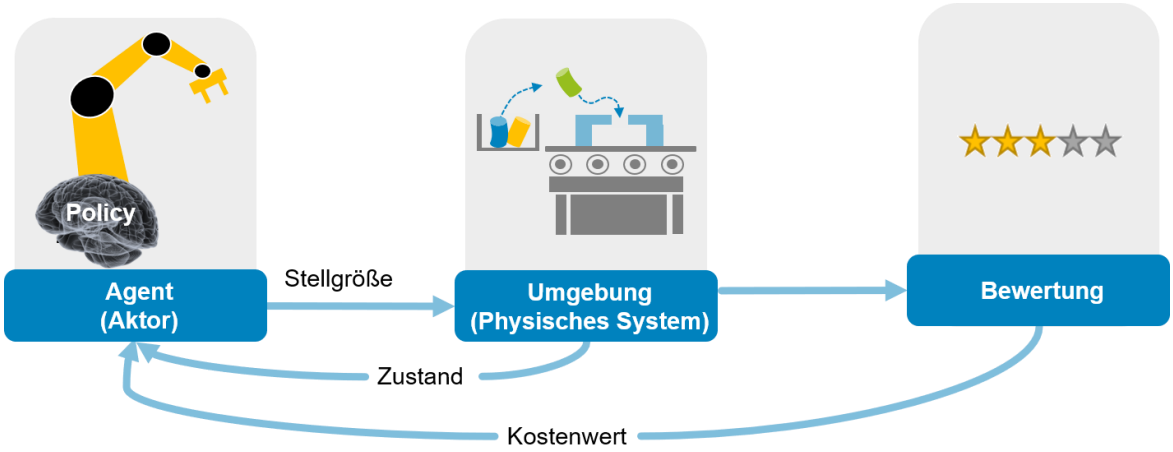

Abbildung 7: Der Reinforcement Learning Zyklus: Der Agent wählt eine Aktion, bzw. bestimmt seine Stellgrößen und wirkt so auf die Umgebung ein. Als Rückmeldung bekommt er einen neuen Zustand und eine Bewertung der durchgeführten Aktion zurück.

Der **Agent** ist ein autonomes Softwareprogramm, das im Reinforcement Learning die Rolle des Entscheiders übernimmt. Er bekommt zu jedem Zeitschritt Informationen über den aktuellen Zustand der Umwelt bzw. des Systems und eine Belohnung für die Ausführung der letzten Aktion. Mithilfe dieses Zustands und der aktuellen Kostenfunktion bestimmt der Agent die Aktion für den nächsten Zeitschritt. Die **Umgebung**, bzw. das physische System, ist durch das zu steuernde System gegeben. Dies kann z.B. eine Produktionsstraße sein. Dieses System wird charakterisiert durch einen aktuellen Zustand und kann durch Aktionen des Agenten direkt beeinflusst werden. Der **Zustand** des Systems wird über die Sensorik des physischen Systems beschrieben. Abhängig vom Prozess kann dies eine Kamera am Endeffektor eines Roboterarms, Temperatur-, Druck-, Strahlungs- oder beliebige andere Sensoren beinhalten.

Der **Zustandsraum** beschreibt alle Zustände, die das System einnehmen kann. Damit ist er unter anderem vom Messbereich der Sensoren abhängig. Eine **Aktion** im Reinforcement Learning Kontext beinhaltet die Signale für alle einstellbaren Aktuatoren des Systems. Der **Aktionsraum** beschreibt dementsprechend den möglichen Einstellungsraum der Aktuatoren. Durch die **Kostenfunktion** wird die aktuelle Prozessgüte beschrieben. Diese Funktion gibt eine Bewertung des aktuellen Zustands und der durchgeführten Aktion des Agenten in Form einer Belohnung oder Bestrafung aus. Die Kostenfunktion wird in jedem Schritt ausgewertet. Anschließend werden die Aktionen des nächsten Schrittes mithilfe der aktuellen Kostenfunktion ermittelt. Die **Policy** beschreibt im Reinforcement Learning die Strategie des Agenten. Diese ist abhängig vom aktuellen Zustand des Systems. So wird eine Strategie gelernt, welche auf verschiedene Zustände optimal reagieren kann. Der Begriff Policy beschreibt im Reinforcement Learning Kontext also eine intelligente Steuerungsstrategie.

Im Reinforcement Learning unterscheidet man zwischen zwei Klassen von Lernverfahren, den modellbasierten und den modellfreien Verfahren. Modellbasierte Verfahren lernen neben einer Policy ein Modell der Umwelt aus dem in der Vergangenheit beobachteten Verhalten bestehend aus Umgebungszuständen und ausgeführten Aktionen. Dieses Modell kann dann genutzt werden, um die optimale Aktionsabfolge zu planen und so die Policy zu verbessern [34]. In diesem Projekt wurden sowohl modellfreie als auch modellbasierte Verfahren untersucht und auf die ausgewählten Use Cases angewandt. Im Folgenden wird der modellfreie Ansatz **Deep-Q-Network (DQN)**, welcher für das Szenario A1 gewählt wurde und der modellbasierte Ansatz **Guided Policy Search (GPS)** eingeführt. Ein auf GPS basiertes Verfahren wurde für die Szenarien A2 und B verwendet.

Q-Learning ist ein modellfreier Reinforcement Learning Ansatz. **Deep Q-Network (DQN)** ist eine Erweiterung von Q-Learning für hochdimensionale Zustandsräume [35][33]. Darin wird die Q-Funktion durch ein neuronales Netzwerk mit den Parametern θ dargestellt.

DQN verwendet eine Technik namens Experience Replay, die den Verstärkungsprozess stabilisiert. Experience Replay ist ein biologisch inspirierter Mechanismus, der anstelle der neuesten Samples Daten aus der vorangegangenen Erfahrung zufällig für das Training des Q-Netzwerks auswählt. In kontinuierlichen Aktionsräumen ist jedoch eine Minimierung über die angenäherte Q-Funktion nicht möglich, da es unendlich viele Aktionen zu berücksichtigen gibt. Daher können die naiven Verfahren nur mit diskreten Aktionsräumen umgehen.

Deep Reinforcement Learning Algorithmen sind in der Lage, durch Verwendung von neuronalen Netzen als allgemeine Policy-Repräsentation eine Vielzahl von Manipulations- und Bewegungsaufgaben zu lösen. **Guided Policy Search** (**GPS**) ist ein modellbasierter Deep-Reinforcement-Learning Ansatz, der modellbasiertes trajektorienzentrisches Reinforcement Learning verwendet, um eine hohe Dateneffizienz zu erreichen [24]. Der im folgenden beschriebene Algorithmus ist **Mirror-Descend GPS** (**MDGPS**) [36], die grundlegenden Merkmale gelten aber auch für andere GPS-Varianten.

Der GPS-Trainingsprozess besteht aus zwei Phasen, einer Kontrollphase (C-Schritt), in der lokale Policies für mehrere Initialzustände optimiert werden, und einer überwachten Phase (S-Schritt), in der die globale Politik trainiert wird.

Im C-Schritt nimmt GPS Trajektoriensamples von jedem Initialzustand. Das heißt, das System wird in einen definierten Zustand gebracht, wovon ausgehend für eine gewisse Anzahl an Zeitschritten eine gelernte Policy (intelligente Steuerungsstrategie) ausgeführt wird. Im ersten Trainingsschritt wird hier die initiale Policy ausgeführt. Die dabei besuchten Zustände und die jeweils von der Policy gewählten Aktionen bilden die Trajektoriensamples. Zu diesen Samples wird dann je ein lokales Dynamikmodell pro Initialzustand angepasst. Die Verwendung einer linearen Bayesschen Regression vermeidet dabei ein zu starkes Overfitten der Dynamiken auch bei wenigen Trainingsdaten [36]. Anschließend werden unter diesen lokalen Dynamiken kostenminimale lineare Controller mittels eines iterativen linear-quadratischen Regulators abgeleitet [36], [37].

Im S-Schritt wird dann eine globale Policy trainiert, die lokalen Policies zu imitieren. Dafür werden zunächst aus den lokalen Policies Zustands-Aktions-Paare gewonnen. Mit diesen Trainingsdaten wird anschließend ein neuronales Netz mittels eines Gradientenabstiegsverfahrens trainiert, welches so in der Nähe der lokalen Policies diese imitiert, und außerhalb der gesampelten Trajektorien begrenze Generalisierungseigenschaften aufweist.

Da GPS nur lokale Dynamikmodelle lernt, hat es sich als sehr dateneffizient erwiesen, im Gegensatz zu modellfreien Algorithmen [24], [38]. GPS wurde bisher erfolgreich auf verschiedene Robotik-Problem angewandt [24], [36].

4.2.2 Arbeitspaket 2.1: Gestaltung von Systemarchitekturen

4.2.2.1 Ziel des Arbeitspakets

Auf Basis der Anforderungen, Analyse und Gestaltung von Lernverfahren (AP 1) sollte in diesem AP eine ganzheitliche Systemarchitektur erarbeitet werden, die den Aufbau und die notwendigen Komponenten zur Integration intelligenter selbstlernender Prozessregelungen im Produktionsumfeld beschreiben. Die Entwicklung einer solchen Systemarchitektur sollte es produzierenden Unternehmen ermöglichen, ein vollständiges Bild der erforderlichen Systemkomponenten sowie deren Wechselwirkungen zu erhalten und nachzuvollziehen.

4.2.2.2 Durchgeführte Arbeiten

Auf der Grundlage, der im Rahmen des Projekts definierten Use Cases und der in AP 1.2 erarbeiteten Lernverfahren wurden in diesem Arbeitspaket Systemarchitekturen für die einzelnen Use Cases erarbeitet. Hierbei wurden zwei unterschiedliche Systemarchitekturen abhängig von den Anforderungen des jeweiligen Reinforcement Learning Verfahrens entwickelt. Für jede Systemarchitektur wurden dabei drei Hauptkomponenten entwickelt. Reinforcement Learning Verfahren interagieren direkt mit ihrer Umgebung, dies erfordert eine Schnittstelle zur Hardware oder zur Simulation. Über diese Schnittstelle werden zum einen Stellgrößen, also Aktionen, an die Hardware oder Simulation gesendet und zum anderen der Systemzustand regelmäßig abgefragt. Die Systemarchitektur beinhaltet daher eine Komponente, welche die Kommunikationsschnittstelle zwischen Maschine und Reinforcement Learning Algorithmus implementiert. Die beim Training einer selbstlernenden Steuerung anfallenden Daten müssen effizient gespeichert und für den Algorithmus aufbereitet werden. Hierfür wurde für alle Systemarchitekturen eine Komponente als Datalake/Datawarehouse entwickelt. Die dritte Komponente der Systemarchitektur enthält den eigentlichen Trainingsprozess.

4.2.2.3 Erzielte Ergebnisse

Ergebnis des AP 2.1 sind zwei Systemarchitekturen zugeschnitten auf die definierten Use Cases. Die autonome Ablaufplanung stellt ein Szenario mit diskretem Zustands- und Aktionsraum dar. Dies führt zu einem im Vergleich zum Szenario A2 und B verringertem Lösungs- und Parameterraum und ermöglicht die Verwendung eines modellfreien Reinforcement Learning Verfahrens zum Erlernen der Ablaufplanung. Modellfreie Verfahren sind im Allgemeinen weniger dateneffizient als modellbasierte Verfahren. Daher eignen sie sich besonders gut für Anwendungsfälle mit einem solchen geringen Parameterraum und der Möglichkeit große Datenmengen durch eine Simulation zu generieren. Der große Vorteil von modellfreien Verfahren ist ihre im Vergleich zu modellbasierten Verfahren erhöhte Generalisierungsfähigkeit. Die Systemarchitektur für den Anwendungsfall A1, dargestellt in ***Abbildung 8***, besteht aus den drei oben genannten Komponenten; der Schnittstelle zwischen Maschinenintelligenz und Simulation, dem Datalake und dem eigentlichen Reinforcement Learning Algorithmus. Zwischen Maschine und Maschinenintelligenz, bzw. der intelligenten Steuerungsstrategie (Policy), werden Steuerungssignale und Systemzustände ausgetauscht.

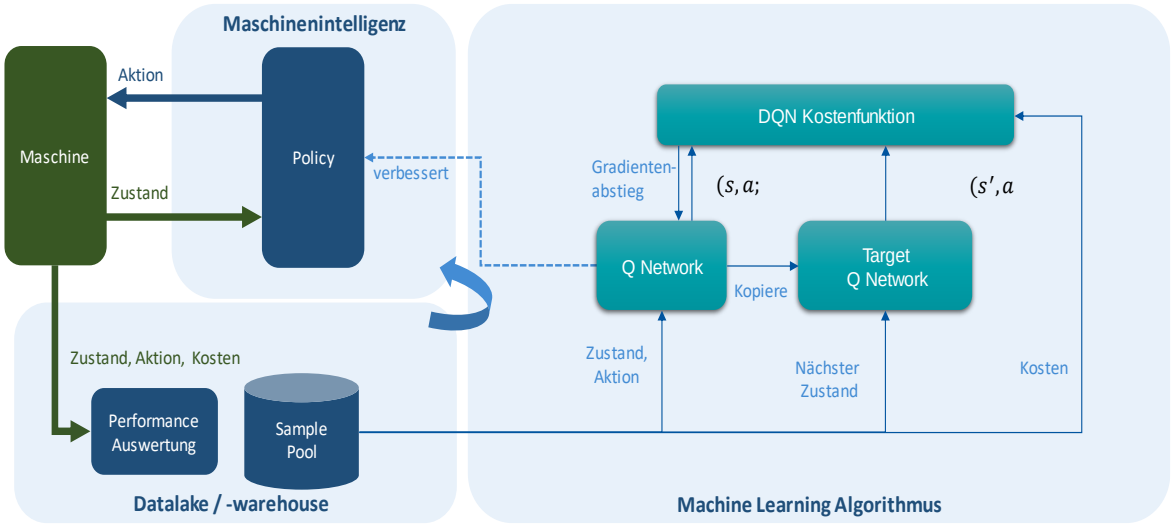

Abbildung 8: Systemarchitektur Use Case A1: autonome Ablaufplanung

Für das Szenario A2 und das Szenario B wurde ein modellbasiertes Reinforcement Learning Verfahren ausgewählt. Wie oben erwähnt, zeichnen sich solche modellbasierten Verfahren durch eine hohe Dateneffizienz aus. Dies ist insbesondere für das Szenario des pneumatischen Schüttgutförderers unabdinglich, da das Sammeln von Trainingsbeispielen unter realen Bedingungen mit einem hohen Personal-, Zeit- und Materialaufwand verbunden ist. Die Systemarchitektur A2 (vgl. ***Abbildung 9***) besteht wie auch A1 aus drei Komponenten: Der Schnittstelle zur Maschinenintelligenz, dem Datalake und dem modellbasierten Reinforcement Learning Algorithmus. Für die Schnittstelle zur Hardware gibt es bei dieser Architektur verschiedene Möglichkeiten. Für den pneumatischen Schüttgutförderer wurde eine Schnittstelle zu einem Beckhoff Industrie-PC implementiert. Dieser IPC wiederrum stellt die Schnittstelle zwischen der OPC-UA Schnittstelle der Anlagensteuerung und der SPS dar und fungiert als Sammelstelle für die Daten der Aktorik und der Sensorik. Außerdem ist der IPC verantwortlich für die Einhaltung der technischen Grenzen der Anlage, sorgt also dafür, dass die Steuersignale innerhalb wohldefinierter Grenzen liegen.

Für den autonomen Montageprozess wurde hier eine weitere Kommunikationsschnittstelle über das Robot Operating System (ROS) implementiert. Dabei handelt es sich um ein Open-Source Framework, welches unter anderem eine gute Infrastruktur für die Kommunikation zwischen verschiedenen Prozessen bietet. Über eine dieser möglichen Schnittstellen werden die Aktionen als Stellgrößen von der Maschinenintelligenz an die Maschine gesendet, welche ihren aktuellen Zustand zurückgibt. Die einzelnen Zustände, Aktionen und Belohnungen der Maschine werden durchgehend im Datalake gespeichert und dort aufbereitet. Aus dem Sample Pool werden dann aufbereitet Datenpunkte ausgesucht und an den Reinforcement Learning Algorithmus weitergeleitet. Zunächst werden Zustände und Aktionen benutzt, um ein Modell der Maschine zu lernen. Mit diesem Modell und der Information über die Belohnung kann dann eine Trajektorienoptimierung stattfinden. Ausgehend vom Resultat des ML-Algorithmus und des Imitation Learning wird daraufhin die aktuelle intelligente Steuerungsstrategie verbessert. Dabei besteht die Möglichkeit, den ganzen Trainingsprozess oder auch nur Teile davon in der Cloud auszuführen. Des Weiteren können einige der aufwendigeren Rechenprozesse auf einem Grafikprozessor (GPU) ausgeführt werden, um eine schnellere Durchführung eines Trainingsschrittes zu erreichen.

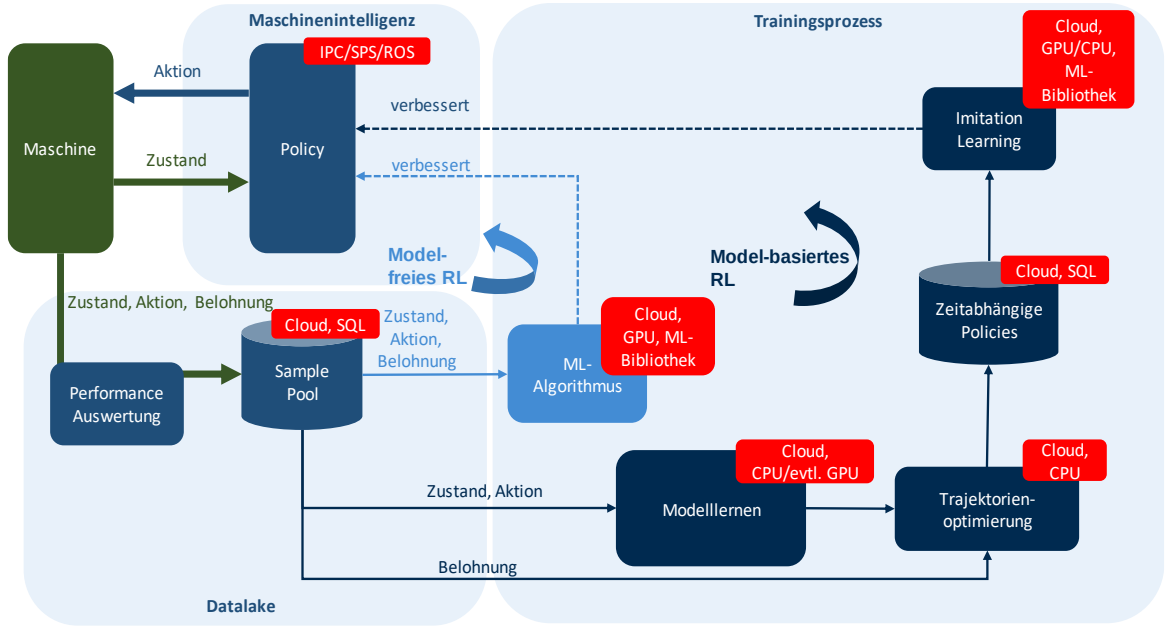

Abbildung 9: Systemarchitektur für modellbasiertes RL (Use Case A2 und B)

4.2.3 Arbeitspaket 2.2: Gestaltung von Zielsystemen für Lernverfahren im Produktionsumfeld

4.2.3.1 Ziel des Arbeitspakets

Eine intelligente lernfähige Steuerung benötigt ein definiertes Zielsystem. Dieses Zielsystem-setzt sich aus Eigenschaften des Produkts und gewünschten Randbedingungen für den Prozess zusammen. Ziel dieses Arbeitspakets war die Gestaltung von für das Produktionsumfeld geeigneten Zielsystemen.

4.2.3.2 Durchgeführte Arbeiten

Das Zielsystem für Reinforcement Lernverfahren wird durch eine Funktion beschrieben, welche häufig Kostenfunktion genannt wird. Innerhalb dieses Arbeitspakets wurden zunächst allgemeine Anforderungen und Gestaltungsprinzipien einer Kostenfunktion im Produktionsumfeld aufgestellt. Zu diesen allgemeinen Anforderungen zählt eine ausreichende Zielerreichung bei gleichzeitiger Erfüllung aller notwendigen Sicherheitskriterien und der Umgang mit heterogenen Daten. Eine Zielfunktion muss dann individuell für den spezifischen Anwendungsfall entwickelt werden. Im Rahmen dieses Arbeitspakets wurde eine individuelle Zielfunktion für jeden Anwendungsfall entwickelt.

4.2.3.3 Erzielte Ergebnisse

Ergebnis dieses Arbeitspakets ist zunächst die Untersuchung und Aufstellung von allgemeinen Gestaltungsprinzipien für Zielfunktionen im Produktionsumfeld. Hierbei wurden die folgenden Punkte als essentiell erarbeitet:

- Zielerreichung
- Stabiles Verhalten ohne Überschwingen
- Vermeidung von kritischen Systemzuständen

Die gleichzeitige Zielerreichung und die Vermeidung von kritischen Systemzuständen führen zu einer Unterscheidung von zu **minimierenden** und zu **vermeidenden** Kosten. Häufig liegen der Zielzustand und kritische Systemzustände nah beieinander. Dies führt zu einem Konflikt,

da die Gesamtperformance in der Nähe des Zielzustandes zwar sehr gut ist, kritische Systemzustände aber gleichzeitig vermieden werden müssen. Um diesen Konflikt aufzulösen, wurde im Rahmen dieses Arbeitspakets ein Zielsystem mit Schalter entwickelt. Dieser Schalter ist im Normalfall 1 und nimmt im kritischen Fall einen sehr hohen Wert an. So können Notfälle vermieden werden. ***Abbildung 10*** zeigt einen solchen Schalter in Abhängigkeit von der relevantesten Zustandsdistanz. ***Abbildung 11*** zeigt die so entstehende Funktion. Die Funktion ist überall 1, außer in der direkten Umgebung des kritischen Zustands. In dieser Umgebung werden die durch den Schalter entstehenden Zustandskosten sehr groß und so können kritische Zustände vermieden werden.

Zustands-kosten mit Schalter

Relevanteste Zustandsdistanz

$$new\_distance = criticSwitch * distance$$

Schaltende Funktion: $criticSwitch = 1 + \left[\frac{criticalThreshold}{criticalStateValue + \mu * criticalThreshold}\right]^2$

Heuristik für μ: $\mu = 0.5 * criticalThreshold$

Abbildung 10: Zustandskosten mit Schalter

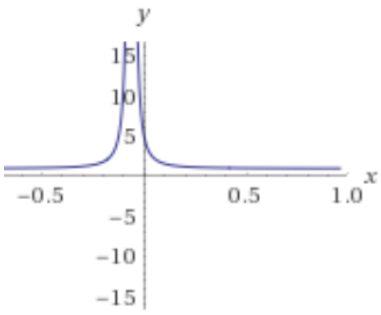

Abbildung 11: Schaltende Funktion des Zustandskostenschalters

Weiter wurde in diesem Arbeitspaket der Unterschied zwischen einem kontinuierlichen Kostenfeedback oder einem diskreten Kostenfeedback bei Zielerreichung untersucht. Hier wurde für alle Szenarien ein kontinuierlicher Kostenwert als überlegen erarbeitet. So führt ein kontinuierliches Kostenfeedback zu einem glatten Gradienten im Trainingsprozess des neuronalen Netzes. Dies bedeutet eine enorme Verbesserung der Dateneffizienz und der Trainingsperformanz. ***Abbildung 12*** zeigt den Unterschied zwischen beiden Kostentypen am Beispiel des autonomen Montageprozesses.

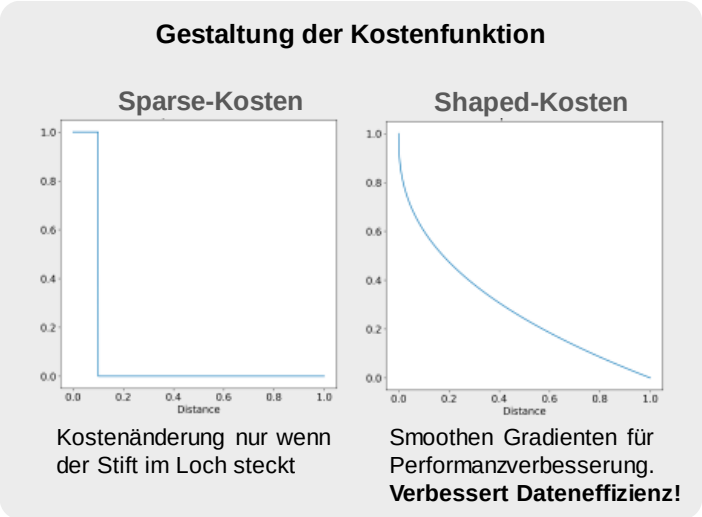

Abbildung 12: Kosteneffizienz durch Kosten-Shaping

Auf Basis der allgemeinen Erkenntnisse sind dann drei auf den jeweiligen Anwendungsfallzugeschnittene Kostenfunktionen entstanden.

Für die autonome Ablaufplanung wurde in AP 2.1 ein modellfreier Reinforcement Learning Ansatz basierend auf Q-Learning gewählt. In Q-Learning Methoden wird eine Funktion approximiert, die für jeden Zustand und einer ausgewählten Aktion die Kosten ausgibt. Damit stellt die Q-Funktion im Grunde eine individuelle Kostenfunktion dar, die während des Trainingsprozesses gelernt wird. Somit beinhaltet die Kostenfunktion (vgl. ***Abbildung 13***) für diesen Anwendungsfall den Approximationsfehler der Q-Funktion.

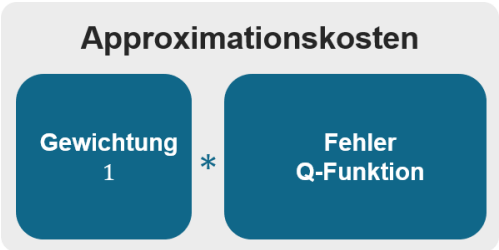

Abbildung 13: Kostenfunktion für den Anwendungsfall A1 (autonome Ablaufplanung)

Im Szenario A2 hat die Kostenfunktion die Aufgabe den Endeffektor des Montageroboters an die Zielposition zu bringen. Hierbei soll eine möglichst glatte Bewegung und eine hohe Positioniergenauigkeit entstehen. Die Kostenfunktion besteht in diesem Anwendungsfall aus zwei Termen, den Aktionskosten und den Zustandskosten. Die Zustandskosten betrachten zum einen die räumliche Distanz des Endeffektors zum Zielpunkt und zum anderen die Geschwindigkeitsdistanz zur Zielgeschwindigkeit. Der Zielpunkt wird durch die räumlichen Koordinaten des Endeffektors am Ende des Fügeprozesses definiert. Am Ende des Montageprozesses soll der Endeffektor in Ruhe sein, daher wird die Zielgeschwindigkeit am Ende des Prozesses zu null gesetzt. Für die Aktionskosten werden die Drehmomente in den Gelenken des Montageroboters betrachtet. Die Berücksichtigung der Drehmomente führt zu einer Steuerungsstrategie, die die erforderliche Fügeaufgabe ausführt bei möglichst geringem Bewegungsaufwand und daraus resultierendem geringem Energiebedarf. Die Gesamtkosten bestehen dann aus der Summe der einzelnen Kostenkomponenten multipliziert mit einem Gewichtungsfaktor. Die

Gewichtung der einzelnen Terme spielt eine essentielle Rolle und muss für jeden Anwendungsfall neu gesetzt werden. Eine graphische Darstellung der Kostenfunktion für den Montageroboter ist in ***Abbildung 14*** dargestellt.

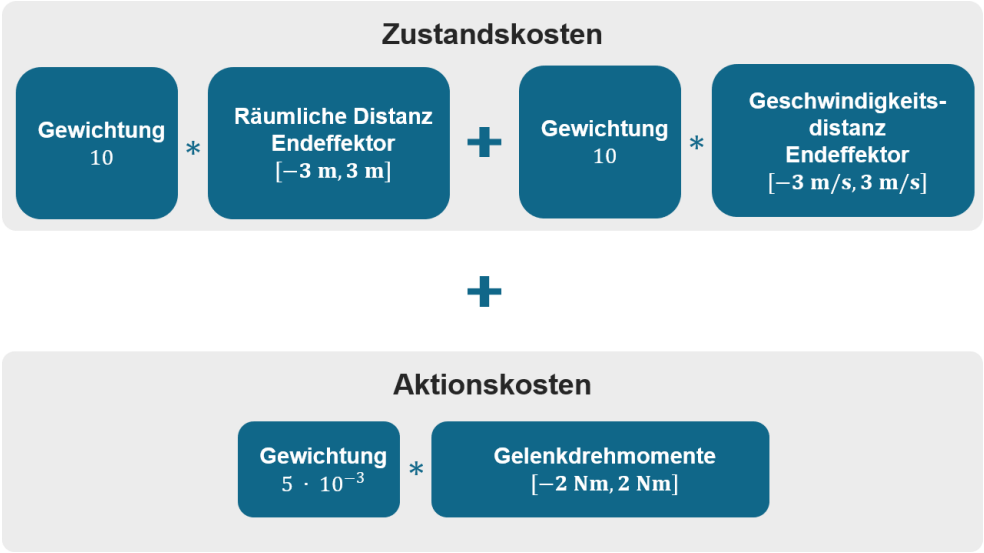

Abbildung 14: Kostenfunktion für den Anwendungsfall A2 (autonomer Montageprozess)

Im Szenario B des pneumatischen Schüttgutförderers soll die Kostenfunktion ein ruhiges, aber effizientes Förderverhalten sicherstellen. Hierzu wurden die Kosten erneut aus den Zustandskosten und den Aktionskosten zusammengesetzt. Die Zustandskosten bestehen aus drei Teilen. Zunächst wird die Distanz zwischen dem Zielgebläsedruck und dem aktuellen Gebläsedruck gewichtet mit der aktuellen Produktgeschwindigkeit betrachtet. Der Zielgebläsedruck war dabei beim Schüttgutförderer ein negativer Wert, da es sich um eine Saugförderanlage handelte. Die Gewichtung mit der Produktgeschwindigkeit führt zu einem geringen Kostenwert bei hoher Produktgeschwindigkeit. Eine hohe Produktgeschwindigkeit bedeutet bei einem pneumatischen Schüttgutförderer eine hohe Förderleistung.

Der zweite Term der Zustandskosten betrachtet die Distanz zwischen einem Sollfördergewicht und dem aktuellen Fördergewicht. Dieser Term sorgt ebenfalls für eine hohe Förderleistung durch die entsprechende Maximierung der geförderten Produktmasse. Die dritte Komponente der Zustandskosten ist essentiell, um eine abrupte Verstopfung des Rohrs, sogenannte Stopfer, in der Förderung zu vermeiden. Stopfer treten immer dann auf, wenn mehr Material eingeleitet wird als über den aktuellen Luftstrom abtransportiert werden kann. Eine solche Verstopfung passiert in der traditionellen Förderung häufig unerwartet und ist meist nicht automatisch zu beheben. Oftmals kann das Rohr nur durch einen manuellen Eingriff wieder befreit werden. Der dritte Term betrachtet erneut das Fördergewicht, diesmal gewichtet mit der Luftgeschwindigkeit. Eine Luftgeschwindigkeit nahe Null ist ein Indikator für einen bevorstehenden Stopfer. Dabei ist die Luftgeschwindigkeit besser zur Stopfervermeidung geeignet als die Produktgeschwindigkeit, da die Produktgeschwindigkeit mit einer größeren Verzögerung auf Prozessveränderungen reagiert als die Luftgeschwindigkeit. Die Aktionskosten bestehen im betrachteten Anwendungsfall aus der Drehzahl des Gebläses und der Drehzahl der Dosierschnecke. Die Aktionskosten tragen damit zu einem ruhigen Förderverhalten und einem geringen Energiebedarf bei. Analog zu der Kostenfunktion im autonomen Montageprozess wurden auch hier beide Kostenanteile gewichtet und aufaddiert. Die gesamte Kostenfunktion ist in ***Abbildung 15*** dargestellt.

Diese Struktur garantiert eine symmetrische positiv definite Kostenmatrix, welche durch den auf Guided Policy Search basierten Reinforcement Learning Algorithmus gefordert ist.

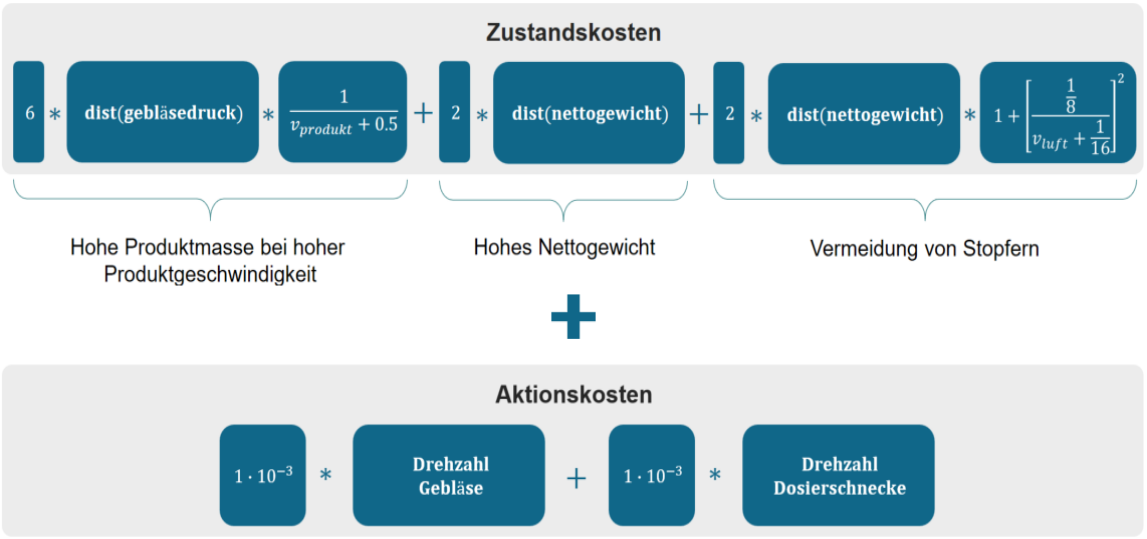

Abbildung 15: Kostenfunktion für den Anwendungsfall B (pneumatischer Schüttgutförderer)

4.2.4 Arbeitspaket 2.3: Gestaltung von Lernverfahren für die Prozessregelung

4.2.4.1 Ziel des Arbeitspakets

Die Systemarchitektur aus AP 2.1 legt die Grundlage zum kontextbasierten Lernen zur Prozessregelung. Dazu wurden in diesem Arbeitspaket Modelle und Methoden aufgestellt, die das Erlernen einer intelligenten Prozessregelung ermöglichen. Besondere Randbedingungen der Produktion wurden berücksichtigt und diese sind in die Entwicklung einer adaptiven, intelligenten und selbstlernenden Prozessregelung miteingeflossen. Dieses Arbeitspaket verfolgte das Ziel, eine intelligente Prozessregelung für einen zielführenden und effizienten Produktionsablauf zu schaffen, ohne dabei ein genaues Verständnis der Wechselwirkungen zwischen einzelnen Entitäten im Gesamtsystem zu besitzen. Ein besonderes Augenmerk lag darin, Lernverfahren für die exemplarischen industriellen Anwendungsszenarien (Schüttgutförderer, kraftgeführter Montageprozess) algorithmisch zu beschreiben und die erforderlichen Rahmenbedingungen zur Einbettung in das Gesamtsystem festzuhalten.

4.2.4.2 Durchgeführte Arbeiten

In diesem Arbeitspaket wurden, aufbauend auf dem Anforderungskonzept und der Analyse der Lernverfahren aus dem ersten Arbeitspaket, zunächst die Anforderungen an Lernmethoden für die industrielle Anwendung von Reinforcement Learning weiter geschärft. Aktuelle Methoden und Modelle wurden hinsichtlich dieser Anforderungen evaluiert und neue, auf die Anforderungen zugeschnittene Lernverfahren für die intelligente Prozessregelung entwickelt. Hierbei lag der Fokus in diesem Arbeitspaket auf der Beschreibung der Lernverfahren, welche in den regelmäßigen Webinaren und Projekttreffen vorgestellt wurden.

4.2.4.3 Erzielte Ergebnisse

Es wurden Anforderungen für das industrielle Reinforcement Learning entwickelt. Hierbei zeichnet sich das industrielle Reinforcement Learning durch die besonderen Anforderungen hinsichtlich Robustheit, Sicherheit und Dateneffizienz der Algorithmen aus (vgl. **Abbildung 16**).

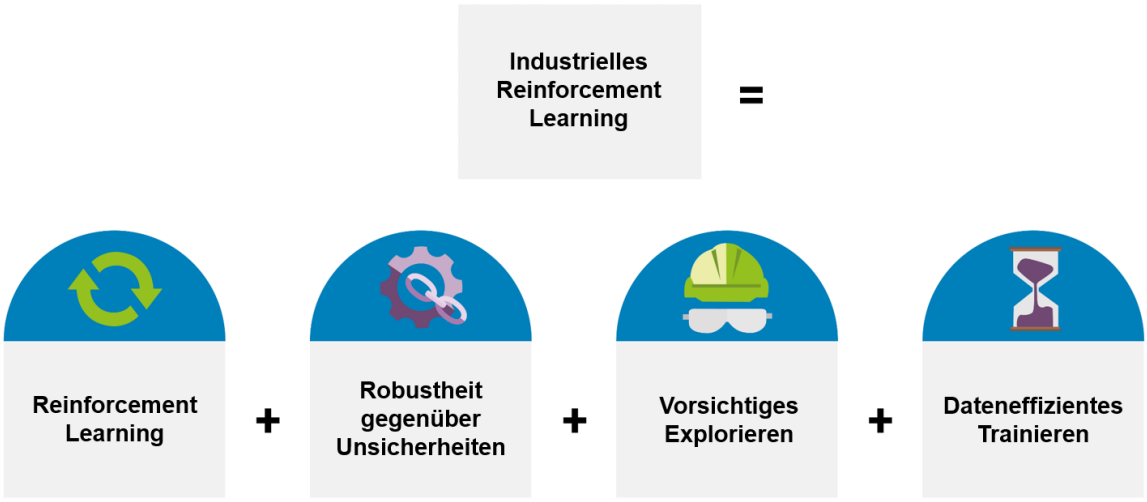

Abbildung 16: Anforderungen des industriellen Reinforcement Learning

Das Trainieren eines Reinforcement Learning Modells benötigt eine gewisse Menge an Ressourcen. Diese Ressourcen, oft Trainingskosten genannt, sollen möglichst geringgehalten werden. Der wichtigste Kostenfaktor ist der zeitliche Aufwand. Für das Trainieren einer Reinforcement Learning Strategie muss eine große Menge an Daten mithilfe von Testdurchläufen des realen Prozesses generiert werden. Dies erfordert, dass der reale Prozess kurz ist und häufig wiederholt durchgeführt werden kann. Zu den Trainingskosten zählen auch die materiellen Ressourcen, beispielsweise Rohstoffe, die ein Prozess verbraucht. Auch diese Kosten sollten so gering wie möglich gehalten werden. Insgesamt stellen diese Trainingskosten besondere Anforderungen an die Dateneffizienz des gewählten Algorithmus.

Alternativ zum realen Prozess können Trainingsdaten mithilfe einer Simulation generiert werden. Ist eine aussagekräftige Simulation des Prozesses möglich, erleichtert dies die Generierung der Trainingsdaten. In einer Simulation spielt der zeitliche Aufwand eine untergeordnete Rolle, da Simulationen nicht zwingend in Echtzeit laufen müssen und parallel ausgeführt werden können. Fehlzustände des Prozesses stellen in einer Simulation keine Gefahr dar und Rohstoffkosten können gänzlich vernachlässigt werden.

Während des Trainingsprozesses einer selbstlernenden Steuerungsstrategie werden neue und potentiell instabile Parametereinstellungen ausprobiert. In einer Simulation können diese Parametereinstellungen gefahrlos getestet werden. Die Anwendung in einem realen System erfordert allerdings ein vorsichtiges Ausprobieren der unbekannten Parametereinstellungen, um die Sicherheit zu jedem Zeitpunkt zu gewährleisten. Hierfür müssen die realen Systeme möglichst fehlertolerant sein bzw. die Fehler müssen rechtzeitig erkannt und behoben werden können. Außerdem muss bereits bei der Wahl des Algorithmus die vorsichtige Exploration der Umgebung beachtet werden.

Die Fehlertoleranz des Systems spielt auch bei der Robustheit gegenüber Unsicherheiten eine große Rolle. Hier ist es allerdings wichtig, den Reinforcement Learning Ansatz so zu wählen oder zu designen, dass dieser robust auf Unsicherheiten in der Umwelt reagiert.

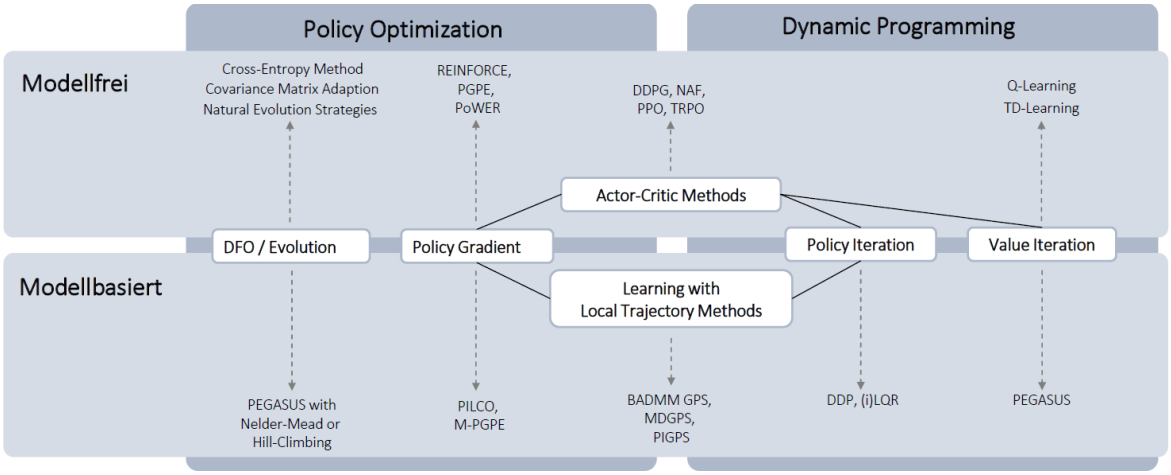

Abbildung 18: Klassifikation der verschiedenen Reinforcement Learning Methodiken

Ein weiteres wichtiges Ergebnis dieses Arbeitspakets ist eine Klassifikation der verschiedenen Reinforcement Learning Methodiken (vgl. **Abbildung 18**). Hier wird zum einen zwischen modellfreien und modellbasierten Methoden unterschieden. Wie in Abschnitt 4.2 beschrieben, sind modellbasierte Methoden häufig deutlich dateneffizienter und dadurch für reale Prozesse ohne Simulation und mit kontinuierlichem Zustands- und Aktionsraum geeignet. Modellfreie Methoden besitzen gegenüber modellbasierten Methoden meist eine höhere Generalisierungsfähigkeit, da sie keine Annahme über das zu approximierende Modell treffen. Weiter können Reinforcement Learning Methoden in Verfahren basierend auf einer Policy Optimierung und basierend auf Dynamic Programming unterteilt werden. Die aktuell performantesten Methoden verbinden beide Aspekte, dazu gehören Aktor-Kritiker Verfahren und trajektorienbasierte Verfahren wie Guided Policy Search. Diese aktuellen Reinforcement Learning Methoden wurden hinsichtlich ihrer Dateneffizienz und ihrer Robustheit evaluiert (vgl. **Abbildung 17**). Hierbei wurde deutlich, dass keine der existierenden Verfahren die Anforderungen hinsichtlich Robustheit und Dateneffizienz erfüllen kann.

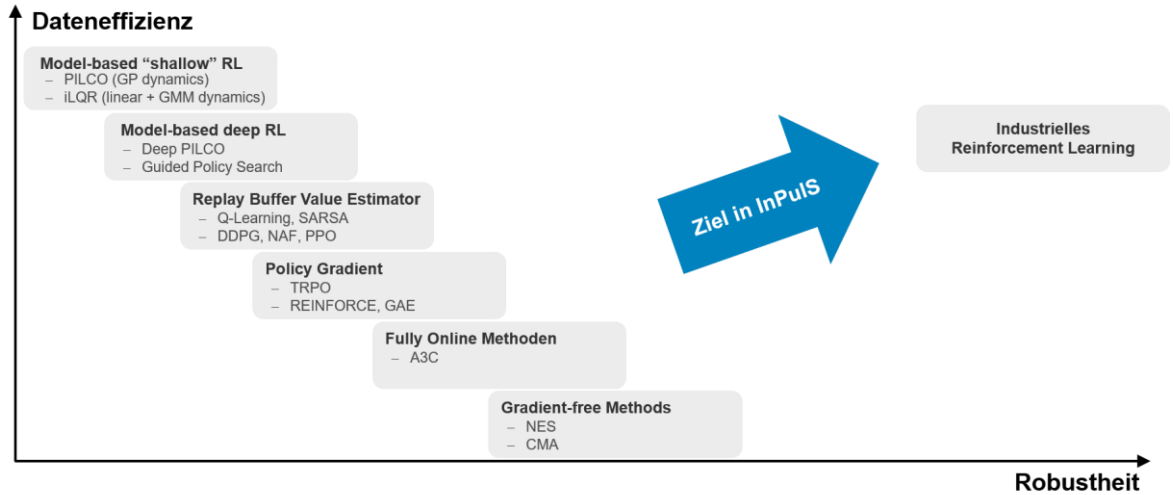

Abbildung 17: Einordnung der aktuellen Reinforcement Learning Methodiken hinsichtlich Dateneffizienz und Robustheit

Guided Policy Search Verfahren sind zwar sehr dateneffizient, erfüllen jedoch noch nicht die Anforderungen hinsichtlich Robustheit. Hierfür wurde ein neuer Ansatz basierend auf Guided Policy Search entwickelt, die **Generative Motor Reflexes** (**GMR**).

Erlernte Policies mit einer hohen Robustheit ermöglichen die Berücksichtigung von Modellierungsfehlern und unerwarteten Störungen. GPS ist nur robust in einer kleinen Region um die beobachteten Zustände herum, außerhalb dieser Region kann sich die globale Policy unberechenbar verhalten. GMR sind eine Erweiterung des ursprünglichen GPS Verfahrens. GMR erreicht eine erhöhte Robustheit durch Einführung einer robusteren Repräsentation der globalen Policy [39] (siehe ***Abbildung 19***). Hierbei verwendet GMR anstelle von Zustands-Aktions-Paaren direkt die linearen Controller der Trajektorienoptimierung und trainiert das neuronale Netz darauf, diese für die gegebenen Zustände zu reproduzieren.

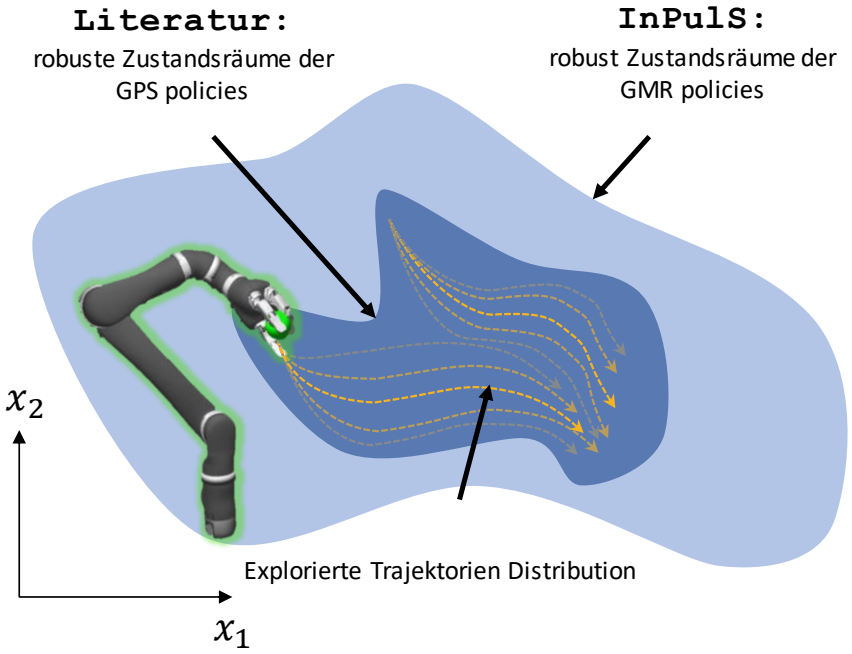

Abbildung 19: Vergleich des robusten Zustandsraumes von GPS Verfahren und den im Rahmen dieses Projektes entwickelten Ansatzes.

Um die generalisierenden Eigenschaften des neuronalen Netzes auch außerhalb des besuchten Zustandsraum zu erhalten, verwendet GMR einen Variational Autoencoder (VAE). Dieser VAE reduziert die Zustände auf einen latenten Zustandsraum und rekonstruiert die linearen Controller von diesen latenten Zuständen. Da der VAE dafür sorgt, dass der latente Raum dicht ist, werden so auch unbekannte Zustände auf bekannte Controller abgebildet. ***Abbildung 20*** illustriert die Funktionsweise des GMR Modells.

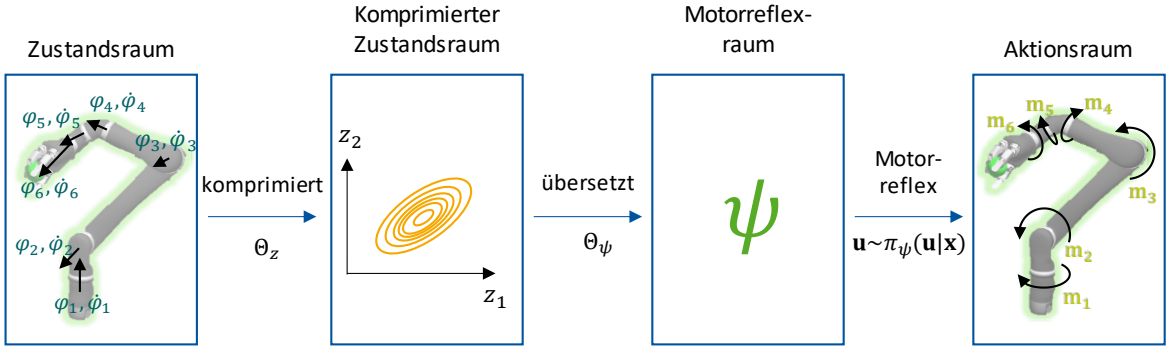

Abbildung 20: Motorreflexe zur Erhöhung der Robustheit

GMR funktioniert sehr gut bei global ähnlichen Dynamiken [4]. Bei abrupten Diskontinuitäten in der Systemdynamik steigt jedoch der Bedarf an Trainingsdaten stark an, weshalb GMR nicht für alle Anwendungsszenarien geeignet ist [40]. Daher wurde die **Self-Regulating Motor Unit (SRMU)** als eine alternative Policy-Repräsentation für diese Umgebungen entwickelt (vgl. ***Abbildung 21***). Diese basiert auf dem gleichen Prinzip wie GMR, indem das Modell einen Motorreflex statt einer direkten Aktion generiert. Der konstante Teil des Reflexes wir bei der SRMU hingegen nicht aus dem latenten Zustandsraum gewonnen, sondern direkt aus dem Zustand und wird zusätzlich darauf trainiert, numerische Fehler im Gradientenschätzer auszugleichen.

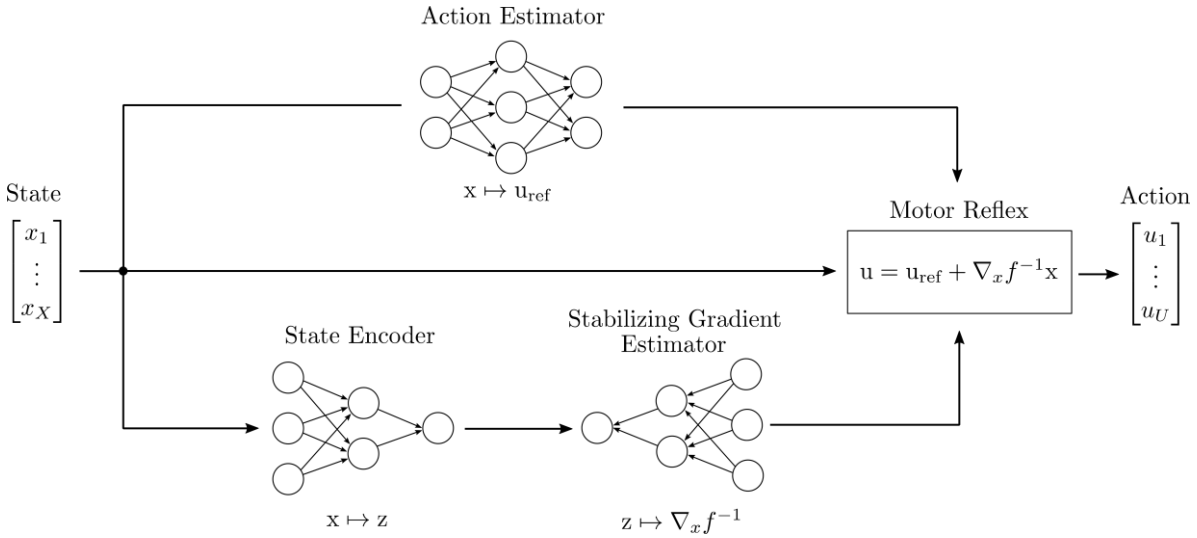

Abbildung 21: Modell der SRMU

Mit dem Abschluss von Arbeitspaket 2 wurde der zweite Meilenstein, Anwendbare Konzepte für Klassen von Produktionsprozessen, erreicht.

4.3 Arbeitspaket 3: Realisierung/Demonstrator

Zur Erprobung und Veranschaulichung der erforschten Modelle und Methoden eines intelligenten selbstlernenden Prozessregelkreises und Produktionssteuerung sollte parallel zur Konzeptionierung ein Forschungsdemonstrator realisiert werden. Dieser sollte eine reale robotergestützte Montage beinhalten, erweitert um einen virtuellen Demonstrator, mit dem ein ganzes Produktionssystem simuliert werden kann. In diesem Gesamtdemonstrator sollten dann sämtliche Aspekte abgebildet und intelligente und selbstlernende Prozessabläufe erprobt werden.

4.3.1 Arbeitspaket 3.1: Konstruktion, Fertigung und Aufbau des Forschungsdemonstrators

4.3.1.1 Ziel des Arbeitspakets

Ziel dieses Arbeitspakets war die Erweiterung einer am IfU existierenden Montagezelle. In dieser Montagezelle werden eingeschränkt modellierbare, kraftgeführte Montageprozesse von Leichtbaurobotern realisiert. Am Beispiel einer Getriebeboxmontage sollte gezeigt werden, wie sich trotz unsystematischer Störungen Montageprozesse realisieren lassen sowie in welcher Form erlangtes Erfahrungswissen in eine übergeordnete Montageablaufplanung integrierbar ist. Der Demonstrator besteht hierzu aus insgesamt zwei Leichtbaurobotern, Materialzu- und abführsystemen, sowie einem Kamerasystem zur Zustandserfassung. Dieser Demonstrator wird um ein Netz, bestehend aus zusätzlichen virtuellen, inselbasierten Montagezellen erweitert und an die Montagezelle angeschlossen. Auf dieser Basis sollte eine Betrachtung auf Makroebene zur Untersuchung der lernfähigen Ablaufplanung erfolgend. Damit wurde der Untersuchungsgegenstand auf kontinuierliche Zustands- und Aktionsräume (Montageprozess), sowie auf diskrete Zustands- und Aktionsräume festgelegt (Montageablaufplanung).

4.3.1.2 Durchgeführte Arbeiten

In diesem Arbeitspaket wurde der Forschungsdemonstrator für den Use Case A2 (autonomer Montageprozess) erweitert. In diesem Use Case wird ein Fügeprozess als eine repräsentative Komponente des Montageprozesses einer Getriebebox betrachtet.

Die Montagezelle verwendet den JACO Roboterarm der Firma Kinova. Der JACO ist ein leichtgewichtiger Roboterarm mit 6 Freiheitsgraden, welcher positions- und kraftgeregelt werden kann. In den folgenden Aufgaben wird der JACO im *torque control* Modus betrieben. Die Zelle wurde mit drei Kameras ausgestattet, wobei zwei fest installiert (von der Seite und von oben) sind und die dritte sich mit dem Endeffektor bewegt (siehe ***Abbildung 22***). Dies ermöglicht die Erweiterung des Zustandsraums mit Bilddaten. Als Kommunikationsschnittstelle zwischen dem JACO Roboter und dem Reinforcement Learning Modul wurde das Robot Operating System (ROS) gewählt. Zusätzlich zu der physischen Montagezelle wurde eine Simulation erstellt. Es wurde ein kinematisches und visuelles Modell des Kinova JACO in einer Simulationsumgebung implementiert (sieht ***Abbildung 23***). Als Physiksimulation für diese virtuelle Montagezelle dient Gazebo, eine Open Source Physiksimulation, welche eine graphische Darstellung des Simulationsszenarios ermöglicht. Die Ansteuerung der Simulation wurde ebenfalls über ROS gelöst, wodurch ein einfaches Wechseln zwischen der Simulation und dem physischen Roboter ermöglicht wurde.

In dieser Umgebung wurden mehrere Aufgabensets definiert, welche für die einzelnen Schritte der Montage einer Getriebebox relevant sind.

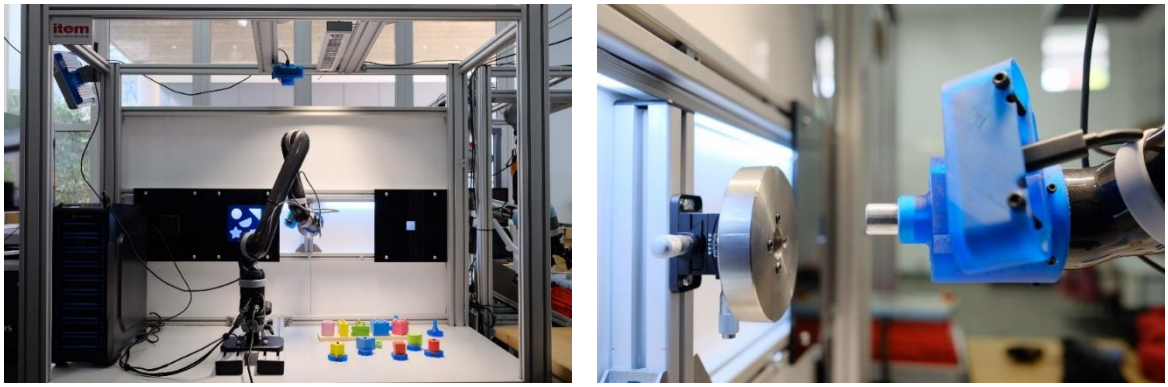

Abbildung 22: Forschungsdemonstrator

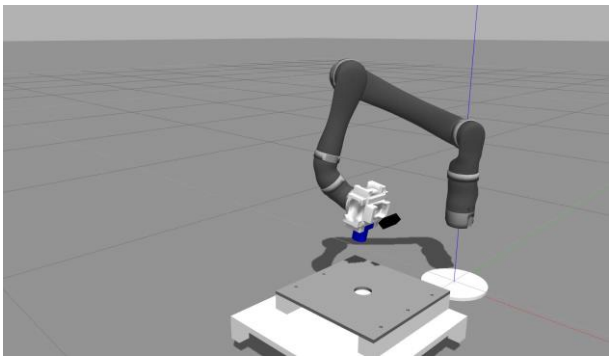

Abbildung 23: Virtuelle Montagezelle

4.3.1.3 Erzielte Ergebnisse

Die Montagezelle wurde fertiggestellt und ist einsatzbereit. Es stehen zwei Aufgabensets für den Fügeprozess zur Verfügung. Das erste Aufgabenset besteht aus mehreren unterschiedlich geformten Bauklötzen, welche auf den Endeffektor montiert werden können. Dieses Aufgabenset ermöglicht eine Vielzahl von verschiedenen Testszenarien, da die unterschiedlichen Geometrien und Toleranzen der Bauklötze unterschiedliche Montagebewegungen erfordern. Die Toleranzen erfordern eine Positioniergenauigkeit von 0,9 bis 0,1mm. Das zweite Aufgabenset besteht aus einer Reihe von Stiften mit unterschiedlichen Toleranzen, welche in ein Loch gefügt werden (f8 in H7 bis h6 in H7). Dieses Aufgabenset ermöglicht die präzise Evaluation der Positioniergenauigkeit des Montageroboters.

***Tabelle* 3** zeigt die genauen Toleranzen der einzelnen Aufgaben. Die beschriebenen Aufgaben stehen in dieser Umgebung und der Simulation zur Verfügung.

Aufgabenset „einfach“:

Loch	20	20	20	20	20	20	20	20	20
Stift	19,9	19,	19,7	19,6	19,5	19,4	19,3	19,2	19,1

Aufgabeset „schwierig“:

Loch	20H7	20H7	20H7	20H7
Stift	20h6	20g6	20f7	20f8

Tabelle 3: Aufgabensets für die Stift-in-Loch-Aufgabe

Die Simulationsumgebung erleichtert die Entwicklung von Lernverfahren durch schnelleres, ortsungebundenes Testen. So können in einer Simulationsumgebung neue Verfahren schnell getestet werden, ohne direkte menschliche Überwachung. Außerdem ermöglicht die Entwicklung in einer Simulationsumgebung ein paralleles Testen verschiedener Ansätze oder Parametersätze.

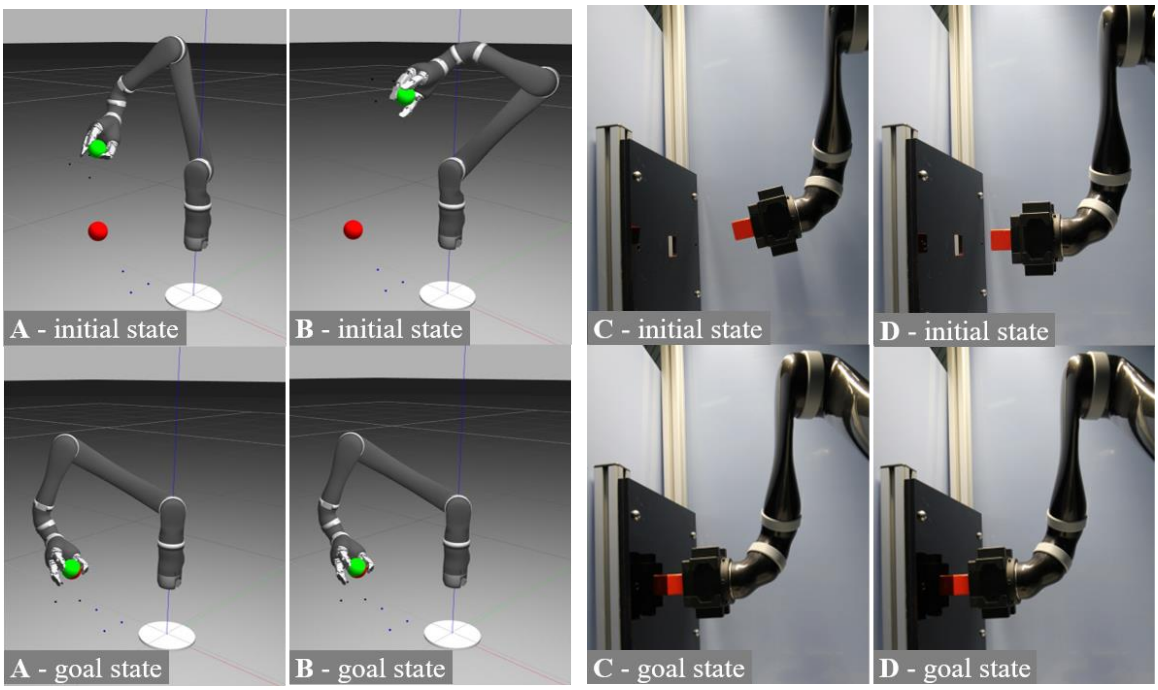

Abbildung 24: Verschiedene Aufgaben in der Simulation und am JACO.

4.3.1 Arbeitspaket 3.2: Implementierung der einzelnen Systemkomponenten

4.3.1.1 Ziel des Arbeitspakets

Ziel des Arbeitspakets 3.2 war die Integration der zuvor entwickelten Komponenten in den Demonstrator. Dazu gehört die Implementierung der Algorithmen zur selbstlernenden Prozessregelung (AP2.3) auf Basis der in AP 2.1 erstellten Systemarchitektur und den in AP 2.2. erstellten Zielsystemen.

4.3.1.2 Durchgeführte Arbeiten

In diesem AP wurden die in AP2.3 entwickelten Algorithmen in eine RL-Toolbox implementiert. Diese Toolbox ist eine institutseigene Weiterentwickelung der Guided Policy Search Implementierung von Chelsea Finn [41]. Die Toolbox basiert auf Python 3.6 und verwendet Machine Learning-typische Standard-Bibliotheken wie numpy, scipy, scikit-learn and matplotlib und Tensorflow. Auf Basis dieser Toolbox wurde zunächst eine klassenbasierte Softwarearchitektur entwickelt, welche in der Lage ist, die verschiedenen in AP 2.1 entwickelten Systemarchitekturen abzubilden. Die Idee von objektorientierter Programmierung ist die Schaffung einer Oberklasse, welche als ein softwaretypischer Prototyp funktioniert. Auf Basis der Oberklassen wurden die speziellen Implementationen der einzelnen Algorithmen und Use Cases entwickelt. Die Codebasis wurde getestet und fungiert als Grundlage für die Validierung der Anwendungsfälle in Arbeitspaket 4. Zuletzt wurde eine Dokumentation des Programcodes mithilfe der Open-Source Dokumentationssoftware Sphinx angefertigt.

4.3.1.3 Erzielte Ergebnisse

Ergebnis dieses Arbeitspakets ist eine Toolbox, welche die in AP 2.4 entwickelten Algorithmen und Schnittstellen zu der Gazebo Physiksimulation, dem Kinova JACO Roboter und der Anlagensteuerung für den pneumatischen Schüttgutförderer enthält. Die Toolbox gliedert sich grob in einen Algorithmus, welcher die Trainingsschritte spezifiziert, einen Agenten, welcher die Schnittstelle zur Umgebung darstellt, eine Kostenfunktion, welche es zu minimieren gilt, sowie eine Policy (vgl. ***Abbildung 25***). Eine detaillierte Dokumentation der einzelnen Software-Module wird separat veröffentlicht. Die Implementierung der neuronalen Netze in Tensorflow erlaubt das Auslagern eines großen Teils der notwendigen Berechnungen auf eine leistungsfähige GPU.

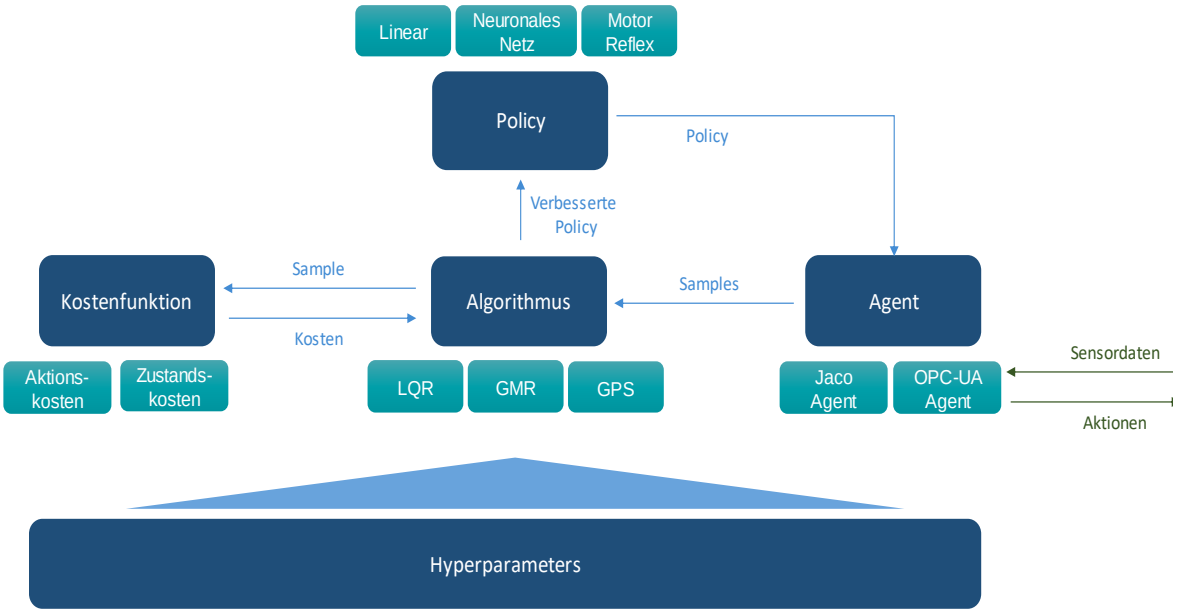

Abbildung 25: Übersicht über das Software-System

Ein typischer Ablauf einer Trainingsiteration ist in ***Abbildung 26*** dargestellt und verläuft wie folgt: Ausgehend von der aktuellen Policy werden Samples genommen. Bei einem Policy-Sample wird die gegebene Policy für eine Anzahl an Zeitschritten ausgeführt. Dabei wird zu jedem Zeitschritt der aktuelle Zustand gemessen. Die Policy bestimmt dann in Abhängigkeit zu dem aktuellen Zustand `x` eine Aktion `u`. Diese Aktion wird dann an die Aktuatoren weitergegeben. Anschließend wird auf den nächsten Zeitschritt gewartet, damit die Umgebung Zeit hat, auf die neue Aktion zu reagieren. Nachdem mehrere Samples gesammelt wurden, werden diese an den Algorithmus weitergegeben. Dieser evaluiert zunächst die Kostenfunktion und bestimmt eine Taylor-Approximation 2. Ordnung (`Cm`, `cv` und `cc`), welche von der Trajektorienoptimierung benötigt wird.

Als nächstes wird ein Modell der lokalen Dynamiken erstellt. Dieses verwendet ein Gaussian Mixture Model als Prior um ein Overfitting der anschließenden linearen Regression zu vermeiden. Dabei entstehen die Parameter `Fm`, `fv`, und `dyn_covar` eines linearen Dynamikmodells mit Gaußscher Unsicherheit. Im nächsten Schritt wird ausgehend von diesem Dynamikmodell und der Approximation optimale lineare Controller berechnet. Dies geschieht mittels eines iterativen linear-quadratisch-Gaußschen Regulators. Dieser alterniert zwischen einem *backward pass*, welcher die Controller optimiert, und einem *forward pass*, welcher die neue Trajektorienverteilung bestimmt. Diese beiden Schritte werden bis zur Konvergenz wiederholt. Das Ergebnis der Trajektorienoptimierung sind lineare Controller mit den Parametern `K`, `k` und `pol_covar`. Zusammen mit den Zuständen der Policy-Samples werden so die optimalen Aktionen bestimmt. Die linearen Controller stellen eine zeitabhängige Policy dar. Die Policyoptimierung erhält die Zustände und die zugehörigen optimalen Controller als Eingabe. Anschließend wird ein neuronales Netz darauf trainiert, das Verhalten dieser Controller in der Nähe der

Zustände zu imitieren. Das Ergebnis ist eine zeitunabhängige Policy, welche für den produktiven Einsatz, oder für eine weitere Iteration der Policy-Suche verwendet werden kann. Die nächste Iteration startet dann wieder mit dem Sammeln von Samples der neuen Policy.

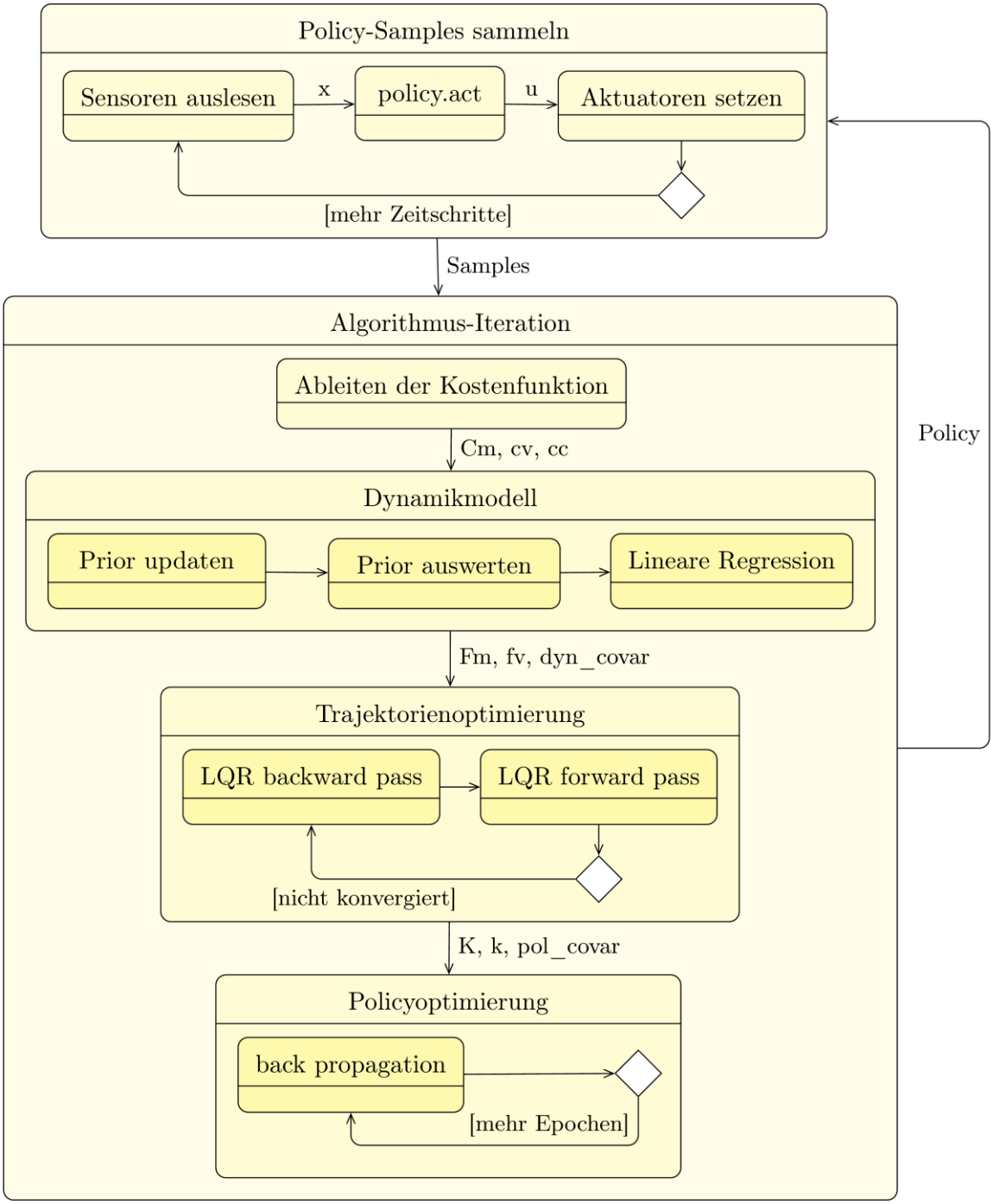

Abbildung 26: Trainingsiteration

Um die Entwickelung ähnlicher Projekte in Zukunft zu beschleunigen, wird die Codebasis zur weiteren Verwendung öffentlich gestellt[1] und auf Themis, der Kommunikationsplattform des VMDA, bereitgestellt.

Mit Abschluss des Arbeitspakets 3 (abgeschlossener Forschungsdemonstrator) und der Entwicklung einer 1. Version des in Arbeitspaket 5 beschriebenen Handlungsleitfadens wurde der dritte Meilenstein des Projekts erreicht.

[1] https://github.com/Cybernetics-Lab-Aachen/InPulS

4.4 Arbeitspaket 4: Validierung/Anwendungsfälle

Aufbauend auf der Konzeptionierung sowie der Umsetzung der Forschungsergebnisse im Demonstrator erfolgte in diesem Arbeitspaket die Validierung des Gesamtsystems auf verschiedenen Ebenen. Zum einen fand eine technische Evaluation von Teilkomponenten sowie des Gesamtsystems in zuvor definierten Testfällen statt. Die Umsetzbarkeit der Ergebnisse sowie die Anwendbarkeit des Handlungsleitfadens wurden in industriellen Anwendungen der Prozessindustrie und der diskreten Produktion und Logistik validiert und wirtschaftlich bewertet.

4.4.1 Arbeitspaket 4.1: Evaluierung der einzelnen Systemkomponenten sowie des Gesamtsystems

4.4.1.1 Ziele des Arbeitspakets

Das Ziel von AP4.1 war die Validierung des Gesamtsystems. Dazu wurden Testfälle aufgestellt, mit denen die einzelnen Systemkomponenten sowie das Gesamtsystem des Demonstrators auf Funktionsfähigkeit hin überprüft wurden. In weiteren Testfällen wurden potentielle Grenzen der entwickelten Algorithmen aufgezeigt. Solche betreffen unter anderem: Stabilität, Robustheit, Transparenz (Lesbarkeit durch Menschen), Konvergenzgeschwindigkeit des Lernverfahrens und das erreichbare optimale Systemverhalten. Die Testfälle wurden anschließend im Rahmen von mehreren Versuchsreihen im Forschungsdemonstrator überprüft.

4.4.1.2 Durchgeführte Arbeiten

In AP4.1 wurde vor Inbetriebnahme an dem physischen Forschungsdemonstrator zunächst die Implementierung des Lernverfahrens aus AP2.3 in zwei Simulationen validiert. Dazu wurde eine zusätzliche Simulationsumgebung ausgesucht und eine entsprechende Schnittstelle zu dieser Umgebung entwickelt.

Die zusätzliche Simulation ist ein Reaching Task mit einem Fetch Mobile Manipulator [42]. Der Fetch hat eine bewegliche Plattform und einen Arm mit sieben Freiheitsgraden. In diesem Szenario soll der Greifarm mit seinem Endeffektor eine vorgegebene Position erreichen. Beispielhafte Start- und Zielkonfigurationen sind in ***Abbildung 27*** dargestellt. Zustand- und Aktionsräume sind kontinuierlich. Diese Aufgabe weist eine geringe Komplexität auf und ist damit gut geeignet, um das generelle Lernverhalten und die Konvergenzgeschwindigkeit zu vergleichen. Nichts destotrotz ist der Zustandsraum dieser Simulationsaufgabe groß genug, dass nicht modellbasierte RL-Verfahren eine erhebliche Anzahl an Trainingsiterationen benötigen. Als Physik-Simulation wird hier Mujoco verwendet. Mujoco ist ein proprietärer Simulator, der speziell für kontaktreiche Umgebungen ausgelegt ist. Zudem zeigte sich Mujoco bei diesen Aufgaben schneller als Gazebo, was die Testphase deutlich verkürzte.

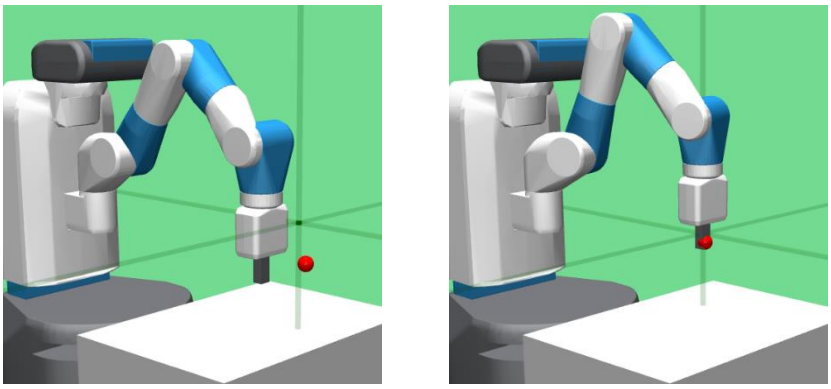

Abbildung 27: Der Fetch Mobile Manipulator. Links in Ausgans- und rechts in Zielposition.

In dieser Umgebung wurde die neu entwickelte SRMU mit drei verschiedenen Algorithmen verglichen, nämlich reiner Trajektorienoptimierung (LQR), Mirror Descent Guided Policy Search (MDGPS) [36] als Referenz, sowie Deep Deterministic Policy Gradient (DDPG) [43] als ein modellfreies Verfahren. Allen vier Algorithmen wird die gleiche Datenmenge zu Verfügung gestellt, nämlich jeweils fünf Trajektoriensamples von 1 bis 8 Initialzuständen. Hierbei

wurden drei unterschiedlichen Rahmenbedingungen untersucht. Zunächst wurde die reine Genauigkeit der gelernten Policy auf die statischen Trainingsziele ausgewertet. Anschließend wurden den gelernten Policies zufällige, beim Training nicht gesehene Ziele vorgegeben, um die Generalisierungsfähigkeit zu prüfen. Schließlich wurden die gelernten Policies von zufälligen Startzuständen aus gestartet, um die Robustheit gegen Auslenkung zu testen.

Das zweite Simulationsszenario ist die Gazebo-Simulation des Kinova Jaco. Hier wurden die gleichen Versuche wie am physischen Demonstrator ausgeführt. Dazu wurde eine Schlüssel-in-Loch-Aufgabe mit leichter Presspassung gewählt. In dieser Umgebung wurde während jeder Iteration fünf Trajektoriensamples zu je 80 Zeitschritten von jedem Initialzustand aus genommen. Über insgesamt zehn Iterationen wurden so insgesamt nur 50 Samples pro Initialzustand gesammelt. Ausgewertet wurden Versuche mit 1 und 2 Initialzuständen. Die Ergebnisse dieser Versuche wurden mit Referenzalgorithmen verglichen. Die gewonnenen Erkenntnisse wurden auf der *IEEE International Conference on Robotics and Automation* (ICRA) in Montreal veröffentlicht [39].

Zur Visualisierung und zum Verständnis des Trainingsprozesses wurde eine Reihe von Grafiken entwickelt, welche beim Training automatisch generiert werden. Die Darstellungen der verschiedenen Modelle, welche beim Trainingsprozess entstehen, ermöglichen eine Diagnose des Lernprozesses und vereinfachen die Wahl der Hyperparameter.

Das Anwendungsszenario A1 wurde als ein simulierter Maschinenverbund ebenfalls implementiert und in diesem Arbeitspaket validiert (vgl. ***Abbildung 28***). Es wurde ein Schwarm von sechs Maschinen mit unbekannten, stochastischen Eigenschaften, Vorbereitungsdauer, Ausführungsdauer, Qualität der ausgeführten Arbeit, Transportzeit, optimiert.

Ein Schritt besteht aus der Zuweisung einer Aufgabe an eine Maschine. Nach jeweils fünf Trainingsiterationen wird eine Belohnung aufgrund der in diesen Iterationen geleisteten Arbeit

Abbildung 28: Vergleich der Vorhersagen des DQN-Ansatzes mit den tatsächlichen Soll-Werten. Die beiden Umrandeten Fälle sind Fehlentscheidungen des DQN.

und der Qualität dieser Arbeit berechnet. Das Ziel des DQN ist es, die Arbeiten so auf die Maschinen zu verteilen, dass die Belohnung maximiert wird.

Abbildung 28: Vergleich der Vorhersagen des DQN-Ansatzes mit den tatsächlichen Soll-Werten. Die beiden Umrandeten Fälle sind Fehlentscheidungen des DQN.Abbildung 28 zeigt exemplarisch Vorhersagen des trainierten DQN. Zu jedem Schritt stehen verschiedene Anlagen zu Verfügung, welche die Aufgabe übernehmen können. Das Soll ergibt sich aus der analytischen Lösung der dem DQN-Algorithmus unbekannten Verteilung. Die Vorhersage sind die normierten Q-Werte für die Eignung der jeweiligen Maschine für die aktuelle Aufgabe. Zur Inferenzzeit verwendet der DQN eine Greedy-Policy, wählt also in jedem Schritt die Maschine mit dem höchsten Q-Wert. In den meisten Fällen stimmen die Vorhersagen und Entscheidungen des DQM gut mit dem tatsächlichen Optimum überein. Jedoch trifft das DQN recht häufig Entscheidungen, welche stark von der optimalen Lösung abweichen.

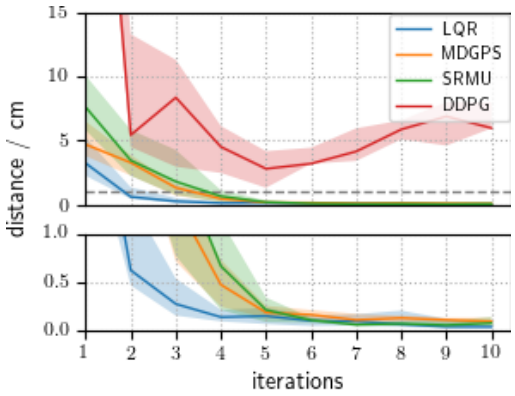

Abbildung 29: Ergebnisse der FetchReach-Experimente mit statischen Zielen.

4.4.1.3 Erzielte Ergebnisse

Die Auswertung der Experimentaldaten des Montageprozesses bestätigen die Effizienz, Robustheit und Generalisierungsfähigkeit der in AP2.3 entwickelten Methoden.

Genauigkeit: Wie **Abbildung 29** zeigt, konvergieren alle drei trajektorienzentrierten Verfahren schnell bei geringer bleibender Abweichung vom Ziel. Insbesondere konvergiert SRMU ähnlich schnell wir MDGPS. Die erreichte Genauigkeit ist etwas besser als die von MDGPS, besonders da MDGPS dazu neigt im Verlauf des Trainings instabil zu werden. Die reine Trajektorienoptimierung (LQR) konvergiert am schnellsten, stellt jedoch nur eine lokale und keine globale Policy dar und ist hier daher nur zu vergleichszwecken aufgeführt. DDPG konvergiert mit dieser Datenmenge überhaupt nicht, Versuche mit mehr Daten über mehr Iterationen (vgl. **Abbildung *30***) zeigen, dass durch die von DDPG benötige Datenmenge, sowie die starke bleibende Abweichung, eine Anwendung von DDPG im Industriekontext nicht realistisch ist. Die Konvergenzrate von DDPG zeigte sich nicht nur wesentlich langsamer als die der modellbehafteten Verfahren, sondern auch die bleibende Abweichung ist um Größenordnungen höher. Damit die anderen Algorithmen in dieser Darstellung noch erkennbar bleiben, wurde DDPG die zehnfache Datenmenge zur Verfügung gestellt (50 Samples pro Initialzustand). Dies bestätigt die in AP1.2 getroffene Wahl eines modellbasierten Algorithmus.

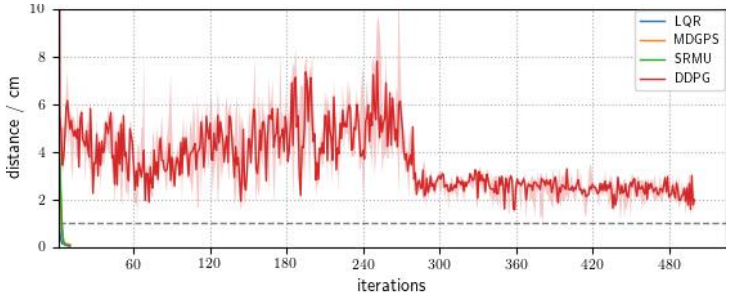

Abbildung 30: Konvergenzrate DDPG

Generalisierung: Bei den Versuchen mit zufälligen Zielen zeigt SRMU deutlich bessere Generalisierungseigenschaften als MDGPS (vgl. ***Abbildung 31***). Nicht nur ist die bleibende Abweichung ist geringer, sondern die Varianz zwischen den einzelnen Trajektorien ist deutlich kleiner. Die Konvergenzrate der SRMU ist initial langsamer, was aufgrund des komplexeren Modells jedoch nicht überraschend ist. Reine Trajektorienoptimierung (LQR) konvergiert unter diesen Bedingungen nicht, da die LQR-Policies nur lokal valide sind und nicht auf unterschiedliche Ziele generalisieren können. DDPG zeigt die gleiche nicht-Konvergenz wie zuvor. Die SRMU ist also durchaus in der Lage, besser zu generalisieren als vorherige Verfahren.

Robustheit: Ein zufällig verteilter Startzustand in dieser Umgebung wirkt sich kaum auf die verschiedenen Verfahren aus (vgl. **Abbildung 32**) MDGPS und SRMU können die Störungen gut ausgleichen, allein der LQR zeigt eine erhöhte Varianz, konvergiert also nicht notwendigerweise immer. Dies zeigt, dass die modellbasierten Verfahren robust gegenüber Störungen sind. Eine Untersuchung der latenten Zustände (vgl. ***Abbildung 33***) im neuronalen Netz der SRMU zeigt eine ausgesprochene Strukturiertheit, was die Robustheit der SRMU bestätigt.

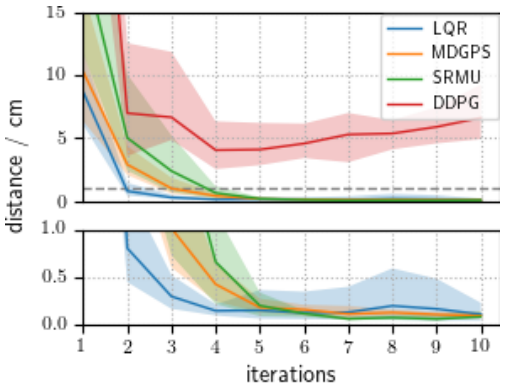

Abbildung 31: Ergebnisse der FetchReach-Experimente mit zufälligen Startzuständen.

Ebenfalls zeigt GMR sowohl in der Simulation als auch am Forschungsdemonstrator eine höhere Robustheit als MDGPS, wie **Tabelle 4** entnommen werden kann. Bei der schwierigeren Stift-in-Loch-Aufgabe weist GMR sogar eine deutlich höhere Erfolgsrate als MDPG auf und ist damit signifikant robuster.

Szenario	Trainingszustände	MDGPS	GMR
Jaco Reaching (Gazebo)	1	12/50	50/50
	2	50/50	50/50
Jaco Stift-in-Loch (Demonstrator)	1	4/50	39/50
	2	9/50	44/50

Tabelle 4: Vergleich der Erfolgsraten von MDGPS und GMR, jeweils nach 10 Trainingsiterationen für 50 zufällig verteilte Testzustände.

Die entwickelten Algorithmen erfüllen somit die eingangs spezifizierten Anforderungen für industrielles Reinforcement Learning in Bezug auf Dateneffizienz, Robustheit, Stabilität und Generalisierungsfähigkeit.

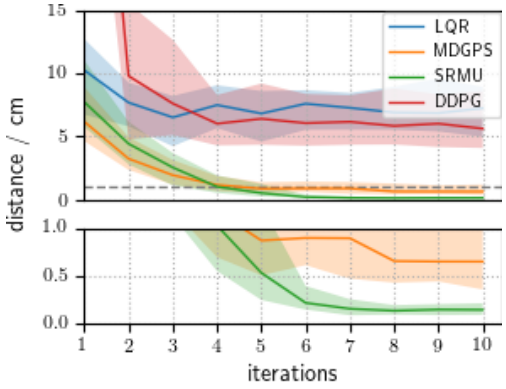

Abbildung 32: Ergebnisse der FetchReach-Experimente mit zufälligen Zielen.

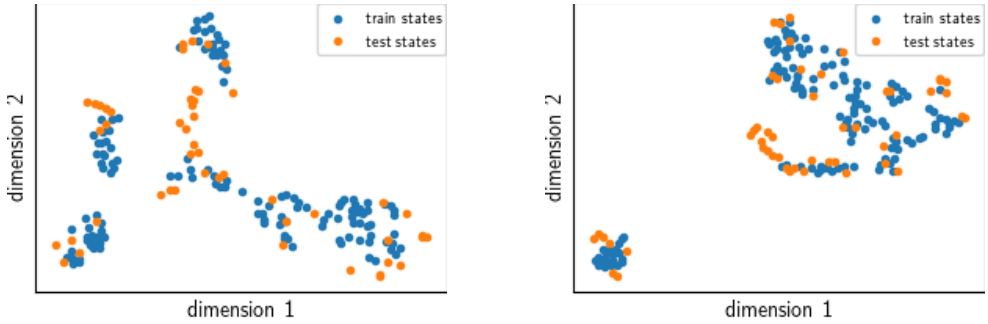

Abbildung 33: Visualisierung der latenten Zustandsräume einer austrainierten SRMU, eingebettet in zwei Dimensionen über t-SNE. Es zeigt sich eine deutliche Strukturierung des Raums und die Testzustände werden gut in die Trainingszustände eingebettet.

4.4.2 Arbeitspaket 4.2: Validierung anhand von industriellen Anwendungsbeispielen

4.4.2.1 Ziele des Arbeitspakets

Der Prozessindustriedemonstrator betrachtet die pneumatische Saugförderung von Schüttgütern. Schüttgüter weisen, bedingt durch Korngröße und -verteilung, Schüttgutdichte, Schüttwinkel, Feuchtigkeit, Temperatur und Reibungswiderstand unterschiedlichste Eigenschaften auf. Zudem zeigen, aufgrund des globalen Markts für die internationale Beschaffung von Roh- und Hilfsstoffen unterschiedlichster Herkunft, Produkte gleichen Namens, gleicher chemischer Formel und Körnung unterschiedliches Fließverhalten, wodurch Potentiale für unvorhersehbare Störungen erzeugt werden. Auf Grundlage der zuvor entwickelten Methodik wird dargestellt, wie der Schüttgutförderer eigenständig Qualitätsabweichungen, bis hin zu Störungen erkennt und auf Basis eines intelligenten, selbstlernenden Ansatzes aktiv kompensiert.

In diesem Arbeitspaket stand das Forschungsinstitut beratend zur Verfügung. Umgesetzt wurde der Demonstrator unter Zuhilfenahme eines Technologietransferkonzepts aus AP5.1 von Unternehmen aus dem PA.

Zudem sollte eine Erfassung und Bewertung der Wirtschaftlichkeit von intelligenten selbstlernenden Prozessregelkreisen erfolgen. Durch Workshops mit den Unternehmen des PAs wurde ein Abgleich der zu Beginn des Projektes aufgenommenen Anforderungen mit den realisierten Lösungen durchgeführt. Die einzelnen Arbeitsschritte wurden ausgewertet und reflektiert. Weiterhin sollte eine Ex-post Wirtschaftlichkeitsbewertung zur Erfassung der Wirtschaftlichkeit unter Einsatz der entwickelten Methodik mit Hilfe des NOWS-Verfahrens erfolgen.

4.4.2.2 Durchgeführte Arbeiten

Der Demonstrator wurde von der AZO GmbH & Co. KG nach den zuvor besprochenen Vorgaben mit der entsprechenden Sensorik und Kommunikationsinfrastruktur auf einer Testanlage aufgebaut.

Die Validierung des entwickelten Lernverfahren erfolgte in zwei Testwochen, im Oktober 2018 und im März 2019.

Im Vorfeld der Inbetriebnahme vor Ort wurde die Toolbox um eine OPC-UA Schnittstelle erweitert und die zuvor definierten Sensoren wurden implementiert.

Vor Ort wurden zunächst die tatsächlich vorhandenen und die zuvor implementieren Signalquellen abgeglichen und harmonisiert. Ein Versagen der ML-Steuerung wurde damit ausgeschlossen, dass durch einen separaten IPC die Einhaltung der Maschinengrenzen gewährleistet wird.

In den Versuchen wurden drei Algorithmen verglichen: LQR als Baseline, MDGPS als Referenz und die neu entwickelte SRMU. Während des Trainings wurden jeweils drei Trajektoriensamples zu je 90s pro Iteration genommen, sowie ein zusätzliches Kontrollsample, um die Güte der bisher gelernten Policy zu vergleichen. Mit den notwendigen Zeiten für den Nachlauf und den Reset der Anlage konnten so in circa einer halben Stunde vier Trainingsiterationen stattfinden. Nach dem Training wurden die gelernten Policies in verschiedenen Situationen getestet, darunter Generalisierung auf längere Förderdauer, Materialwechsel, und simulierter Sensorausfall.

Die verwendeten Produkte sind in **Abbildung 34** dargestellt. Das Training wurde stets mit dem Produkt S-PVC durchgeführt. Die Fördereigenschaften des Produkts sind während der Trainingsdurchläufen nicht fest, sondern ändern ihre Schüttdichte, können sich statisch aufladen und hängen vom Wetter (Luftdruck, Luftfeuchtigkeit) ab. Daher kann auch ein Produkt aus derselben Quelle unterschiedliche Eigenschaften ausweisen.

Abbildung 34: Die geförderten Produkte. Links S-PVC, ein feines Pulver mit einer Schüttdichte von 0,56kg/l und guten Fördereigenschaften. Rechs Senfmehl mit einer Schüttdichte von 0,39kg/l und bedingt durch die Klebrigkeit, schlechteren Fördereigenschaften.

Die Auswertung und Analyse der erhobenen Daten erfolgten schließlich im Nachgang.

4.4.2.3 Erzielte Ergebnisse

Die Auswertung der Versuche am Schüttgutförderer zeigt zunächst, dass Reinforcement Learning an einer solchen Anlage eingesetzt werden kann. ***Abbildung 35*** zeigt wie Reinforcement Learning in der Lage ist, die Fördermenge stetig zu verbessern. Hier konnte die Ausgangsfördermenge der initialen Policy von ~35kg regelmäßig auf bis zu 50kg erhöht werden. Hierbei zeigte insbesondere MDGPS oft sprunghafte Verbesserungen. Dabei waren die Verfahren in der Lage mit 12-18 Trajektoriensamples gutes Verhalten zu erlernen und sind daher als sehr dateneffizient zu bezeichnen. Der zeitliche Aufwand zum Antrainieren einer Policy betrug damit nur zwischen 30 und 60 Minuten, im Vergleich zu mehreren Tagen Simulationszeit mit modellfreien Reinforcement Learning Ansätzen.

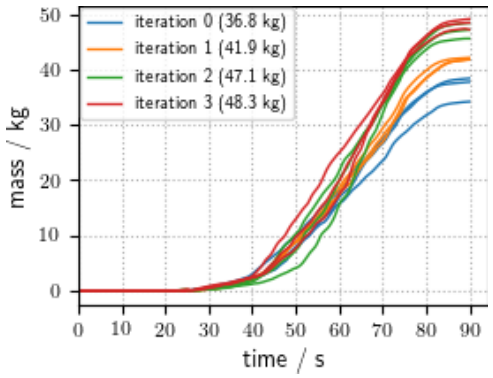

Abbildung 35: Zunahme der Fördermenge im Verlauf der Trainingsiterationen.

Zeitunabhängigkeit: Die gelernte Policy soll zeitunabhängig sein und aus dem Training kein zeitabhängiges Verhalten lernen. Deshalb wurden die gelernten Policies auch über längere Zeiträume getestet, nämlich 120s und 180s. ***Abbildung 36*** zeigt die Ergebnisse dieser Testreihen. Der LQR ist hier nur zum Vergleich angegeben, da die LQR-Policy inhärent zeitabhängig ist und daher nicht auf längere Zeiträume angewendet werden kann. Die MDGPS Policies

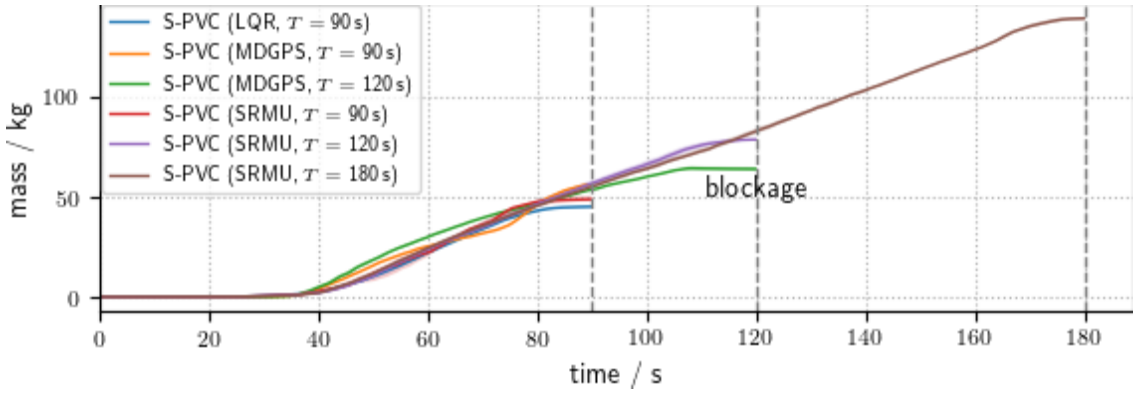

Abbildung 36: Vergleich verschiedener gelernter Policies über längere Zeiträume. Alle Policies wurden mit 90s trainiert. Bei Zeiträumen von 120s und mehr hat MDGPS unweigerlich Stopfer produziert.

konnten nicht auf längere Zeiträume generalisieren. Alle Versuche über 120s resultierten in einem Stopfer. Aufgrund des hohen zeitlichen Aufwands zur Stopferbeseitigung wurden diese nicht oft genug wiederholt, um eine Varianz darzustellen und fehlen daher in dem Diagramm. Die Tests mit 120s resultierten nicht immer in einem Stopfer, aufgrund der Länge des Nachlaufes kann davon ausgegangen werden, dass diese kurz davorstanden. Die SRMU hingegen war in der Lage, auch über länger Zeiträume effektiv zu Fördern und es wurde damit gezeigt, dass sie auch über längere Zeiträume generalisieren kann.

Materialwechsel: In einer weiteren Testreihe wurde die Robustheit gegenüber Materialwechsel geprüft. Dafür wurde in der Anlage statt des S-PVC Senfmehl gefördert. Senfmehl hat weniger gute Fördereigenschaften als S-PVC, da es klebrig ist. Diese Versuche wurden nur mit der SRMU durchgeführt, da sich die gelernten MDPGS Policies im Vorfeld bereits als weniger robust gezeigt hatten und ein Stopfer mit Senfmehl zu viel wertvolle Experimentalzeit verschwendet hätte. Die Ergebnisse dieser Testreihe sind in ***Abbildung 357*** dargestellt. Hierbei

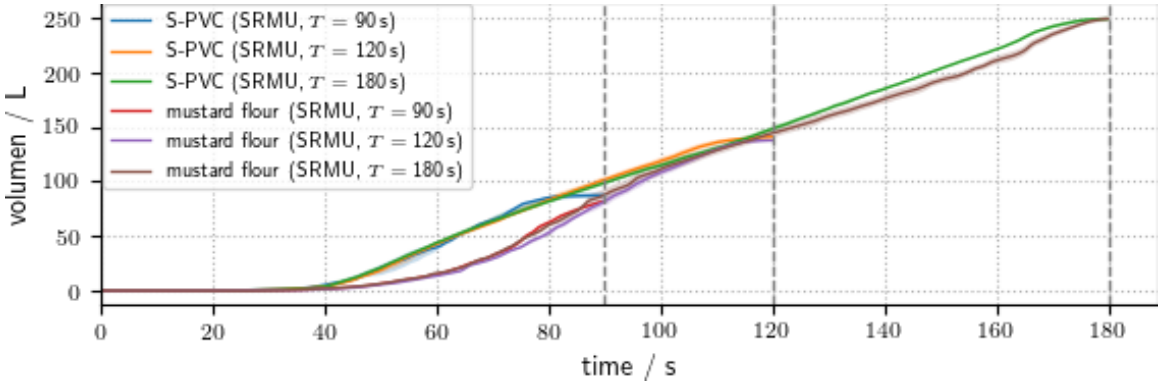

Abbildung 37: Vergleich des Förderverhaltens der gelernten Policy mit den Trainingsmaterial S-PCV und einem Testmaterial Senfmehl. Die SRMU erreicht auch mit einem unbekannten Material eine gleichmäßige Förderung.

wurden die Fördermengen zur besseren Vergleichbarkeit mit der jeweiligen Produktdichte normalisiert. Auch wenn sich das Einschwingverhalten zu dem bei S-PVC unterscheitet, erreicht die SRMU Policy auch mit einem anderen Material mit anderen Eigenschaften eine stabile Förderung.

Während der Projektlaufzeit wurde sich gegen eine Wirtschaftlichkeitsbewertung mithilfe des NOWS-Verfahrens entschieden, da dieses Verfahren die Einführungsrisiken nicht mit einbezieht und diese beim Einsatz von Reinforcement Learning Strategien nicht zu vernachlässigen sind. Die Validierung am Schüttgutförderer zeigt eine Erhöhung der Förderleistung um ca. 20%, abhängig von der initialen Förderstrategie. Daraus ergibt sich eine direkte Ersparnis von 20% der bisherigen Förderkosten. Diese ist allerdings schwierig abzuschätzen, da die Förderkosten sehr individuell sind und vom Produkt, der Förderstrategie sowie der exakten Ausprägung der Anlage abhängen. Die Wirtschaftlichkeit wurde in der direkten Anwendungsfall bei der AZO GmbH & Co. KG gezeigt.

4.5 Arbeitspaket 5: Dokumentation und Aufbereitung für KMU

Im Sinne des Technologietransfers sowie der projektbezogenen Dokumentation sollten im AP5 fortlaufend alle Forschungsergebnisse dokumentiert und veröffentlicht sowie weitere Transfermaßnahmen entwickelt und durchgeführt werden. Aufgrund dieser übergeordneten Zielsetzung handelt es sich um ein parallel laufendes Arbeitspaket.

Das AP5 ist in zwei Teilpakete gegliedert:

- AP5.1 Entwicklung und Umsetzung des Technologietransfers (durchgängig seit Projektbeginn)
- AP5.2 Erstellung von Zwischenberichten, Reports und des Abschlussberichts

Diese werden in Abschnitt 4.5.2 und 4.5.3 gesondert vorgestellt.

4.5.1 Theoretische Grundlagen

4.5.2 Arbeitspaket 5.1: Entwicklung und Umsetzung des Technologietransfers

4.5.2.1 Ziel des Arbeitspaketes

Ziel des Arbeitspaketes AP5.1 ist die Entwicklung und Umsetzung eines Technologietransferkonzepts zur Nutzbarmachung der Arbeitsergebnisse in der industriellen Praxis. Dazu wird der Entwicklungsprozess zur Gestaltung von intelligenten und selbstlernenden Systemen in Form eines Handlungsleitfadens allgemeinverständlich beschrieben. Außerdem wird eine Broschüre erstellt, in der die aufgebauten Demonstratoren sowie die erreichten Ergebnisse präsentiert werden. Zusätzlich wird ein Konzept für Schulungen von Mitarbeitern zum Umgang mit der vorgestellten Technologie erarbeitet. Weiterhin werden die erzielten Ergebnisse in Fachzeitschriften und auf Konferenzen veröffentlicht.

Zur Umsetzung dieses Arbeitspaketes werden zuvor herausgearbeitete Anforderungen und Konzepte benötigt, welche mit der Umsetzung der Aufgaben und Validierung der Ergebnisse abgeglichen werden können. Das Ergebnis ist ein Handlungsleitfaden zur Modellierung und Implementierung von intelligenten und selbstlernenden Produkten und Systemen in KMU, welcher die Grundlage für den Technologietransfer bildet. Die aufgebauten Demonstratoren dienen zusätzlich als Modell zur Veranschaulichung der Machbarkeit der Projektinhalte und zur Verdeutlichung möglicher Einsatzgebiete.

4.5.2.2 Durchgeführte Arbeiten

Im Rahmen des AP5.1 wurden während der gesamten Projektlaufzeit Schritte zur Vorbereitung des erfolgreichen Technologietransfers unternommen. Dabei fand ein kontinuierlicher Austausch mit den Industriepartnern statt. So konnten gemeinsam Use Cases erarbeitet werden und die gewonnenen Erkenntnisse direkt in den industriellen Kontext eingebracht werden. Strukturiert wurde diese Kooperation durch vier **Webinare**, vier **Treffen des Projektausschusses** und das **Abschlusstreffen** in Frankfurt am Main am 12. September 2019. Das Ergebnis der Kooperation ist ein **Handlungsleitfaden** für intelligente und selbstlernende Produktionsprozesse. Um eine zusätzliche Hilfestellung für KMU zu schaffen, wurde eine ausführliche **Dokumentation** zu der während des Projektes entwickelten **Software** erstellt. Um die Forschungserkenntnisse einer breiteren Öffentlichkeit zugänglich zu machen, wurde eine **Internetpräsenz** aufgebaut (http://www.i40-inpuls.de). Außerdem wurde zur Verbreitung erzielter Ergebnisse in die Fach- und Wissenschaftscommunity eine durch Peer-Review validierte **Veröffentlichung** unter dem Titel „Learning Robust Manipulation Skills with Guided Policy Search via Generative Motor Reflexes“ eingereicht [39]. Außerdem wurde im Nachgang des

Projekts eine Dissertation zum Thema industrielles Reinforcement Learning verfasst [43]. In diesem Rahmen wurden am IfU **Demonstratoren** aufgebaut, welche die Einsatzmöglichkeiten der Forschungsergebnisse veranschaulichen. Im Folgenden werden die einzelnen Schritte genauer erläutert.

Der direkte Austausch mit den 19 beteiligten Partnern aus der Industrie wurde über virtuelle Konferenzen und Präsenztreffen erreicht. Während der ersten beiden Webinare wurde eine Einführung in die Ziele des Projektes gegeben und Grundlagen aus dem Bereich des Machine Learning vermittelt. Außerdem konnte hier erstes Feedback zu dem bisherigen Projektverlauf und ersten Ergebnissen gesammelt werden. Das folgende Webinar war auf die Vertiefung der für das Projekt relevanten Kenntnisse im Bereich des Reinforcement Learning ausgerichtet. Im letzten Webinar wurden konkrete Use Cases diskutiert. Insgesamt wurden durch dieses Format projektspezifische Kenntnisse vermittelt, aber auch die Einbindung und der Bedarf der Industrie während des gesamten Projektverlaufes sichergestellt.

Während der Treffen des projektbegleitenden Ausschusses, welcher sich aus Experten der Industrie zusammensetzt, wurden zunächst Anforderungen aus der Industrie erarbeitet. Um diese mit den Forschungszielen zusammenzubringen, wurden verschiedene Lernverfahren vorgestellt. So konnten in speziellen Projekttreffen Use Cases für einen Schüttgutförderer erarbeitet werden. In den weiteren Treffen des projektbegleitenden Ausschusses wurden der Projektverlauf und vorläufige Ergebnisse sowie das weitere Vorgehen diskutiert und die Anforderungen aktualisiert. Die Abschlussveranstaltung fand im Rahmen des vom VDMA ausgerichteten Informationstag *Intelligente Produktionsprozesse – Forschung zu Machine Learning und Künstlicher Intelligenz* statt. Das Projekt wurde dort anhand verschiedener Vorträge vor rund 150 VDMA Mitgliedern vorgestellt. Vertreterinnen des IfU gaben einen Einblick, wie die künstliche Intelligenz die zukünftige Produktion transformieren wird. Außerdem stellten sie die Erfahrungen und Erkenntnisse des Projekts anhand eines Fachvortrages zum Thema industrielles Reinforcement Learning vor. Zum Abschluss der Projektvorstellung schilderte ein Vertreter der AZO GmbH & Co. KG seine Erfahrungen im Anwendungsfall des Schüttgutförderers. Somit konnte den Besuchern des Informationstages ein umfassendes Bild des Projektes sowohl aus Forschungssicht als auch aus der Perspektive der Industrie gegeben werden.

Die Forschungsergebnisse, das Feedback aus den Webinaren und Projekttreffen, die Ergebnisse aus den Use Cases und aus dem Aufbau der Demonstratoren am IfU wurden in einem Handlungsleitfaden für KMU zusammengefasst. Dieser unterstützt die Betriebe bei der Entwicklung einer eigenen Strategie zur Einführung von Reinforcement Learning für industrielle Anwendungen. So sollen KMU befähigt werden, Potentiale zu erkennen und die nötigen Rahmenbedingungen zu schaffen. Über einen Werkzeugkasten kann diese Einführung erleichtert werden.

Als weiterer Bestandteil des Technologietransfers wird auf der Internetseite des Cybernetics Labs IMA & IfU sowie auf einer eigens eingerichteten Homepage über InPuls informiert. Die Informationen, welche über diese Kanäle vermittelt werden, sind für ein breites Publikum aufbereitet. Für ein spezifischeres Fachpublikum sind Forschungserkenntnisse aus dem Projekt in der oben genannten Veröffentlichung zugänglich.

Durch die genannten Maßnahmen wurde ein Technologietransfer auf mehreren Ebenen umgesetzt. Unterschiedliche Zielgruppen wurden über verschiedene Kanäle über den Projektverlauf und die Ergebnisse auf dem Laufenden gehalten. Besonders hervorzuheben ist der entstandene Handlungsleitfaden, welcher die Anwendung der Forschungsergebnisse in der Industrie erleichtert. Die in diesem Leitfaden zusammengefassten Ergebnisse des Projektes werden im folgenden Abschnitt näher erläutert.

4.5.2.3 Ergebnisse

Die veröffentlichten Ergebnisse des Projektes umfassen sowohl Forschungsergebnisse als auch Erfahrungen aus der Anwendung in der Industrie. Zugänglich gemacht wurden diese in dem veröffentlichten Handlungsleitfaden in einer Print- und einer Online-Version sowie in einem veröffentlichten Paper. Die Ergebnisse der Webinare sind in Protokollen festgehalten. Die Inhalte der genannten Quellen tragen zu einer weit gefächerten Verbreitung der Erkenntnisse in der Industrie bei und werden im Folgenden weiter ausgeführt.

Handlungsleitfaden

Der Handlungsleitfaden zu intelligenten und selbstlernenden Produktionssystemen in der industriellen Anwendung ist eine Zusammenführung von Forschungsergebnissen und Erfahrungen aus dem Aufbau von Demonstratoren und industriellen Anwendungsbeispielen. Als Hilfestellung zur Einführung von Reinforcement Learning im industriellen Kontext bietet er eine Orientierung durch Leitfragen und einen Werkzeugkasten für das Vorgehen bei der Integration dieser neuartigen Produktionsprozesse. Der Aufbau lässt sich dabei in drei große Bereiche unterteilen: Die Einführung in die Möglichkeiten des Einsatzes maschinellen Lernens in der Produktion, das Vorgehen bei der Einführung dieser Prozesse in industrielle Anwendungen und die Vorstellung von Anwendungsbeispielen. Mit diesen Bereichen wird die Zielsetzung des Leitfadens (siehe ***Abbildung 38***) realisiert.

Abbildung 38: Zielsetzung des Handlungsleitfadens

Nach der theoretischen Einführung in maschinelles Lernen wird der Bereich des Reinforcement Learning vertieft. Dabei wird auf die besonderen Anforderungen industrieller Anwendungen an diese Technologie eingegangen. Dazu gehören unter anderem das Trainieren auf Grundlage weniger Daten und die Robustheit gegenüber Unsicherheiten (siehe ***Abbildung 39***). Die hier vorgestellten Anforderungen sind das Ergebnis aus der kontinuierlichen Zusammenarbeit mit Industriepartnern und somit direkt auf KMU anwendbar. Dieser Abschnitt des Leitfadens ermöglicht Lesern, Potentiale für ihr Unternehmen zu erkennen.

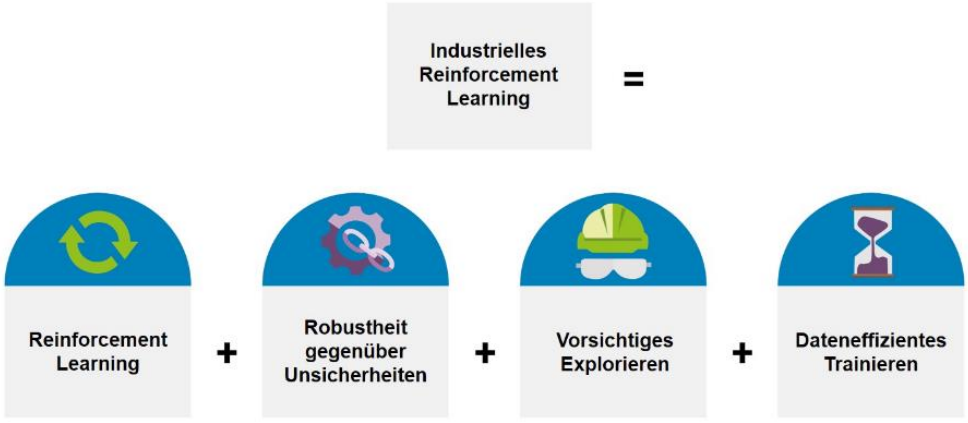

Abbildung 39*:* Besondere Anforderungen des industriellen Reinforcement Learning

Anschließend stellt der Leitfaden Werkzeuge zur Entwicklung einer Einführungsstrategie der zuvor beschriebenen Technologien vor. Dazu werden zunächst Leitfragen formuliert, welche bei der Findung eines geeigneten Pilotprojektes helfen sollen. Dieses wird empfohlen, um die sehr komplexe Einführung neuer Technologien zu erleichtern. Dabei werden unterschiedliche Bereiche hinterfragt: personelle und materielle Ressourcen sowie die Charakterisierung des vorliegenden Systems (Prozessanalyse) (vgl. ***Abbildung 40***).

Abbildung 40: Übersicht Leitfragen: bezüglich der Prozessanalyse (grün), der personellen Ressourcen (grau) und der materiellen Ressource (lila)

Zur Beantwortung dieser Leitfragen wird als Hilfestellung ein Werkzeugkasten eingeführt. Dabei gibt es einen Kasten für die Prozessanalyse und einen für die Beschreibung der personellen Ressourcen. Im ersten Schritt kann mit dem Werkzeugkasten für die Prozessanalyse ermittelt werden, ob ein Prozess sich für den Einsatz von Reinforcement Learning eignet bzw. welche Anforderungen eventuell ergänzt werden müssen. Anschließend wird ein Überblick über die benötigten Kompetenzen und entsprechenden personellen Ressourcen gegeben. Abschließend werden Hilfestellungen zur Wahl der Hardware gegeben.

Nachdem so die prozessseitigen, personellen und materiellen Anforderungen ermittelt wurden, gibt der Handlungsleitfaden einen Fahrplan für die Einführung eines Reinforcement-Learning-Prozesses vor. Das vorgestellte Schema gliedert sich in eine Planungs- und eine Realisierungsphase (siehe **Abbildung 41**) und nimmt auch eine personelle Zuordnung zu einzelnen Phasen vor.

Um den ersten Schritt, die Identifikation eines Pilotprojektes zu erleichtern, führt der Leitfaden zwei Anwendungsbeispiele industriellen Reinforcement Learning auf: einen autonomen Montageprozess und einen selbstlernenden Prozess auf einem Schüttgutförderer. Dabei wird zunächst das Szenario vorgestellt, wobei auf besondere Anforderungen hingewiesen wird (z.B. Behaftung des Prozesses mit Unsicherheiten). Anschließend wird das ermittelte Setting für das Reinforcement Learning beschrieben. Zum Schluss werden der Lernprozess und die erzielten Ergebnisse vorgestellt.

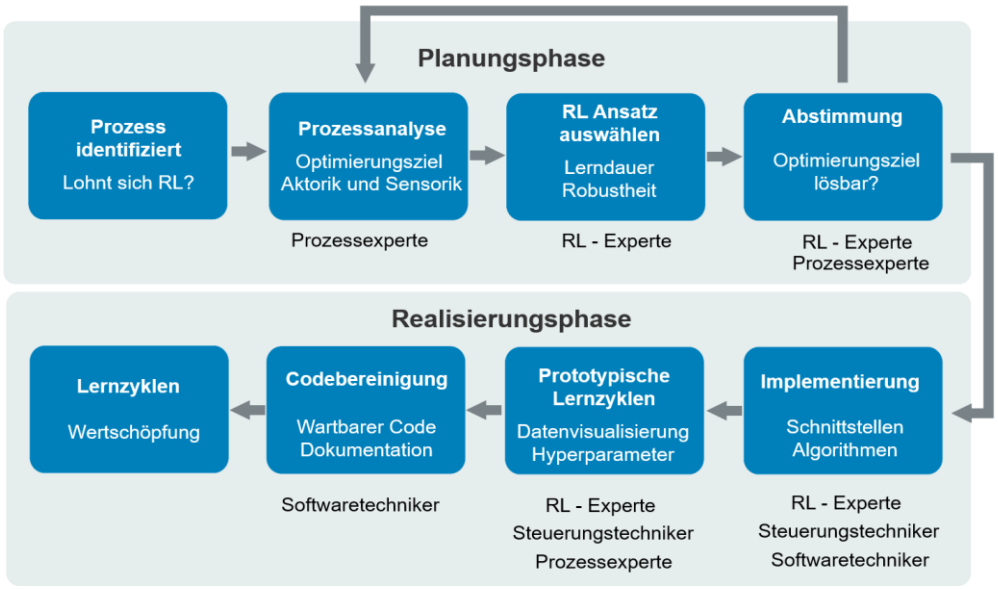

Abbildung 41: Prozess zur Integration von Reinforcement Learning

Softwaredokumentation

Die Software, welche während des Projektes entwickelt und für die Anwendungsbeispiele verwendet wurde, ist auf der Kommunikationsplattform THEMIS des FKM zugänglich. Um den KMU eine weiterführende Nutzung der verwendeten Methoden und Architekturmuster zu ermöglichen, wurde eine ausführliche Dokumentation erstellt. So gewinnen die Leser einen Überblick über bereits vorhandene Implementierungen und können diese als Hilfestellung für eigene Anwendungen einsetzen. Die Softwaredokumentation wird ebenfalls auf der Kommunikationsplattform THEMIS zur Verfügung gestellt.

Wissenschaftliche Veröffentlichungen

Während der Handlungsleitfaden darauf ausgelegt ist, die Integration von Reinforcement Learning in der Industrie, speziell in KMU, zu erleichtern, wird mit dem veröffentlichten Paper ein interessiertes Fachpublikum erreicht. Vorgestellt wurde die Publikation auf der hochrangigen International Conference on Robotics and Automation (ICRA) im Mai 2019 in Montreal.

Ausgehend von der in Abschnitt 4.2.1 beschriebenen „Guided Policy Search“ wird ein neuer Ansatz zur Verbesserung der Robustheit komplexer Manipulationsaufgaben beschrieben. Die zugrundeliegende Problemstellung begründet sich auf dem Fehlen umfassender Daten-sätze zur Trajektorienplanung in industriellen Anwendungen. Die vorgestellte Strategie namens „Generative Motor Reflexes“ berücksichtigt dies und ermöglicht im Vergleich zu bisherigen Ansätzen eine Ausweitung des Zustandsraumes. Der Ansatz wurde sowohl innerhalb einer Simulationsumgebung als auch an einem Demonstrator getestet. So konnte gezeigt werden, dass die Verwendung der Generative Motor Reflexes robustere Manipulationen bei geringerem Trainingsaufwand ermöglicht.

Die veröffentlichten Ergebnisse haben hohe Relevanz für das Projekt, da besonders in KMU häufig wenig Trainingsdaten für Reinforcement Learning zur Verfügung stehen. Aufgrund der hohen Anforderungen an die Robustheit von Prozessen in der industriellen Anwendung bedeutet dies einen signifikanten Fortschritt. Diese erweiterten Möglichkeiten der Technologie sind daher in dem oben beschriebenen Handlungsleitfaden berücksichtigt.

Webinare

Die Webinare, welche während der gesamten Projektlaufzeit durchgeführt wurden, hatten eine große Bedeutung bei der Einbindung der Industrie und dem angestrebten Technologietransfer. Für eine ausführliche Dokumentation wurde zu jedem Webinar ein Protokoll angefertigt. Diese sind auf der Kommunikationsplattform THEMIS zugänglich, welche vom FKM betrieben wird.

4.5.3 Arbeitspaket 5.2: Erstellung von Zwischenberichten, Reports und des Abschlussberichtes

4.5.3.1 Ziel des Arbeitspaketes

Das AP5.2 umfasst die kontinuierliche und vollständige Dokumentation der Ergebnisse während der gesamten Projektlaufzeit. Diese dient insbesondere zur Identifikation der wesentlichen Erkenntnisse zur Präsentation und Diskussion mit dem Projektausschuss. In die vollständige Dokumentation gehen unter anderem die Ergebnisse aus Workshops und Befragungen ein. Es werden besonders relevante Szenarien für den Einsatz intelligenter selbstlernender Produktionssysteme dokumentiert und Anforderungen an diese herausgestellt. Die Dokumentation wird dem Projektausschuss zugänglich gemacht.

4.5.3.2 Durchgeführte Arbeiten

Die Dokumentation von Ergebnissen wurde während des gesamten Projektes mit Hilfe von Protokollen und Berichten gewährleistet. Sowohl für die durchgeführten Webinare als auch für die Sitzungen des Projektausschusses und die Projekttreffen wurden Protokolle angefertigt. In diese wurden Zwischenergebnisse und Erfahrungen aufgenommen. Diese wurden unter anderem für die Abschlussveranstaltung im September 2019 aufbereitet und präsentiert. Alle Ergebnisse des Forschungsprojektes wurden in dem vorliegenden Abschlussbericht zusammengefasst.

4.5.3.3 Ergebnisse

Alle Protokolle zu Webinaren, Projektausschusstreffen und Projekttreffen sind auf der Kommunikationsplattform THEMIS verfügbar. Auch der hier vorliegende Abschlussbericht wird dort

verfügbar gemacht. So können die KMU auch nach Abschluss des Projektes weiterhin auf die Dokumentation der Forschungsergebnisse zugreifen.

Mit dem Abschluss des AP 4 und des AP 5 wurde der letzte Meilenstein, die abgeschlossene Validierung anhand eines industriellen Anwendungsbeispiels und die Fertigstellung des Technologietransferkonzepts abgeschlossen.

5 Dokumentation der Zielerreichung

5.1 Gegenüberstellung der erreichten und der geplanten Ziele

Zusätzlich zu der detaillierten Erläuterung der durchgeführten Arbeiten in dem vorherigen Kapitel ist nachfolgend in **Tabelle *5*** eine detaillierte Gegenüberstellung der geplanten Ziele und den tatsächlich durchgeführten Arbeiten gegeben.

AP	Geplante Ziele	durchgeführte Arbeiten und Ergebnisse
1.1	• Erarbeitung industrieseitiger Erwartungen an selbstlernende Produktionssysteme • Untersuchung des erwarteten Mehrwerts • Anforderungen an selbstlernende Produktionsprozesse analysieren • Mögliche Anwendungsszenarien erarbeiten	• Durchgeführte Arbeiten vgl. Kapitel 4.1.1 • Alle Teilziele wurden erreicht
1.2	• Untersuchung der Eignung existierender Lernverfahren für das Produktionsumfeld • Erarbeitung einer Übersicht über in der Produktion einsetzbare Lernverfahren	• Durchgeführte Arbeiten vgl. Kapitel 4.1.2 • Alle Teilziele wurden erreicht
2.1	• Entwicklung einer ganzheitlichen Systemarchitektur für die Integration selbstlernender Prozessregelungen	• Durchgeführte Arbeiten vgl. Kapitel 4.2.24.2.3 • Alle Teilziele wurden erreicht
2.2	• Untersuchung der Eignung unterschiedlicher Formen der Zielsystemgestaltung für lernfähige Steuerungen	• Durchgeführte Arbeiten vgl. Kapitel 4.2.3 • Alle Teilziele wurden erreicht
2.3	• Algorithmische Beschreibung der Lernverfahren für spezifische, exemplarische Anwendungsszenarien • Erarbeitung der erforderlichen Rahmenbedingungen zur Einbettung der Lernverfahren in das Gesamtsystem	• Durchgeführte Arbeiten vgl. Kapitel 4.2.4 • Alle Teilziele wurden erreicht
3.1	• Konstruktion, Fertigung und Aufbau des Forschungsdemonstrators • Untersuchung der lernfähigen Ablaufplanung	• Durchgeführte Arbeiten vgl. Kapitel 4.3.1 • Alle Teilziele wurden erreicht
3.2	• Integration der zuvor entwickelten Systemkomponenten in den Demonstrator • Implementierung der Algorithmen zur selbstlernenden Prozessregelung aus AP2.3	• Durchgeführte Arbeiten vgl. Kapitel 4.3.1 • Alle Teilziele wurden erreicht

4.1	• Aufstellen von Testfällen für einzelne Systemkomponenten und das Gesamtsystem • Durchführung von Versuchsreihen am Forschungsdemonstrator zur Validierung der Funktionsfähigkeit • Aufzeigen der potentiellen Grenzen der entwickelten Algorithmen bzgl. Stabilität, Robustheit, Transparenz	• Durchgeführte Arbeiten vgl. Kapitel 4.4.1 • Alle Teilziele wurden erreicht
4.2	• Validierung der zuvor entwickelten Methodik anhand eines industriellen Anwendungsbeispiels • Abgleich der zu Beginn gestellten Anforderungen mit gefundenen Lösungen • Bewertung der Wirtschaftlichkeit mit Hilfe des NOWS-Verfahrens	• Durchgeführte Arbeiten vgl. Kapitel 4.4.2 • Das NOWS-Verfahren vernachlässigt die Risiken und wurde daher während der Projektlaufzeit als nicht geeignet zur Wirtschaftlichkeitsbewertung festgelegt. Die Wirtschaftlichkeit wurde im direkten Anwendungsfall bei der AZO GmbH & Co. KG gezeigt. • Alle Teilziele wurden erreicht
5.1	• Entwicklung und Umsetzung eines Technologietransferkonzepts • Erstellung eines Handlungsleitfadens für KMU	• Durchgeführte Arbeiten vgl. Kapitel 4.5.2 • Alle Teilziele wurden erreicht
5.2	• Erstellung von Berichten und Reports zur Dokumentation der erzielten Ergebnisse	• Durchgeführte Arbeiten vgl. Kapitel 4.5.3 • Alle Teilziele wurden erreicht

Tabelle 5: Gegenüberstellung erreichte und geplante Ziele

6 Plan zum Ergebnistransfer in die Wirtschaft

6.1 Durchgeführter und geplanter Ergebnistransfer

Zusätzlich zu den Zwischen- und Schlussberichten wird durch die Gewährleistung des Informationstransfers zu den Unternehmen und in die Wirtschaft während der Laufzeit die Verbreitung der Forschungsergebnisse durch verschiedene Maßnahmen sichergestellt. Zentrale Maßnahmen während der Projektlaufzeit sind in **Tabelle *6*** dargestellt.

Zeitraum	**Maßnahme**	**Ziel**
10 / 2017	Projektkickoff mit dem Industriearbeitskreis (kontinuierliche Information über Projektverlauf und Ergebnisse; Diskussion des weiteren Vorgehens).	Information von Firmen mit Bezug zu den Systemeinzelkomponenten, Einbeziehen der Industrie
12 / 2017	Projektinformation auf der Internetpräsenz des Cybernetics Lab IMA & IfU	Interesse bei einem breiten Publikum wecken
12 / 2017	Webinar (kontinuierliche Information über Projektverlauf und Ergebnisse; Diskussion des weiteren Vorgehens)	Ergebnisse aus dem Kickoff, Einführung in Machine Learning
02 / 2018	Webinar (kontinuierliche Information über Projektverlauf und Ergebnisse; Diskussion des weiteren Vorgehens)	Einführung Deep Reinforcement Learning, Vorstellung der Anwendungsszenarien
02 / 2018	Internetpräsenz: http://www.i40-inpuls.de	Interesse bei einem breiten Publikum wecken
02 / 2018	Projekttreffen zur Use-Case Absprache mit AZO GmbH & Co KG	Analyse des pneumatischen Schüttgutförderers, Definition des exakten Use Cases.
03 / 2018	2. Treffen des Projektbegleitenden Ausschusses	Vorstellung verschiedener Lernverfahren, Erarbeitung der industriellen Anforderungen
06 / 2018	Webinar (kontinuierliche Information über Projektverlauf und Ergebnisse; Diskussion des weiteren Vorgehens)	Systemarchitektur, Modellfreies, Modelbasiertes Reinforcement Learning

07 / 2018	Projekttreffen zur Use-Case Absprache mit AZO GmbH & Co KG	Analyse des pneumatischen Schüttgutförderers, Definition der Anforderungen, der Maschinenschnittstellen und der Bewertungsfunktion.
09 / 2018	Webinar (kontinuierliche Information über Projektverlauf und Ergebnisse; Diskussion des weiteren Vorgehens)	Diskussion der 3 Use Cases. Informationen über die Kommunikationsinfrastruktur bei der AZO GmbH & Co KG.
10 / 2018	3. Treffen des projektbegleitenden Ausschusses bei der AZO GmbH & Co KG.	Vorstellung und Diskussion des Ansatzes für industrielles Reinforcement Learning, Demonstration des selbstlernenden pneumatischen Schüttgutförderers
02 / 2019	Projekttreffen zur Use-Case Absprache mit AZO GmbH & Co KG.	Testen der Generativen Motor Reflexe zur Steuerung des pneumatischen Schüttgutförderers
03 / 2019	4. Treffen des projektbegleitenden Ausschusses	Vorstellung des Projektfortschritts, Diskussion von Unsicherheitsquellen im industriellen Reinforcement Learning, Vorstellung der Struktur des Handlungsleitfadens
05 / 2019	Ennen, P., Bresenitz, P., Vossen, R., & Hees, F. (2019, May). Learning Robust Manipulation Skills with Guided Policy Search via Generative Motor Reflexes. In *2019 International Conference on Robotics and Automation (ICRA)* (pp. 7851-7857). IEEE.	Zielgruppenorientierte Information für Fachpublikum
09/2019	Veröffentlichung des Handlungsleitfadens	Verbreitung der Forschungsergebnisse in KMU
09 / 2019	Abschlussveranstaltung	Projektvorstellung, Vorstellung Handlungsleitfaden
Voraussichtlich 2019	Dissertation: Ennen, P. (noch nicht publiziert). Industrial Reinforcement Learning with Stabilizing Gradients.	Ergebnistransfer in die Wissenschaft

Gesamter Projektzeitraum	Durchführung von Bachelor-/Masterarbeiten	Weiterbildung und Motivation von Studenten durch Mitarbeit an aktuellen, praxisrelevanten Fragestellungen der Forschung

Tabelle 6: Tabellarische Übersicht der durchgeführten Maßnahmen.

Die Veröffentlichungen auf der Instituts-Internetseite erfolgten routinemäßig. Die wissenschaftliche Veröffentlichung wurde im Mai 2019 auf der international renommierten Konferenz „International Conference on Robotics and Automation“ (ICRA) vorgestellt. Im Rahmen des Projektes wurden verschiedene Bachelor- und Masterarbeiten betreut. Kurz nach Projektabschluss wurde eine Dissertation mit dem Schwerpunkt industrielles Reinforcement Learning fertiggestellt [43].

Maßnahme	**Ziel**	**Rahmen**	**Zeitraum**
Integration der Ergebnisse in Vorlesungs- und Übungsinhalte und zur Berufsweiterbildung	Information und Ausbildung des Ingenieurnachwuchses	**Vorlesungen:** Informatik im Maschinenbau I und II	ab 2020
Durchführung von Demonstrationsworkshops	Verbreitung der Forschungsergebnisse	Führungen und Demonstrationen für Vertreter von KMU am IfU	ab 2020
Weitergabe des Handlungsleitfadens	Detailinformation interessierter, deutscher Firmen	**Inhalte:** Hilfestellung zur Planung des Einsatzes von Reinforcement Learning im industriellen Kontext	nach Projektende

Tabelle 7: Tabellarische Übersicht der geplanten Transfermaßnahmen.

Der Inhalt des Handlungsleitfaden wird in Kapitel 4.5.2 beschrieben. Auf dessen Verfügbarkeit wird auf der Internetseite des VDMA verwiesen. Zur Verbreitung stehen sowohl eine Print- als auch eine digitale Version zur Verfügung. Die Aktualisierung der Vorlesungsinhalte hinsichtlich neuer Forschungsergebnisse erfolgt turnusmäßig am Forschungsinstitut. Daher wird das Konzept für den weiteren Transfer als sehr gut realisierbar eingeschätzt.

6.2 Aussagen zur voraussichtlichen industriellen Umsetzung der FuE-Ergebnisse nach Projektende

Ziel von InPulS war es, Handlungskonzepte zur Implementierung von intelligenten, selbstlernenden Produktionsprozessen speziell für KMU zu entwickeln. Die Forschungsergebnisse des Projektes sind vorteilhaft für unterschiedliche Anwendergruppen nutzbar. So können sowohl KMUs aus der Robotik- und Automatisierungsbranche als auch aus der produzierenden Industrie von der Anwendung von Reinforcement Learning profitieren. Die ermittelten Handlungskonzepte ermöglichen den KMUs dabei

- die Entwicklung einer Einführungsstrategie für Reinforcement Learning inklusive der Entwicklung von Geschäftsmodellen sowie
- die Optimierung vorhandener Prozesse durch robuste, dateneffiziente Verfahren.

Vor diesem Hintergrund stellt der entstandene Leitfaden eine vielseitig anwendbare und erweiterbare Basis dar. Es werden theoretische Hintergründe vermittelt und besondere Anforderungen an Reinforcement Learning im industriellen Kontext erläutert. Darauf aufbauend können Betriebe eigene Potenziale, aber auch mögliche Risiken abschätzen. Sind mögliche Anwendungsbereiche identifiziert, wird durch den vorgestellten Werkzeugkasten die Anforderungsanalyse erleichtert. Darüber hinaus gibt der Leitfaden eine Hilfestellung zur Implementierung von Projekten mit Reinforcement Learning. Die im Rahmen des vorliegenden Projektes entwickelte Software und die ausführliche Dokumentation geben eine Übersicht, welche Tools und Architekturmuster bereits vorhanden sind und eingesetzt werden können. Dass die vorgeschlagenen Handlungskonzepte eine gute Grundlage für die Entwicklung einer Strategie zu Integration von industriellem Reinforcement Learning in KMU bilden, zeigt die AZO GmbH + Co. KG. Nachdem dort im Rahmen von InPulS ein Use Case entwickelt und umgesetzt wurde, soll nun ein Demonstrator zur weiteren Forschung aufgebaut werden. Die ausführliche Dokumentation und die Handlungskonzepte werden daher als anforderungsgerechte Basis für die Entwicklung einer Einführungsstrategie eingeschätzt.

Die besonderen Anforderungen an Reinforcement Learning im industriellen Kontext stellen, neben den häufig fehlenden finanziellen Mitteln und Kapazitäten in KMU, ein zusätzliches Hindernis für den Einsatz dieser Technologie dar. Dazu gehören vor allem die Robustheit gegenüber Unsicherheiten wie veränderten Umgebungsbedingungen und der bestmögliche Umgang mit wenigen vorhandenen Trainingsdaten. Mit Hilfe eines am IfU aufgebauten Demonstrators konnte gezeigt werden, dass die in dem Paper „Learning Robust Manipulation Skills with Guided Policy Search via Generative Motor Reflexes“ vorgestellte Strategie die Robustheit und die Dateneffizienz bisheriger Lernverfahren verbessern kann. Durch diesen technologischen Fortschritt wird eine weitere Hilfestellung für KMU in der Entwicklung von Anwendungen selbstlernender Prozesse geleistet.

Der Nutzen für die Unternehmen liegt potenziell in einer Effizienzsteigerung durch intelligente Steuerung. Durch den Einsatz von Reinforcement Learning, besonders unter Berücksichtigung der Verbesserungen hinsichtlich Robustheit und Dateneffizienz, können zum Beispiel die **Verkürzung von Rüstzeiten** und eine **Reduktion der beim Anfahren produzierten Ausschussware** erreicht werden. Insbesondere kann eine Steuerung erlernt werden, welche in der Lage ist, auf veränderte Umweltbedingungen und Störungen zu reagieren. Dies macht die Technologie für unterschiedlichste Anwender attraktiv, weshalb angenommen werden kann, dass sie in Zukunft breitere Verwendung in den genannten Branchen finden wird.

6.3 Zusammenstellung von Veröffentlichungen

Ennen, Philipp & Benmoussa, Pia & Vossen, René & Hees, Frank. (2019): Learning Robust Manipulation Skills with Guided Policy Search via Generative Motor Reflexes, *2019 International Conference on Robotics and Automation (ICRA)*, 7851-7857.

VDMA, FKM, Institut für Unternehmenskybernetik e.V. (Hrsg.): Leitfaden Selbstlernende Produktionsprozesse. Einführungsstrategie für Reinforcement Learning in der industriellen Praxis, Frankfurt am Main, Aachen 2019.

Ennen, P. (noch nicht publiziert). Industrial Reinforcement Learning with Stabilizing Gradients.

6.4 Angaben über gewerbliche Schutzrechte

Im Rahmen des Projektes wurden keine Schutzrechte beantragt.

7 Zusammenfassung

Das wesentliche Forschungsziel war die Aufarbeitung von kontextbasierten Lernverfahren in einem **selbstlernenden Produktionsprozess** sowie die Bewertung dieser Verfahren für konkrete Einsatzmöglichkeiten in KMU. Mithilfe von Reinforcement Learning können Prozesse gesteuert werden, für die eine Modellierung mit konventionellen Methoden zu komplex wäre. Hier lässt sich zum einen ein Effizienzgewinn bezüglich konventioneller Steuerungen erreichen und zum anderen entfällt häufig aufwendiges manuelles Einstellen der Anlagenparameter.

Es hat sich gezeigt, dass die industrielle Anwendung neue Anforderungen an Reinforcement Learning Algorithmen stellt. Diese Anforderungen sind insbesondere eine hohe Robustheit gegenüber Unsicherheiten, eine sichere Explorationsphase und ein dateneffizienter Trainingsprozess. Im Rahmen des Projektes wurde ein Algorithmus entwickelt, die **Self-Regulating-Motor-Unit**, der diese industriellen Anforderungen erfüllt. Der Algorithmus basiert auf dem modellbasierten Reinforcement Learning Verfahren **Guided Policy Search** und zeichnet sich insbesondere im Vergleich zu aktuellen Guided Policy Search Varianten durch eine signifikant erhöhte Robustheit gegenüber Störungen und Unsicherheiten aus. Damit ist das entwickelte Verfahren das erste Reinforcement Learning Verfahren, das die speziellen industriellen Anforderungen erfüllt und somit für die Anwendung geeignet ist.

Die Anwendung auf dem Schüttgutförderer hat das enorme Potential des industriellen Reinforcement Learning zur Prozesssteuerung bestätigt. Mittels des neuen Verfahrens konnte eine Effizienzsteigerung von **20%** gegenüber einer initial durch einen Prozessexperten gewählten Steuerungsstrategie erreicht werden. Außerdem konnte eine auf einem Kunststoffgranulat trainierte Steuerungsstrategie bei gleicher Leistungsfähigkeit auf einen Förderungsprozess von Senfmehl übertragen werden. Mithilfe einer solchen selbstlernenden Steuerungsstrategie können somit potentiell **geringere Rüstzeiten** und eine höhere **Produktvielfalt** erreicht werden, indem der Algorithmus sich selbstständig auf die neuen Gegebenheiten anpasst. So kann die Förderanlage stets in der Nähe des **optimalen Betriebspunktes** gefahren werden und wartungsaufwendige Verstopfungen der Anlage werden frühzeitig erkannt und vermieden. Im Rahmen des Projektes wurde so einer der ersten industriellen Anwendungsfälle von Reinforcement Learning realisiert und das Potential dieser hochkomplexen Verfahren auch für KMU demonstriert.

Es bleibt die Frage, wie sich ein solcher Algorithmus auf weitere Szenarien übertragen lässt. Hierfür müssen in Zukunft **weitere Anwendungsfelder** identifiziert und weitere Algorithmen entwickelt werden. Bei der Entwicklung dieser Algorithmen müssen insbesondere die speziellen Anforderungen der Industrie hinsichtlich Sicherheit und Robustheit erfüllt werden. Hier ist weitergehende anwendungsorientierte Grundlagenforschung notwendig, um neue Anwendungsgebiete und entsprechende Algorithmen für Reinforcement Learning zu entwickeln.

Der im Rahmen dieses Projektes entwickelte **Handlungsleitfaden** gibt KMU einen Überblick über die Potenziale dieser Lernverfahren und stellt die wichtigsten Projekterkenntnisse in Form von Werkzeugkästen vor. Außerdem wird der Ablauf einer Reinforcement Learning Einführung skizziert. Somit ermöglicht der Handlungsleitfaden den Transfer der Forschungsergebnisse in die industrielle Praxis und die Erarbeitung neuer Anwendungsfälle und Geschäftsmodelle für KMU.

8 Anhang

8.1 Literaturverzeichnis

[1] M. P. Deisenroth, G. Neumann, and J. Peters, ‘A survey on policy search for robotics’, *Found. Trends® Robot.*, vol. 2, no. 1–2, pp. 1–142, 2013.

[2] P. Hilgraf, ‘Grundlagen der pneumatischen Förderung’, in *Pneumatische Förderung*, Springer, 2019, pp. 109–232.

[3] G. Mittelstand, ‘Motor der deutschen Wirtschaft (2014)’, *Bundesminist. Für Wirtsch. Energ.*

[4] ‘Multi-Annual Roadmap for Horizon 2020’, *SPARC Robot. EuRobotics AISBL Bruss.*, 2017.

[5] S. Russell and J. F. Canny, *Künstliche Intelligenz. Ein moderner Ansatz, 1st edn. Informatik*. Pearson Studium, München, Boston [ua], 2004.

[6] C. Büscher, *Ontologiebasierte Kennzahlentwicklung für Virtual Production Intelligence*. VDI Verlag GmbH, 2015.

[7] A. Y. Ng *et al.*, ‘Autonomous inverted helicopter flight via reinforcement learning’, in *Experimental robotics IX*, Springer, 2006, pp. 363–372.

[8] Y. LeCun, Y. Bengio, and G. Hinton, ‘Deep learning’, *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[9] T. Niemueller, S. Reuter, D. Ewert, A. Ferrein, S. Jeschke, and G. Lakemeyer, ‘The carologistics approach to cope with the increased complexity and new challenges of the RoboCup logistics league 2015’, in *Robot Soccer World Cup*, 2015, pp. 47–59.

[10] B. Vogel-Heuser *et al.*, ‘Agentenbasierte cyberphysische Produktionssysteme’, *Atp Mag.*, vol. 57, no. 09, pp. 36–45, 2015.

[11] B. Hernández, J. Jiménez, and M. J. Martín, ‘Customer behavior in electronic commerce: The moderating effect of e-purchasing experience’, *J. Bus. Res.*, vol. 63, no. 9–10, pp. 964–971, 2010.

[12] F. Graser, ‘Selbstlernende IT-Systeme halten Einzug in die Fertigung’. [Online]. Available: http://www.elektronikpraxis.vogel.de/selbstlernende-it-systeme-halten-einzug-in-die-fertigung-a-584633/. [Accessed: 22-Aug-2017].

[13] H. Giese, S. Burmester, F. Klein, D. Schilling, and M. Tichy, ‘Multi-agent system design for safety-critical self-optimizing mechatronic systems with UML’, in *OOPSLA*, 2003, pp. 21–32.

[14] P. Leitão, ‘Agent-based distributed manufacturing control: A state-of-the-art survey’, *Eng. Appl. Artif. Intell.*, vol. 22, no. 7, pp. 979–991, 2009.

[15] W. Shen, Q. Hao, H. J. Yoon, and D. H. Norrie, ‘Applications of agent-based systems in intelligent manufacturing: An updated review’, *Adv. Eng. Inform.*, vol. 20, no. 4, pp. 415–431, 2006.

[16] J. Langford and B. Zadrozny, ‘Relating reinforcement learning performance to classification performance’, in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 473–480.

[17] P. Leitão, V. Mařík, and P. Vrba, ‘Past, present, and future of industrial agent applications’, *IEEE Trans. Ind. Inform.*, vol. 9, no. 4, pp. 2360–2372, 2012.

[18] P. Leitão and S. Karnouskos, *Industrial Agents: Emerging Applications of Software Agents in Industry*. Morgan Kaufmann, 2015.

[19] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, ‘An efficient k-means clustering algorithm: Analysis and implementation’, *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 7, pp. 881–892, 2002.

[20] T. Zhang, R. Ramakrishnan, and M. Livny, ‘BIRCH: an efficient data clustering method for very large databases’, in *ACM Sigmod Record*, 1996, vol. 25, pp. 103–114.

[21] P. Ennen, S. ReuteR, R. Vossen, and S. Jeschke, ‘Automated Production Ramp-up Through Self-Learning Systems’, *Procedia CIRP*, vol. 51, pp. 57–62, 2016.

[22] V. Mnih *et al.*, ‘Human-level control through deep reinforcement learning’, *Nature*, vol. 518, no. 7540, p. 529, 2015.

[23] J. Peters, K. Mulling, and Y. Altun, ‘Relative entropy policy search’, in *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.
[24] S. Levine and V. Koltun, ‘Guided policy search’, in *International Conference on Machine Learning*, 2013, pp. 1–9.
[25] D. Silver *et al.*, ‘Mastering the game of Go with deep neural networks and tree search’, *nature*, vol. 529, no. 7587, p. 484, 2016.
[26] K. Muelling, J. Kober, and J. Peters, ‘Learning table tennis with a mixture of motor primitives’, in *2010 10th IEEE-RAS International Conference on Humanoid Robots*, 2010, pp. 411–416.
[27] S. Levine, N. Wagener, and P. Abbeel, ‘Learning contact-rich manipulation skills with guided policy search (2015)’, *ArXiv Prepr. ArXiv150105611*.
[28] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert, ‘Learning movement primitives’, in *Robotics research. the eleventh international symposium*, 2005, pp. 561–572.
[29] D. Ewert, *Adaptive Ablaufplanung für die Fertigung in der Factory of the Future*. VDI Verlag, 2014.
[30] T. Niemueller, A. Ferrein, S. Reuter, S. Jeschke, and G. Lakemeyer, ‘The RoboCup Logistics League as a Holistic Multi-Robot Smart Factory Benchmark’, 2015.
[31] P. Flachskampf, A. Voets, and C. Michulitz, ‘Die nutzenorientierte Wirtschaftlichkeitsbewertung als Instrument zur Entscheidungsfindung im Management’, *SemRadar Z. Für Syst. Entscheid. Im Manag.*, vol. 7 (1), pp. 29–57, 2008.
[32] D. Weydandt, *Beteiligungsorientierte wirtschaftliche Bewertung von technischen Investitionen für prozessorientierte Fertigungsinseln*. Shaker, 2000.
[33] S. Printz, R. Vossen, and S. Jeschke, ‘Adaption of the profitability estimation focus on benefits due to personal affection’, *Assess. Methodol. Energy Mobil. Real World Appl.*, pp. 249–263, 2015.
[34] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
[35] V. Mnih *et al.*, ‘Playing atari with deep reinforcement learning’, *ArXiv Prepr. ArXiv13125602*, 2013.
[36] W. H. Montgomery and S. Levine, ‘Guided policy search via approximate mirror descent’, in *Advances in Neural Information Processing Systems*, 2016, pp. 4008–4016.
[37] S. Levine and P. Abbeel, ‘Learning neural network policies with guided policy search under unknown dynamics’, in *Advances in Neural Information Processing Systems*, 2014, pp. 1071–1079.
[38] W. Montgomery, A. Ajay, C. Finn, P. Abbeel, and S. Levine, ‘Reset-free guided policy search: Efficient deep reinforcement learning with stochastic initial states’, in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 3373–3380.
[39] P. Ennen, P. Bresenitz, R. Vossen, and F. Hees, ‘Learning Robust Manipulation Skills with Guided Policy Search via Generative Motor Reflexes’, in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7851–7857.
[40] R. Kupper, ‘Reset-free Generative Motor Reflexes: Efficient Learning of Robust Manipulation Skills’, RWTH Aachen University, 2019.
[41] C. Finn *et al.*, ‘Guided policy search code implementation, 2016’, *Softw. Available Rll Berkeley Edugps*.
[42] M. Plappert *et al.*, ‘Multi-goal reinforcement learning: Challenging robotics environments and request for research’, *ArXiv Prepr. ArXiv180209464*, 2018.
[43] P. Ennen, ‘Industrial Reinforcement Learning with Stabilizing Gradients’, noch nicht publiziert.
[44] G. Brockman *et al.*, ‘Openai gym’, *ArXiv Prepr. ArXiv160601540*, 2016.

8.2 Abbildungsverzeichnis

8.3 Tabellenverzeichnis

FKM

Forschung im VDMA

Forschungskuratorium Maschinenbau e.V.
Lyoner Straße 18
60528 Frankfurt am Main

Telefon +49 69 66 03 16 81
Fax +49 69 66 03 16 73

www.fkm-net.de

VDMA

AiF Mitglied