THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

On Reinforcement Learning and Digital Twins for Intelligent Automation

CONSTANTIN CRONRATH

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2023

On Reinforcement Learning and Digital Twins for Intelligent Automation

CONSTANTIN CRONRATH
ISBN 978-91-7905-838-8

Doktorsavhandlingar vid Chalmers tekniska högskola.
Ny serie nr 5304
ISSN 0346-718X

This thesis has been prepared using LaTeX.

Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Phone: +46 (0)31 772 1000
www.chalmers.se

Printed by Chalmers Digitaltryckeri
Gothenburg, Sweden, April 2023

To those who wander.

Abstract

Current trends, such as the fourth industrial revolution and sustainable manufacturing, enable and necessitate manufacturing automation to become more intelligent to meet ever new design requirements in terms of flexibility, speed, quality, and cost.

Two distinct research streams towards intelligent manufacturing exist in the scientific literature: the model-based digital twin approach and the data-driven learning approach. Research that incorporates advantages of the one into the other approach is frequently called for.

Accordingly, this thesis investigates how machine learning can be used to mitigate the model-system mismatch in digital twins and how prior model-based knowledge can be introduced in reinforcement learning in the context of intelligent automation.

In terms of mitigating mismatches in digital twins, research presented in this thesis suggests that learning is of limited usefulness when employed naively in static and systemic mismatch scenarios. In such settings, blackbox optimization algorithms, that leverage properties of the problem, are more useful in terms of sample-efficiency, performance within a given budget, and regret (i.e. when compared to an optimal controller). Learning seems to be of some merit, however, in individualized production control and when used for adapting parameters within a digital twin.

An additional research outcome presented in this thesis is a principled method for incorporating prior knowledge in form of automata specifications into reinforcement learning. Furthermore, the benefits of introducing rich prior model-based knowledge in form of economic non-linear model predictive controllers as model class for function approximation in reinforcement learning is demonstrated in the context of energy optimization.

Lastly, this thesis highlights that adaptive economic non-linear model predictive control may be understood as a unifying framework for both research streams towards intelligent automation.

Keywords: Intelligent automation, reinforcement learning, digital twin.

List of Publications

This thesis is based on the following publications:

[A] **Constantin Cronrath**, and Bengt Lennartson, "How Useful is Learning in Mitigating Mismatch between Digital Twins and Physical Systems?". *Published in IEEE Transactions on Automation Science and Engineering, (Early Access),* Dec. 2022.

[B] **Constantin Cronrath**, Abolfazl Rezaei Aderiani, and Bengt Lennartson, "Enhancing Digital Twins through Reinforcement Learning". *Published in Proceedings of the 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, Vancouver, Canada, pp. 293–298, Sept. 2019.

[C] Anders Sjöberg, Magnus Önnheim, Otto Frost, **Constantin Cronrath**, Emil Gustavsson, Bengt Lennartson, and Mats Jirstrand, "Online Geometry Assurance in Individualized Production by Feedback Control and Model Calibration of Digital Twins". *Published in Journal of Manufacturing Systems*, vol. 66, pp. 71–81, Jan. 2023.

[D] **Constantin Cronrath**, Tom P. Huck, Christoph Ledermann, Torsten Kröger, and Bengt Lennartson, "Relevant Safety Falsification by Automata Constrained Reinforcement Learning". *Published in Proceedings of the 2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, Mexico City, Mexico, pp. 2273–2280, Aug. 2022.

[E] Mattias Hovgard, **Constantin Cronrath**, Kristofer Bengtsson, and Bengt Lennartson, "Adaptive Energy Optimization of Flexible Robot Stations". *Revised version submitted to IEEE Transactions on Automation Science and Engineering,* Mar. 2023.

Other publications by the author, not included in this thesis, are:

[F] **Constantin Cronrath**, Bengt Lennartson, and Marco Lemessi, "Energy Reduction in Paint Shops Through Energy-Sensitve On-Off Control". *Published in Proceedings of the 2016 IEEE 12th International Conference on Automation Science and Engineering (CASE),* Fort Worth, Texas, pp. 1282-1288, Aug. 2016.

[G] Emilio Jorge, Lucas Brynte, **Constantin Cronrath**, Oskar Wigström, K. Bengtsson, Emil Gustavsson, Bengt Lennartson, and Mats Jirstrand, "Reinforcement Learning in Real-Time Geometry Assurance". *Published in Proceedings of the 51st CIRP Conference on Manufacturing Systems,* Stockholm, pp. 1073-1078, Jun. 2018.

[H] **Constantin Cronrath**, Emilio Jorge, John Moberg, Mats Jirstrand, and Bengt Lennartson, "BAgger: A Bayesian Algorithm for Safe and Query-Efficient Imitation Learning". *Published in Proceedings of the Machine Learning in Robot Motion Planning–IROS 2018 Workshop,* Madrid, Spain, Oct. 2018.

[I] **Constantin Cronrath**, Ludvig Ekström, and Bengt Lennartson, "Formal Properties of the Digital Twin–Implications for Learning, Optimization, and Control". *Published in Proceedings of the 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE),* Hong Kong, pp. 679-684, Sep. 2020.

[J] Tom P. Huck, Yuvaraj Selvaraj, **Constantin Cronrath**, Christoph Ledermann, Martin Fabian, Bengt Lennartson, and Torsten Kröger, "Hazard Analysis of Collaborative Automation Systems: A Two-layer Approach based on Supervisory Control and Simulation". *Accepted to the 2023 IEEE International Conference on Robotics and Automation (ICRA),* London, England, arXiv preprint arXiv:2209.12560, May 2023.

Acknowledgments

This thesis marks the end of a years-long research journey. Many have accompanied me on this adventure and have contributed in one way or another to its completion. While acknowledging every single one of them here would be reasonable, highlighting a few, who had an outsized impact during these years, will be, however, significantly less wordy.

First and foremost, I would like to express my gratitude to my supervisor Professor Bengt Lennartson for his seemingly unwavering faith in my capabilities. Whenever needed, he has provided gentle guidance and feedback from anywhere in the world at anytime of the day, while also giving me, whenever I wanted, ample opportunities to find my own way and to put forward my best. The research in this thesis would not exist without him.

This research would also not have been possible without the financial support by The Swedish Foundation for Strategic Research, through the Smart Assembly 4.0 project within the Winquist Laboratory, and by Vinnova under the ITEA3 AITOC project. The support is gratefully acknowledged.

Most of the presented research has also been conducted in collaboration with other researchers. Of all my co-authors, I would like to thank especially Oskar Wigström, Emilio Jorge, Magnus Önnheim, Anders Sjöberg, Yuvaraj Selvaraj, Tom P. Huck, Martin Kaiser, and Mattias Hovgard for the many creative technical discussions, and their good company at odd hours before deadlines and slow mornings thereafter.

Furthermore, I would like to thank the senior members of the automation research group – Martin, Knut, Petter, Kristoffer, Dean, and Sahar – for sharing their knowledge, wisdom, and constructive feedback, starting from my first day in the group up to my very last. Many of my colleagues in the research group and division have also become friends during this time. For the good times at lunch, over coffee, and after work, I would like to thank especially Zahra, Mattias, Sabino, Sarmad, Fredrik, Endre, Alvin, Albert, Maxi, Remi, Rita, and Sten-Elling, but also anyone else that I might have missed in this list.

Lastly, I would like to express my deepest gratitude to my family for all the opportunities and support over the years that have led up to this point.

Thank you!
Constantin Cronrath
GÖTEBORG
April 2023

Acronyms

AI:	Artificial Intelligence
CPS:	Cyber-Physical System
DT:	Digital Twin
ENMPC:	Economic Non-linear Model Predictive Control
HRC:	Human-Robot Collaboration
IIoT:	Industrial Internet of Things
IoT:	Internet of Things
MDP:	Markov Decision Process
ML:	Machine Learning
MPC:	Model Predictive Control
RL:	Reinforcement Learning
SUT:	System Under Test
UKF:	Unscented Kalman Filter

Contents

Part I

Overview

CHAPTER 1

Introduction

This chapter outlines how current societal mega-trends necessitate and enable manufacturing automation to become more intelligent. While the model-based digital twin approach and the data-driven reinforcement learning approach make advances towards intelligent manufacturing automation, it is argued in this chapter that a combination of both may prove more fruitful. Therefore, the research strategy employed in this thesis aims to extend either approach by principles of the other. To that end, the main outcomes of the research and the thesis itself are outlined.

1.1 Research Motivation

Manufacturing systems are, broadly speaking, the arrangement of the three factors of production – man, material, and machine – to satisfy demands of the market. Each of these three factors on the supply side of the market, as well as customer demands on the other, are exposed to societal mega-trends of fundamental significance for the design of manufacturing systems. On the one hand, market demands have shifted since the 1980s from mass production and competition via cost, towards mass customization and competition via product variety [1]. Moreover, the customization trend extends to personalized production and a shift towards business-to-customer

business models, such that manufactures can better meet individual demands of their customers. This change in the external environment requires manufacturing systems to become more flexible, while ensuring competitive speed, quality and cost [2]. On the other hand, Fig. 1.1 depicts one of the changes affecting the internal factors of production: the working age population is peaking within a generation (India), stagnating (United States), about to decline (China), or declining (Europe) in major industrial geographic regions [3]. While some of this demographic challenge may be mitigated by a higher labour force participation and migration, the implication for manufacturing systems design is twofold: the work environment must become more attractive to the shrinking number of (increasingly better educated) job seekers, and place lower physical demands on the aging workforce [4]. At the same time, sustainability demands an efficient use of resources, such as material and energy. Arguably, the most prominent driver of change in manufacturing system design in the last decade has been, however, the advent of internet technology in the manufacturing context. The advances in computing and networking technology have enabled the digitization of manufacturing machines, heralding in a technological shift, which has been coined the fourth industrial revolution, also known as Industry 4.0 [5]. Key features of Industry 4.0 are the industrial internet of things, data analytics, and smart manufacturing. In this technological shift, the manufacturing system is transformed into a Cyber-Physical System (CPS) [6], in which the physical and digital world blend into one. Taken together, these trends fundamentally change the demands on – and possibilities in – the design of manufacturing systems.

Manufacturing automation must become more collaborative and intelligent, though, to be capable of meeting its new requirements [7]–[9]. Firstly, automation needs to become more intelligent to achieve the flexibility required in quickly varying manufacturing processes for personalized production. Secondly, automation takes a crucial role in addressing the emerging requirements by the demographic structure through automating physically demanding and tedious work steps that have been previously beyond the reach of automation. For that, automation needs to become more collaborative and – most importantly – more intelligent. Collaborative robots, for instance, may allow for more flexible manufacturing, and physically less demanding operations for their human collaborators, but may require more intelligent features to ensure – for example – safety in that [7], [10]. To that end, the European Union introduced a concept called Industry 5.0 to complement the emerging technological enabler, Industry 4.0, with guiding values [11], [12]. Industry 5.0 strives for human-centric automation, that puts the needs of the worker first, and sustainable

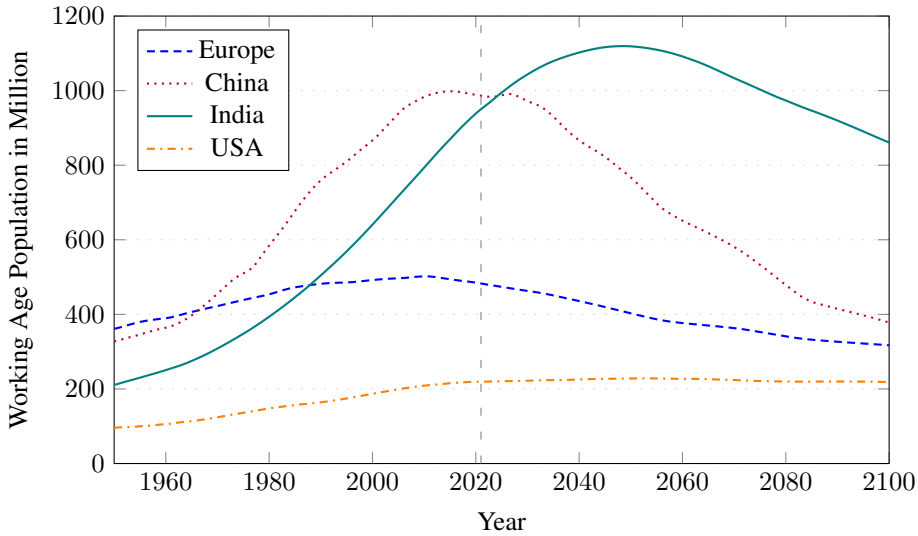


Figure 1.1: Historic and estimated size of the working age population (15-64 years) of selected industrial regions. Data obtained from the United Nations Population Division [3]. Projections after 2021 are based on the United Nations Medium (i.e. baseline) Scenario.

and resilient manufacturing systems, which can cope flexibly with disruptions [11], [13]. By that, intelligent manufacturing automation may contribute to meeting these current and emerging requirements.

Defining intelligence in the automation context is, however, not trivial. Indeed, researchers first circumvented the problem of defining intelligence of machines in the early years of their scientific endeavours. Alan Turing states, for instance, in his seminal paper *"Computing Machinery and Intelligence"* [14] in 1950, that a definition of machine intelligence in the normal use of the word is dangerous and instead proposes a procedure for testing machine intelligence that is known nowadays as the Turing Test. The Turing Test is passed if a machine can convincingly imitate a human in written communication, such that a human interrogator is unable to distinguish machine and human. Similarly to A. Turing, Claude E. Shannon proposes in the same year (1950) to use the ability of playing chess as the benchmark for assessing *"thinking"* skills of machines [15]. Even when today's popular term *"Artificial Intelligence"* (AI) was coined in a proposal for a *"Summer Research Project on Arti-*

ficial Intelligence" in 1955, its authors, McCarthy, Minsky, Rochester and Shannon, solely circumscribed it by *"every aspect of learning or any other feature of intelligence"* [16]. The problem of defining AI remained unsolved, leading McCarthy and Hayes to ascertain in 1969 [17]:

> *"Since the philosophers have not really come to an agreement in 2500 years, it might seem that artificial intelligence is in a rather hopeless state if it is to depend on concrete enough information out of philosophy to write computer programs."*

Nevertheless, various researchers have put forward definitions of artificial intelligence, that can be broadly categorized into the four combinations of: thinking versus acting, and doing so humanly versus rationally [18]. Interestingly, human intelligence has long been seen as the hallmark of intelligence. Indeed, the term "artificial intelligence" may invoke, in some readers, ideas that are commonly attributed to the concept of *strong* Artificial *General* Intelligence, which is to describe human-like abilities in a wide range of tasks stemming from an "actual mind". At the other end of the spectrum lies the, practically more relevant to date, *weak narrow* AI, that solely simulates intelligence in one or another specialized area [18]. In the absence of a more grounded philosophical definition, this thesis employs the notion of weak and narrow AI acting rationally when discussing intelligent automation. It seems important to emphasise here that this notion is agnostic to the methods used in producing intelligent behavior in the automation of manufacturing systems. Indeed, a critical reader might object to the idea of *"intelligent"* automation once specific methods are discussed. On that note, Marvin Minsky observes in his 1961 paper *"Steps toward Artificial Intelligence"* that to him *"'intelligence' seems to denote little more than the complex of performances which we happen to respect, but do not understand"* [19]. This causes the so-called AI effect, which describes that once the complex of performances is understood the goalpost for what AI is is moved.

In an attempt to understand AI, M. Minsky, nevertheless, outlines the parts of intelligence in [19]. To him, intelligence breaks down into the epistemological part, which is the representation or model of the world, and the heuristic part, which are the mechanisms that solve the following five problems. First, the problem of search constitutes finding an *optimal* answer. Second, abstract representations of sensory input must be formed in the problem of pattern recognition. Third is the problem of learning from reinforcement feedback of the world. Fourth, complex tasks must be solved through following a sequence of actions that are a solution to the problem of planning. Fifth, general statements, which go beyond any recorded experience, must

be inferred in the problem of induction. While this *model-based* approach to AI was popular throughout the 20th century, we argue that its most likely heir is the concept of the Digital Twin (DT).

Similar to the early years of AI, the definition of a digital twin is undergoing frequent adjustments since its conception [20] in 2002. In Chapter 3, we work out a definition of digital twins in system theoretic terms, but refer to an intuitive, early description in the context of manufacturing systems by Boschert and Rosen [21] from 2016, at this point that can be summarized as:

> The digital twin is a linked collection of relevant digital artefacts, including data and models of suitable fidelity that evolve along the real system and are used to optimize operation and service.

Whereas the epistemological (model) part is central in this definition, the heuristic part becomes apparent at closer inspection of the description of the evolutionary and optimal operational aspects. The digital twin concept has become increasingly popular (see Figure 1.2) also in the manufacturing context, since a digital twin is a possible outcome of the virtual commissioning process of manufacturing systems [22]. Virtual commissioning generally refers to the testing and validation of control logic in simulation. To that end, expressive simulation models are built during the design and development phase of the manufacturing system. These models may be repurposed into digital twins of the system during its operational life-time. As such, digital twins provide an excellent basis for model-based intelligent automation solutions.

A diametrical trend to the model-based approach is the learning-based approach to AI fueled by recent breakthroughs in deep artificial neural networks in machine learning (ML). Already in 1950, A. Turing wondered [14]:

> *"Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain."*

This approach thus places heavy emphasis on the learning aspect. Similar to the breakdown of AI by Minsky, one can divide the modern field of machine learning into knowledge-based systems and machine learning as such. Machine learning itself further separates into unsupervised, supervised, and reinforcement learning. Of these three, Reinforcement Learning (RL) is the category concerned with acting optimally and thus of prime interest in the context of this thesis. RL has profited from recent developments in deep artificial neural networks leading to the first *deep* RL paper in

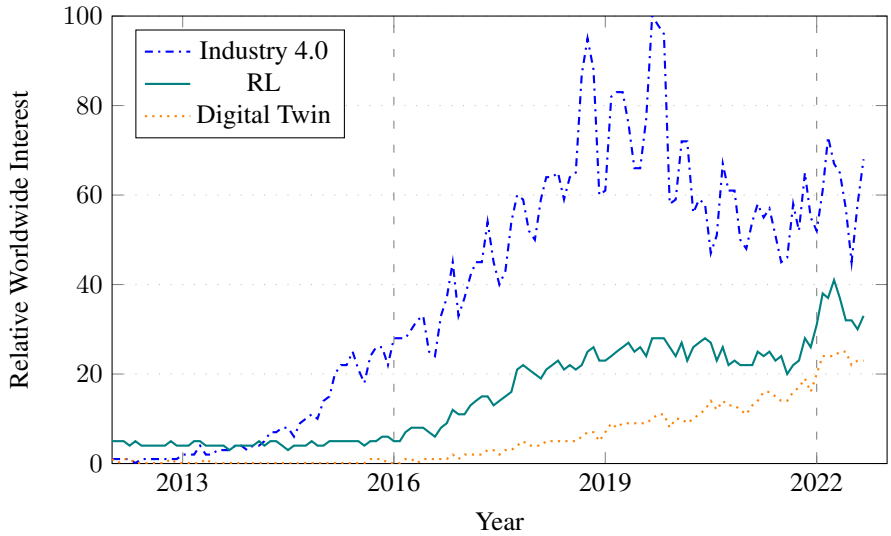


Figure 1.2: Worldwide interest in the three topics: Industry 4.0, digital twins, and reinforcement learning. Data obtained from Google Trends. The y-axis is normalized to the maximum observed interest. Changes have been made to the data collection procedure in January 2016 and 2022.

2013 [23], [24] and the media-effective mastering of the boardgame *Go* in 2016 [25]. This sparked renewed interest in RL as an AI method (see Figure 1.2). While Turing believed that reinforcement through *"the use of punishment and rewards can at best be part of the teaching process"* [14], this learning-based approach is considered a viable alternative to the model-based approach to intelligent automation.

However, it is the author's conviction that neither the model-based digital twin approach, nor the learning-based RL approach will succeed on their own, because of the needs and challenges imposed on intelligent automation by the manufacturing application context. These needs and challenges include, but are not limited to:

1. **The challenge of model-system mismatch:** digital twins strive to built models of suitably high fidelity to optimize the operation of the system. Still, these digital models always remain an abstract representation of the world. This fundamental epistemological challenge is denoted by the model-system mismatch. As its results, any control action derived in the digital twin may be optimally rational with respect to the model but not with respect to the system.

2. **The need for real-time decisions:** the pace of production in manufacturing systems is ultimately governed by customer demand and the capacity of the system. Often, this so-called tact time is only around a few minutes in e.g. automotive manufacturing. Within each tact, multiple process steps may need to be performed to further the product in the manufacturing process. This means that the time to come to a control decision may be fairly limited. Lengthy optimizations over computationally intensive digital models, may therefore not be feasible. On the contrary, a good enough solution in time may well be preferred over an optimal one too late. This real-time requirement is a challenge especially to the model-based approach to intelligent automation. Taken to its extreme, this requirement may demand distributed decision making. Instead of optimizing a monolithic model of the system on a powerful cloud server, a decomposition into smaller optimization problems distributed throughout the system may prove more manageable to save on communication and optimization time. Yet, this may still require quick learning-based inference of the solution or parts thereof to meet time constraints.

3. **The need for explainability and safety:** automation solutions in manufacturing systems need to make decisions that may have consequences for the human operators in its proximity, the system, or the economy of the company. Random or erratic decision making is thus difficult to justify. Instead, all automated decisions should ideally be easily explainable, such that a human could follow the steps that were taken to arrive at a particular decision. Moreover, when human safety is concerned, one ought not to leave the decision to chance, but ought to take sufficient steps to minimize any risks. This is a challenge for learning-based methods that frequently use blackbox function approximators, such as neural networks, to identify and leverage correlations in data. Moreover, this is especially true for reinforcement learning methods that need to take untried actions once in a while to learn better decisions. During this exploration, one ideally would wish for some estimate of the outcome of the decision, such that safety in the system is maintained at all times. Popular learning-based methods, however, often lack such mechanisms.

4. **The challenge of small data:** in the manufacturing context, each data sample may correspond to a single instance of the product. Depending on the product's production rate, it may thus take considerable time to accumulate enough data for a sufficient training set required by learning-based methods. Generally speaking, the more complex the learning task, the more data is needed for

learning. Beyond this rule of thumb, learning-based methods benefit from diverse data distributions that make the learned model robust. However, in the manufacturing context, machines are often tightly controlled within a small operating window. If the manufacturing process is well-tuned, all data may, thus, come from a rather small region of the possible operating conditions. This may prove detrimental if the learned approximator is then queried in a region that requires significant extrapolation beyond its recorded experience.

To summarize, the model-based approach to intelligent automation has the advantage of incorporating prior knowledge from engineers in form of a digital twin, which describes the operating conditions in a possibly larger region than the one the process is usually operated in, and helps making good and explainable decisions starting from the first data sample. The model-based approach does suffer, however, from the fact that high-fidelity models that try to minimize the model-system mismatch are computationally expensive to simulate and optimize. Contrary to that, the learning-based approach has the possibility to learn arbitrarily good and quick to evaluate approximations given a sufficient amount of data. The learning approach may perform poorly during its training phase, though, and may lack the desired explainability and safety of model-based approaches. It seems a combination of both approaches may prove fruitful in the context of intelligent automation to meet the new requirements imposed by the ongoing societal trends.

1.2 Research Gap

To recap, intelligent automation aims to unlock remaining performance improvement potentials and meet current design requirements in cyber-physical manufacturing systems. Two diametrical trends in automation try to make progress on that front: model-based digital twins, and learning-based AI (see also [8]). In the thesis at hand, it is hypothesized that a combination of digital twins and learning may prove fruitful.

Digital twins address the issue by striving for high-fidelity, minimal-mismatch digital models built by experts and are regarded as a main driver in smart and intelligent manufacturing [26], [27]. Indeed, Zhuang *et al.* [28] conceptualize a proactive strategy as the highest form in the evolution of shop-floor production management and control strategies. Such a proactive strategy *"can even drive and control physical entities in the real space based on CPS' self-adaptive and self-reconfiguration functions* [through] *AI and digital twin technology."* Also, Wang *et al.* [29] identify digital twins and AI as key technologies of smart and intelligent manufacturing, and

call attention to further development of these key technologies. On closer inspection, it becomes apparent that many of the postulated properties of digital twins require in themselves data-driven learning. Tao and Zhang [30] outline *"iterative optimization, self-learning, self-adaptation, real-time interaction and convergence"* as key technologies for digital twins of shop floors. Along those lines, Tao *et al.* [31] raise the use of deep learning in digital-twin-driven manufacturing as a *"huge"* research challenge. Zhou *et al.* [32] assign to the digital-twin *"knowledge-based intelligent skills"* in the control of a manufacturing cell as a key feature, but also list those as a challenge ahead for researchers. Nevertheless, Zhou *et al.* place hope in deep-learning to address these challenges in the future. Similarly, Zhong *et al.* [8] point out the combination of human domain expertise and data-enabled machine learning as a pathway to intelligent manufacturing in the context of the fourth industrial revolution.

On the other hand, reinforcement learning is a particularly popular data-driven learning method in AI for control. In the context of smart manufacturing automation, Lu *et al.* [33] emphasize the need to progress on networked self-organizing manufacturing system automation in which reinforcement learning is highlighted as a means to achieve *"advanced cognitive capabilities"*. However, in contemporary RL, the system is often approximated by deep neural networks, which require big data sets to reach control performances comparable to established model-based methods. Accordingly, Kuhnle *et al.* [34] point out the combination of model-based and model-free (reinforcement) learning methods as a promising research direction for control of manufacturing systems. Similarly, Arents and Greitans [35] emphasize the need to combine expert knowledge and reinforcement learning in the manufacturing context to arrive at smart industrial robot control. However, the authors of that paper refer to expert knowledge extracted by sim-to-real [36] and imitation learning [37] methods. The sim-to-real approach is also identified by Li *et al.* [38], [39], Li *et al.* [39], and Panzer and Bender [40] as a promising research avenue. These authors furthermore highlight digital twins as a means for reducing the model-system mismatch or sim-to-real gap inherent to this approach. In the research of Li *et al.* [38], the digital twin is furthermore used as a safety supervisor, while the learned deep neural network policy executes the task in the physical system. This approach may prevent catastrophic failures but does not necessarily ensure "expert-level" performance in all circumstances. The incorporation of known process constraints in the RL formulation is moreover mentioned frequently as a future research direction in the manufacturing context [40]–[42].

In summary, both digital twins and reinforcement learning are seen as promising

approaches to intelligent manufacturing automation, while multiple recent publications call for combinations of both approaches. This thesis is positioned, therefore, at the intersection of digital twins and learning-based artificial intelligence in the context of intelligent manufacturing automation.

1.3 Research Questions

Two research streams towards intelligent manufacturing automation are observed: (1) the model-based digital twin approach, and (2) the data-driven learning approach. As argued in Section 1.1, a combination of both approaches is believed to be more promising though. This thesis aims to explore this identified research gap at the intersection of both approaches. An advance into the gap from either side seems hence natural. Figure 1.3 illustrates the preceding sections and places the following two research questions into context:

RQ1: *How can machine learning be used to mitigate the model-system mismatch in digital twins, while minimally altering the provided solution?*
This first research question assumes an intelligent automation solution is provided already in form of a packaged digital twin. It is thus of interest how machine learning could be used to mitigate the challenges of this model-based approach – while minimally altering the provided solution. In that, it is furthermore assumed that the provided solution is capable of making decisions in the time frames required by the physical system. *RQ1* thus focuses solely on the use of learning to mitigate the model-system mismatch challenge.

RQ2: *How can prior model-based knowledge be introduced in reinforcement learning to improve sample-efficiency, explainability, and safety of the learned control policy?*
This second research question assumes the learning-based approach to intelligent manufacturing automation has been selected and its challenges need to be overcome. In particular, it is of interest how prior knowledge about the system and its desired behavior, which is mostly available to a system engineer, can be incorporated into the learning mechanism. The intention here is to improve explainability and safety of the learned control policy, as well as sample efficiency of the learning process itself. *RQ2* thus focuses on incorporating model-based knowledge into RL.

While it cannot be hoped for that an advance from both sides, each guided by the

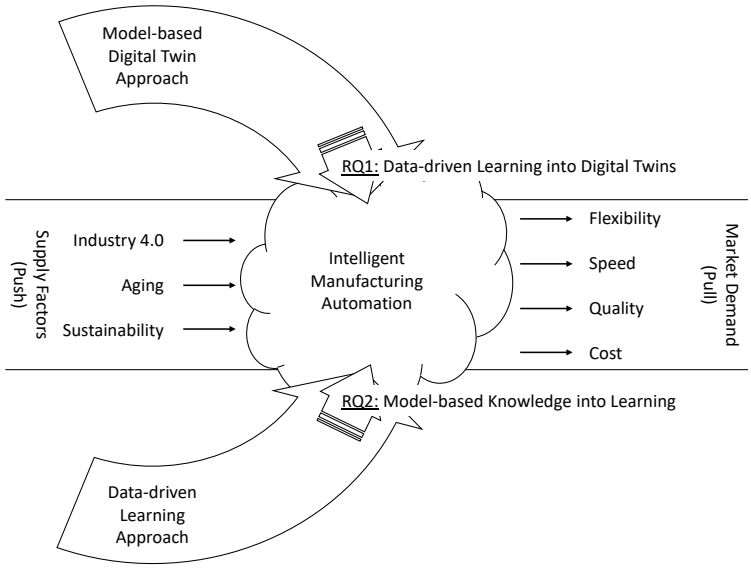


Figure 1.3: An illustration of the research context, the research gap, and the two research questions of the thesis. Intelligent manufacturing automation is influenced by the drivers of change on the side of the supply factors as well as on the market demand side. Two research streams approach the topic of intelligent manufacturing automation from different angles. This thesis is positioned at the intersection of both approaches. The two guiding research questions of this thesis aim to investigate this intersection with either one of the approaches as point of departure.

two research questions above, will result in a unified comprehensive framework for intelligent manufacturing automation, it is nevertheless believed by the author that the corresponding research endeavor may be fruitful.

1.4 Research Approach

This thesis is the outcome of a five-year endeavour into a technical field. Although, one could view the research process as a sequence of structured activities leading to insights novel to the field, in hindsight it turned out to be less predictable, goal-directed, and methodical than expected, but rather exploratory, serendipitous, and iterative. Bell, Bryman and Harley [43] describe this as the “messiness” of business

research (which includes organizational and operational research and thus extends to research on automation). On a more philosophical note, A. Chalmers concludes in his book *What is this thing called Science?* [44]:

> *"I reaffirm that there is no general account of science and scientific method to be had that applies to all sciences at all historical stages in their development. [...] Although it is true that scientist themselves are the practitioners best able to conduct science and are not in need of advice from philosophers, scientist are not particularly adept at taking a step back from their work and describing and characterising the nature of that work."*

Without such a general scientific method, it seems all that philosophy has to offer are mental crouches that aide in making sense of one aspect or another of the research process. To that end, we may ask ourselves what the nature of intelligent manufacturing automation is, i.e. its ontology, and what the nature of the scientific knowledge about it is that we can hope to obtain from our research, i.e. its epistemology.

There are two major ontological perspectives one may take when contemplating intelligent manufacturing automation. Ontology itself refers to the general nature of the entity in the research focus. From the ontological perspective of objectivism, it is assumed that the entity exists separately of the social circumstances and independent from human perception or action. It cannot be influenced by social actors. Constructionism, on the contrary, emanates from the assumption that the entity of research is defined by social actors and is continuously revised by their interactions. The entity will thus constantly change, and research on it will always be biased by the researcher's subjective perception. Whether science should be understood from the objectivist or constructionist, or repsectively the positivist or constructionist (in the epsitemological sense) position cannot be argued for conclusively [45]. In the author's opinion, intelligent manufacturing automation fits more a constructionist point of view, since automation design is governed by user requirements, whereas the idea of a one and only "true" intelligent manufacturing automation design seems absurd.

Our ontological considerations give rise to possible epistemological perspectives. Epistemology is the question about the value and truth of research and knowledge in a certain discipline. Positivism values knowledge that was gathered through the methods of natural science and can be experienced by the human senses. Therefore, knowledge has to always be based on facts, be objective and testable anytime. This goes hand in hand with the ontological perspective of objectivism. Realism acknowledges elements in a theory that are not explainable or observable with cur-

rent scientific possibilities. Science, in this view, is a simplification of observable phenomena, but overlooks the underlying generating structures of these phenomena. Interpretivism departs from realism and argues that a theory is always subjective, meaning that social phenomena are interpreted by social actors according to their background and environment. Hence, research is always biased by the experimental settings and the researcher's perception. Given our definition of artificial intelligence as acting rationally, research and knowledge in intelligent manufacturing automation seems to fit most naturally the ontological perspective of objectivism, though. It should be noted here that one need not subscribe to a single ontology and epistemology within a particular field of study. Indeed, a change of one's position may open up new research avenues [43].

Our ontological and epistemological assumptions are of practical relevance, because these assumptions inform which research methodology would most likely produces valuable knowledge about the entity of research [43]. In a mostly deductive research approach, hypotheses are derived from theory to be tested against reality. In a mostly inductive approach, theory is formed from generalizations of made observations. However, any research approach is seldom purely one or the other. Indeed, one may argue that the research presented within this thesis is both. Intelligent manufacturing automation is a comparably nascent field of study. When taking a dynamic perspective on science similar to the one put forward by T. Kuhn [44], it may be argued that this field of study is in a pre-paradigmatic phase. In such phases, researchers propose competing paradigms that eventually get evaluated against each other and from which the one new dominating paradigm emerges. In an area where relatively little prior research has been done, a more exploratory, that is a qualitative, research strategy may be more suitable than a quantitative one. Such a – relatively unstructured – approach may generate new theories and hypotheses that can then be tested quantitatively [43]. Undeniably, the research presented in this thesis has been part of a broad exploratory research endeavour and employs for most parts a more qualitative case study research design. This is the inductive element of the presented research. Within each study, however, a hypothetico-deductive method [46] has been used. As such, the research approach of this thesis fits elements of the DRM – design research method by Blessing and Chakrabarti [47] – even though its proponents also need to admit a lack of consensus regarding what the "right" research methodology for design research is.

Table 1.1 provides an overview of the appended research papers. The six papers cover multiple application areas and system types. In the DRM framework of Bless-

Table 1.1: A Categorization of the Appended Research Papers.

Paper	Application	Focus	System Type	RQ
A	Sheet Metal Assembly	Quality Assurance	Static	1
B	Sheet Metal Assembly	Quality Assurance	Contextual	1
C	Sheet Metal Assembly	Quality Assurance	Dynamic (continuous)	1
D	Collaborative Robotics	Safety Falsification	Dynamic (discrete)	2
E	Collaborative Robotics	Energy Optimization	Dynamic (hybrid)	2

ing and Chakrabarti [47], research is classified into four phases: (1) the research clarification, (2) the descriptive study I, (3) the prescriptive study, and (4) the descriptive study II. This first part of the thesis may be understood as part of the research clarification in which the current understanding and expectations are clarified. Paper A may be regarded mainly as a descriptive study I, due to its comparison of established methods. Although not the first in the temporal sequence, Paper A lends itself to be the first appended paper, therefore. Papers B through E are more prescriptive in nature. In them, the purpose is rather found in the next-step conceptual development of intelligent manufacturing automation. This is reflective of the current phase that the field of study is in according to the author's conviction.

1.5 Contributions

In the research presented in this thesis, we make the following contributions with regard to **RQ1** – *How can machine learning be used to mitigate the model-system mismatch in digital twins, while minimally altering the provided solution?*:

C1: In Paper A and Paper B, we evaluate machine learning to mitigate mismatches in the context of direct input adaptation. We show that learning in direct input adaptation is of limited usefulness in the systemic mismatch case (Paper A). In such settings, blackbox optimization algorithms that leverage properties of the problem are more useful in terms of sample-efficiency, performance within a given budget, and regret. Our research indicates, however, that learning might

have some (limited) merit in individualized production control (Paper B).

C2: In Paper C, we present a mitigation method for drifting mismatches in individualized production control. To that end, the mismatch problem is re-formulated such that it can be solved by a state observation algorithm. Our method is thus a suitable approach for mitigating mismatches through modifier learning – a strategy in which a model of the mismatch is estimated.

C3: In Paper E, we show the connection between adaptive model predictive control and the digital twin concept. This method uses reinforcement learning to adjust parameters within the (digital) model to achieve convergence between observed performance and performance predicted by the model. Our research introduces this method as a pathway to realizing the vision of the digital twin concept.

Our contributions related to **RQ2** – *How can prior model-based knowledge be introduced in reinforcement learning to improve sample-efficiency, explainability, and safety of the learned control policy?* are:

C4: In Paper D, we develop a principled method for incorporating prior knowledge in the form of automata specifications into reinforcement learning. In contrast to other comparable approaches, our method optimally balances specification and environment rewards trough Lagrangian optimization.

C5: In Paper E, we apply a recent generic adaptive model predictive control method into a new, specific and important application area. This method uses economic non-linear model predictive controllers as model class for function approximation in reinforcement learning. We show the benefits of introducing rich prior model-based knowledge in this form in the context of energy optimization.

1.6 Thesis Outline and Scope

The remainder of Part I of this thesis is structured as follows:

- In **Chapter 2:** *Applications*, three application cases in this thesis are introduced: geometry assurance in sheet metal joining, human-robot collaboration, and energy optimization of robots. At the end of this chapter the reader should have an intuitive understanding of recent developments on the three factors of

production: digital twins and Industry 4.0 in manufacturing (machine), collaborative robotics and safety (man), and sustainable manufacturing (material).

- In **Chapter 3:** *Digital Twins*, the published literature is analysed with respect to characteristics of digital twins. It is argued that three functions are required to achieve the "twinning" key characteristic of digital twins: state observation, system identification, and optimal control. This thesis exclusively focuses on optimal control under poorly identified systems. At the end of this chapter the reader should have an intuitive understanding of what a digital twin is and how it needs (reinforcement) learning to mitigate the model-system mismatch.
- In **Chapter 4:** *Reinforcement Learning* is described. A particular focus of this chapter is on model-based RL, sim-to-real learning, and learning with *"expert"* controllers, such as established model-based control methods. Research challenges are highlighted. At the end of this chapter the reader should have an intuitive understanding of the shortcomings of pure reinforcement learning.
- In **Chapter 5:** *Summary of Appended Papers*, our attempts of bridging the gap between digital twins and reinforcement learning in the context of intelligent automation are summarized.
- In **Chapter 6:** *Conclusions and Future Work*, we provide some of our insights so far and point out possible avenues for further research.

Topics that are excluded from this thesis, include: modelling and simulation of systems as digital twins; distributed networking aspects (e.g. computation in the cloud, fog, edge, etc.); hybrid or hierarchical control architectures (for instance model-based control and learning on two different levels); intelligent automation as acting humanly.

CHAPTER 2

Applications

Three applications are introduced in this chapter. Each application illustrates current developments on one of the factors of production: Section 2.1 centres around Industry 4.0 and digital twins in manufacturing (machine), Section 2.2 introduces collaborative robotics and safety in interaction with humans (man), and Section 2.3 sketches out energy-efficient control in sustainable manufacturing (material). A particular focus in their descriptions is on intelligent automation aspects to provide context for the subsequent chapters.

2.1 Smart Assembly 4.0

The arrival of internet technology to the manufacturing context has been named the fourth industrial revolution. Internet-enabled sensors can now continuously capture data on the factory floor and stream that data to the cloud for sophisticated data processing and analysis. This section outlines such a data processing concept in the context of sheet metal assembly and illustrates the challenges for a model-based approach to intelligent automation in that.

In an assembly process, two or more parts are joined together. This is a ubiquitous task in manufacturing systems. In the automotive industry, for instance, parts of

the vehicle body are stamped from sheet metal rolls and then joined by e.g. spot welding to form the exterior of the vehicle. Variations in the parts as well as in the manufacturing equipment propagate through the assembly process, and affect the resulting geometry of the product. The assembly process has thus major influence on the aesthetic quality, functionality and safety of the final product. The goal of geometry assurance is to reduce the effect of these variations and thereby improve quality. This in turn reduces cost, since geometry related issues are a major driver for re-work and delays in the production process.

By using virtual tools, geometry related issues can be detected and mitigated earlier in the product life cycle. For example, Söderberg *et al.* [48] outline the use of computationally intensive simulation models, based on e.g. the finite element method, in the product design phase to determine product tolerances and requirements and robust locating schemes. These simulation models are built from first principles, historical experience, and historical data. Statistical variation simulation can further enhance the reliability of these models in the decision making process. Moreover, these virtual models of the product are useful in designing the assembly process, as well as in the design of the quality inspection process, and the off-line programming of the manufacturing equipment. Indeed, virtual geometry assurance tools enable the detection of quality issues early in the design process rather than during production, which saves money and time Söderberg *et al.* [48].

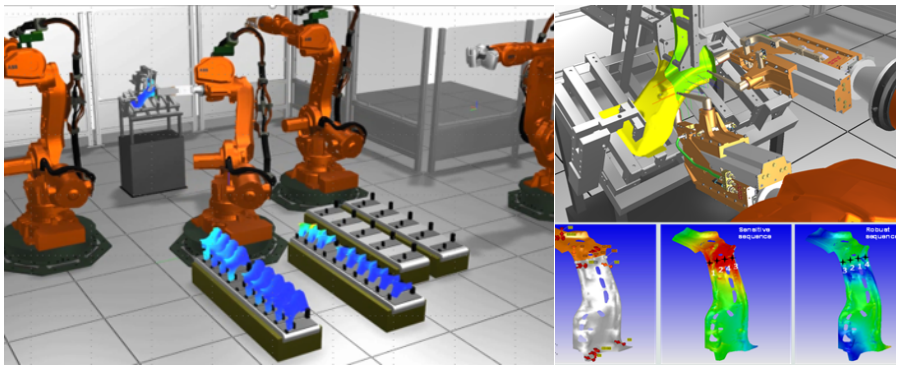
Beyond that, virtual models of the product and its assembly process, that were developed during the design phases, can be extended into digital twins for the production phase to unlock further improvement potentials through individualized process control [48]. This is because in a digitized and internet-of-things-enabled factory, each individual part may be measured by inline scanners, for instance, and its assembly process optimized in real-time. Thereby, individual part variations can be accounted for through selective part matching, virtual trimming of fixture locators and joining sequence optimization. Digital twins are seen, accordingly, as one of the most important development areas by practitioners in the geometry assurance field [49]. The development of a digital twin for geometry assurance in sheet metal assembly has been the goal of the Smart Assembly 4.0 project, which is outlined in Söderberg *et al.* [48] and initiated the research presented in this thesis. Its vision was *"the autonomous, self-optimizing robotized assembly factory, which maximises quality and throughput, maintaining flexibility by a sensing, thinking and acting strategy"* by means of digital twins. To that end, Bohlin *et al.* [50] proposed a modular system architecture with decentralized asynchronous data streams between physical and

virtual space, as well as between digital platforms. Data capturing for analysis and real-time optimization in the digital twin can happen then through inline scanning in the physical system. However, the data capturing by geometry inspection must be sufficiently quick to keep up with the tact time of the assembly process. It is thus crucial to capture the *right* data rather than the *most* data by leveraging expert domain knowledge. This enables a certain speed in the computational analysis and keeps data storage volumes and costs down [51]. The measured data is then used to update the digital twins of products and processes. Virtual parts can subsequently be sorted and matched to reduce geometry variations of the assembly. Furthermore, quality can be improved through such a self-adjusting assembly concept without tightening tolerances.

However, such an individualized production must be enabled by intelligent manufacturing automation capabilities. For instance, the weld sequencing and other geometry assurance measures on the individual part level result in a unique joining scheme for each assembly. This requires an online update of the robot control programs to optimally coordinate the welds between robots. In that, detailed simulation models of the robots are needed on the one hand, and on the other hand algorithms that can arrive at a feasible solution quickly [50].

As a test bed for such algorithms, a use case has been developed, which is depicted in Figure 2.1. The case is a sub-assembly of a reinforcement and torsional stiffness bracket of a car body. The virtual geometry of the parts is given by their measured point clouds of about 2.5 thousand points each. Each of the fixture's twelve locators are adjustable along their axes, and the two parts are joined by seven spot welds. The overall objective is to minimize the root mean square error of the resulting assembly from the nominal assembly by adapting the assembly process for each individual part geometry and measurements fed back to the digital twin.

The difficulties in realizing a digital twin for the automated assembly process lay in the sensitivity of the process and the time constraints imposed by the customer tact. According to Wärmefjord *et al.* [52], several factors affect the geometric quality of sheet metal assemblies, such as: fixture deviations; clamping force and clamping stiffness deviations; joining point, force, and tool variations; mechanical deterioration; friction; part geometry variations; and material property variations. Even highly sophisticated Finite Element Analyses of non-rigid deformations, thus, do not fully capture the complete real physical behavior [52]. Such uncertainties in the manufacturing process and model simplifications, needed to compute model-based solutions quickly, lead to a mismatch between the digital twin models and the physical sys-



(a) Spot welding cell (b) Fixture detail

Figure 2.1: The use case of the Smart Assembly 4.0 project. Incoming parts are scanned for their geometric variation (as seen color coded in (a)), matched individually with a part of the other type, positioned in an adjustable fixture and joined by robotic spot welding (see upper picture in (b)). The resulting assembly geometry is then scanned and fed back to the digital twin. Locator adjustments and joining sequence affect geometric quality as detailed in the lower part of (b). Therefore, the robot and fixture control programs are continuously adapted based on the individual digital twins of the parts and of the process.

tem [50]. A major focus in the Smart Assembly project is thus on improving models of the manufacturing process (e.g. simulating of robot dress packs), extended sensing of the process to feed said models (e.g. scanning of parts), and deciding on optimal process inputs (e.g. locator adjustments). The result of these optimizations, however, hinges on the quality of the model and the sensed data.

In summary, virtual models of the assembly process combined with real-time data from the factory floor may unlock the improvement potentials of individualized production control, but may potentially suffer from the model-system mismatch. One possible mitigation strategy against the model-system mismatch is to incorporate uncertainties explicitly into the model and make the control optimization robust against them (e.g. variational analysis). A second strategy is to leverage measurements from the system to adapt the digital twin through data-driven learning methods. This second strategy is explored in the previously described Smart Assembly 4.0 context in [53]–[56].

2.2 Collaborative Robots

The workforce in major industrial regions is aging and/or shrinking. This requires the automation of previously manual tasks to maintain and grow productivity on the national level. This often means that automation must enter into unstructured environments that are shared with humans. New "intelligent" features are needed in automation therefore.

Industrial robots are the workhorses of factory automation. Traditionally, robots have been tightly fenced off in industry to protect the human operators from physical harm. Common safety measures have been fences, gates and safety switches that ensure a physical separation of the robot's and human's workspaces when in operation. However, if a robot has the capabilities to share its workspace with a human, it can be considered to be collaborative [57]. These capabilities need not be built into the robot as such. A collaborative robot can also be a conventional industrial robot, which was equipped with external sensors, such as cameras or lidars, combined with appropriate control logic. Commercial retrofitting solutions are available from most major robot manufacturers [57]. If the robot is, furthermore, equipped with appropriate inbuilt sensors, physical collaboration of robot and human become possible. For instance, robot and human may join forces in manipulating large objects (see e.g. [58]) or flexibly share tasks between each other [57]. In such collaborative settings, communication between robot and human through voice, gestures, graphical user interfaces, or physical contact (force feedback, guiding etc.) may be needed [57], [59].

Collaborative robots may partially address the present challenges in manufacturing system design. Robots that do not require to be fenced off and are equipped with additional interactive capabilities can have a smaller footprint on the factory floor and increase flexibility of the manufacturing process [7]. Through intuitive programming interfaces, automation of process steps becomes more adaptable and accessible also for smaller enterprises without in-depth expertise in robotic programming [7]. Combined with advanced sensing capabilities, it furthermore enables automation in small volume manufacturing, including tasks that have been considered commonly manual work, such as material logistics, handling and feeding [7]. In that, robots often provide power, accuracy and repeatability, while the human provides flexibility and cognitive adaptability (see e.g. [7], [10], [57]). Physical demands on human operators may be therewith lowered and workplace attractiveness increased [10], [57], for instance by providing a sense of purpose to the human in a human robot interaction if the human's abilities are needed to execute the task [60].

Making collaborative robots more intelligent may generate additional improve-

ment potential. A basic form of collaborative robots are industrial robots, retrofitted with external sensors and control logic akin an emergency stop function when a human is in close vicinity. Although a physical fence is no longer required in this context, this relative lack of intelligence demands industrial robots to work in virtually fenced and safeguarded zones to ensure safety of the worker [10]. Without any more autonomous and cognitive abilities, this restricts the scope of their application in industry, since close Human Robot Collaboration (HRC) would frequently induce safety stops and thus hamper productivity [9], [57], [61]. Ideally, however, the collaboration between human and robot should be responsive in the sense that the robot responds to human actions instantaneously [10], [58]. The more *"intelligent"* capabilities needed for such a responsive collaboration may be achieved by: a combination of reactive behavior-based control (direct mapping from sensory input to control actions) and sense-think-act control architectures, which derive control actions from planning over a model (see e.g. [59]); learning (see e.g. [57], [59]) and adaptation [60]; or digital twins, which provide awareness of its environment to the collaborative robot [10]. Indeed, better environment models and predictive capabilities are said to enable flexible automation [9] and more effective safety in human robot collaboration [62].

Safety in human robot collaboration is crucial for the deployment in industry. Indeed, automation that is *perceived* as unsafe, for instance because of unpredictable or erratic movements, may reduce psychological safety of the worker, and cause discomfort, stress and lower productivity (see e.g. [10], [62], [63]). Trust in the automation solution can be established for example through legible motions that communicate intent to the human collaborator [64], [65]. Furthermore, suitable safety measures (see [62] for a comprehensive overview) must have been implemented and verified [66], [67]. As a matter of fact, the commissioning of HRC solutions is a time-consuming task and a major obstacle towards their deployment in industry [7], [9]. Moreover, risk assessments must be repeated when the collaborative robot is used in a new way, which becomes increasingly difficult for learning systems [61].

While traditional risk assessment methods may prove impractical, more automated analysis methods may facilitate the uptake of flexible collaborative robots in industry. Traditional methods are mostly based on human reasoning and simple tools like checklists [66], [68]. However, these manual methods may neither be sufficient nor scalable enough for complex system such as intelligent HRC systems [69]. Automated analysis methods categorize broadly into formal verification and simulation-based testing. Formal verification requires formal models of the system, such as au-

tomata, that can be checked against their specifications. Formal verification methods can provide safety guarantees with respect to the formal model and have been applied in the context of cyber-physical systems [70]–[74]. For complex systems that require safety analysis on a more detailed level, for instance because an estimate of collision forces is needed, formal methods are often intractable. In such cases, simulation-based testing methods (e.g. [75]–[77]) may be better suited. These methods search the input space of the simulation model for inputs that cause a given specification to be violated or falsified. However, such simulation-based falsification methods can at most provide counterexamples, but no safety guarantees, since the state-space of the simulated system is often infeasible to explore exhaustively [78]. A method to restrict exploration in simulation-based falsification through formal verification is presented in [79]. Nevertheless, automated safety falsification may benefit from dynamically updated digital twins [80] and may help to overcome the safety obstacle in the deployment of intelligent, collaborative robots in industry.

To sum up, collaborative robots may enable automation of previously manual tasks and provide greater flexibility to the manufacturing system than previous automation solutions. "Intelligent" features of the robots are needed to overcome the safety obstacle when collaborating with humans. Digital twins and learning could be a path towards such features and automated safety assessments and has thus motivated the research presented in [81].

2.3 Sustainable Manufacturing

Manufacturing systems produce from limited resources the goods to satisfy our near unlimited desires. As a growing population on a planet with finite resources, a smart and sustainable usage of these resources becomes increasingly imperative. The concept of sustainable manufacturing is not conclusively defined in the literature yet, and is furthermore subjected to the same value-based political debates as sustainability in general. Nevertheless, one of the first definitions of sustainability was put forward by the 1987 UN Brundtland Commission [82] as being *"development which meets the needs of current generations without compromising the ability of future generations to meet their own needs."* Although this definition is not directly applicable in the manufacturing context [83], it provides sufficient context for understanding the 6R approach to sustainable manufacturing: reduce, reuse, recover, redesign, re-manufacture and re-cycle. Of those, a reduction of waste and required resources by zero-defect and resource-efficient manufacturing can be seen as an ob-

vious – although hardly the sole – operational goal in the sustainability dimension of manufacturing systems [83].

To that end, energy-efficient operation of robotic and automatic manufacturing systems (e.g. through on-off control [84]) offer great reduction potential in resource use [85], since robot energy use is significant in many industries [86]. Such approaches offer great potential, since they are largely organizational measures that do not incur high investment costs [87]. In that, the fourth industrial revolution is a potential catalyst in the development towards sustainable manufacturing [83] because of simplified data collection by IoT senors and smart analysis and decision making algorithms [88]. Indeed, smart manufacturing technologies, such as digital twins and intelligent algorithms, have the potential to reduce the energy use in industry by up to $30\,\%$ according to the International Energy Agency (IEA) [89]. This significant energy saving potential has been confirmed in practical case studies of automotive factories [90]–[92]. For instance, a reduction of about $24\,\%$ of the robot energy use has been demonstrated in [93] for an automotive production line.

Nevertheless, an obstacle to achieving the desired energy reduction is the often conflicting criterion of productivity (the number of produced products per unit of time). Productivity frequently holds precedence over energy reduction (see e.g. [94]). Productivity often manifests as hard deadlines in the production system that need to be met in order to avoid production disruptions. Meeting deadlines becomes especially challenging if there exist uncertainties in the production system. Examples are uncertainties in execution times, breakdowns and arrival times of parts or orders. Robustness to such stochastic disturbances conflicts with energy reduction objectives [95]. Highly detailed digital simulation models may thus facilitate the development and testing of new energy-efficiency procedures [83]. Especially online methods, which react to events in dynamic manufacturing systems, are a promising development direction, to improve gains from energy-efficient operations further [96].

To conclude, a sustainable usage of material and resources in manufacturing becomes ever more important. Intelligent automation solutions that can optimize manufacturing processes online using the latest process data seem promising for realizing efficiency potentials. Such a method has thus been developed in [97].

2.4 Opportunities for Intelligent Automation

Man, machine, and material constitute the three factors of production. Societal megatrends affect each one of them. At the same time, market demands have shifted to-

wards requiring personalized production in many industries. Intelligent automation may leverage – and help to meet the challenges of – these ongoing changes. The ongoing digitisation of manufacturing systems enables new automation capabilities, such as individualized production control to improve quality and flexibility, human robot collaboration to meet demands by an aging and/or shrinking workforce, or energy-efficient control of processes to use resources sustainably. Two approaches towards intelligent automation have been identified: the model-based digital twin approach and the data-driven reinforcement learning approach. These are described in the two subsequent chapters.

CHAPTER 3

Digital Twins

The emergence of the internet of things, big data and cloud computing has given rise to the concept of the digital twin in manufacturing. The digital twin is seen as a core enabler for smart and autonomous manufacturing systems [98]. In essence, it is a sufficiently realistic digital model of the product or system, linked by a bidirectional automated data exchange, used for simulation, optimization, and control. At the end of this chapter the reader should have an intuitive understanding of the model-based digital twin approach to intelligent manufacturing automation, and how it needs (reinforcement) learning to mitigate the model-system mismatch.

3.1 Historical Development of the Concept

The idea of the digital twin concept dates back to 2002 when M. Grieves presented a conceptual ideal for product lifecycle management, that *"did have all the elements of the digital twin: real space, virtual space, the link for data flow from real space to virtual space, the link for information flow from virtual space to real space and virtual sub-spaces."* [20]. It took a further 10 years, however, for the term "digital twin" to be coined and for the concept to gain wide-spread attention. In 2012, Shafto *et al.* [99] at NASA defined the term in the context of aeronautics as follows:

> *"A digital twin is an integrated multiphysics, mulitscale simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its corresponding flying twin."*

Subsequently, various researchers put forward slightly differing visions of the digital twin in the manufacturing context. For instance, Boschert and Rosen [21] describe in 2016 the digital twin for manufacturing as a collection of relevant digital artefacts, including data and models of suitable fidelity that evolves along the real system and is used to optimize operation and service.

In 2017, Söderberg *et al.* [48] envisioned a digital twin for the assembly of sheet metal parts. In their concept, the digital twin is built successively throughout the product life cycle. First, a digital model of the product is built from nominal and historical data. This model aids in the design of, for instance, robust locating schemes for the sheet metal assembly. Then, the digital model is used to develop the assembly and quality control process, centred around inline scanning of the sheet metal geometries. The outcome of these product and process design phases are digital models that can be re-purposed in the production phase for real-time optimization and control. In this last phase, the virtual model is updated with individual part geometries for online joining sequence optimization and virtual trimming of fixture locators. Inspection data of the final product is then used to identify errors in the physical process and possibly for capturing unmodeled effects through machine learning. The presented use of the digital twin is intended to reduce quality related design changes and costs.

In the same year, Tao *et al.* [31] outline the benefits of a digital twin in all product life-cycle phases from design to recycling. The digital twin in the manufacturing phase is further detailed in [30]. In this phase, the digital twin consists of four parts: (1) the physical shop-floor, (2) the virtual shop-floor, (3) the shop-floor service system, and (4) the shop-floor digital twin data. In this concept, the physical shop-floor generates data that update the state and the models of the virtual shop-floor. The virtual shop-floor generates control inputs for its physical counterpart. In that, the shop-floor service system acts as platform for various algorithms that adapt the models of the virtual shop-floor and verify control decisions in the virtual shop-floor before they are applied in the physical system. The shop-floor service system is therefore seen as an enabler for better reliability and productivity of the manufacturing system.

In 2021, the International Standard Organisation defines a digital twin for manufacturing in the standard ISO23247 [100] as follows:

> *"A digital twin in manufacturing is a fit for purpose digital representa-*

> *tion of an observable manufacturing element with synchronization between the element and its digital representation."*

Synchronization is described here as bi-directional updates of the physical and digital elements, such that the manufacturing system is constantly optimised based on real-time information from it. The standard is nonrestrictive in the objective of these optimizations and list for instance real-time control, predictive maintenance, and dynamic risk management as possible applications.

Although these digital twin concepts heavily emphasise the control aspect, other authors rather highlight – particularly in the earlier years of the concept – the fidelity of the digital model. For example, Grieves and Vickers [20] revise in 2017 their earlier description of the concept and instead write: *"The digital twin is a set of virtual information constructs that fully describes a potential or actual physical manufactured product from the micro atomic level to the macro geometrical level,* [operated in] *an integrated, multi-domain physics application space."* Also, Zhuang *et al.* [28] summarize the literature available in 2018 by stating that a *"digital twin is a virtual, dynamic model in the virtual world that is fully consistent with its corresponding physical entity in the real world and can simulate its physical counterpart's characteristics, behavior, life, and performance in a timely fashion."* This may be an expression of the fact that aspects of optimization and control are often neglected in reports on digital twin implementations [101], [102]. Recent reviews of the literature by Kritzinger *et al.* [101] (2018) and Fuller *et al.* [103] (2020) also highlight the absence of a common and clear definition of the digital twin to date, resulting in debatable implementations. Both reviews unsurprisingly call for additional work on the definition and conceptual basis of digital twins. In the hope of providing a clearer picture of the concept, the following section outlines the formal properties of digital twins as described in the scientific literature.

3.2 Formal Properties of Digital Twins

Despite slightly differing descriptions of the conceptual details, the main elements remained mostly constant since M. Grieves conceived the idea of a digital twin in 2002. Accordingly, the following text is organized around the information flow from physical to virtual space, the virtual space itself, and the information flow from virtual to physical space.

Fully Observing

Rosen *et al.* [98] write that *"the digital twin at any time represents the full environment and process state"*, and also Tao *et al.* [26] propose that digital twins *"can visualize and update the real-time status [. . .] for monitoring a production process"*. To understand this aspect of the digital twin concept, the information flow from the physical to the virtual space is thus investigated first. Formally, the state of a system is the information required to predict the output of the system at the next time instance given the history of control inputs to the system. Commonly, the output of a system is captured by sensors, which in theory may be all the information needed to exactly determine the state of the system. The information measured by the sensors and the information contained in the system state need not to coincide though. In addition, the sensor readings may be corrupted by noise. Nevertheless, the system is said to be observable, if its state can be inferred in retrospect from the history of control inputs and measured system outputs by means of a state observation method (e.g. Kalman filtering). Observability is thus assumed when describing the digital twin as *"mirroring"* [20] the information of the physical space. Note that observability only requires the ability to reconstruct the state *in retrospect*. However, the specification of the digital twin as reflecting the *real-time* state (see e.g. [28], [31], [104]) leaves little room for prolonged data collection and reconstruction. The information flowing from physical to virtual space must thus be sensor data of sufficient quantity and quality to determine the state of the physical system in a timely fashion.

The state of the physical system is a function of time, i.e. the state variables change with time. The synchronization of the states of the virtual and the physical space is thus a key property of the digital twin [105]. To model a continuous system digitally, the state variables must be discretized and their values sampled at discrete time instances. Such a sampling event may take place in a synchronous fashion. Models that are updated solely with synchronous and regular sampling events are called time-driven. However, when sensors change value at arbitrary times (e.g. like a relay in a conveyor system), an event-driven approach is more suited, where state changes are induced asynchronously. Both approaches are in use and Lu *et al.* [27] call for further scientific evaluations of each in the context of digital twins, while the DT standard ISO23247 [100] remains nonrestrictive on the use of either. We can understand event-driven models as the more generic ones, since those may include the timed sampling events of a time-driven model. The information flow between physical and virtual space may hence be understood as an event-driven flow of sensor data required to fully observe the state of the physical system.

Adaptive

At the core of the virtual space is a collection of simulation models [21], [28], [99]. Tao and Zhang [30] postulate moreover that this collection comprises models of various types, such as geometrical, physical, behavior and rule-based models, and describe the use of virtual and augmented reality as one aspect of the digital twin to provide *"vivid three dimensional images"* of the physical space. This vision of vivid virtual models is in line with Shafto *et al.* [99] calling for the *"best available physical models"* and Grieves and Vickers [20] describing *"information constructs [...] from the micro atomic level to the macro geometrical level."*

One of the ways the digital twin concept attempts to tackle the complexity of such high-fidelity models, is through modularization of the digital twin. Each component of the physical system is supposed to come with its own simulation model that can be plugged into the overarching simulation of the system [6], [21], [28], [98], [106]. Such a modular architecture may be realized through Functional Mock-up Units (FMUs). FMUs are models, created according to the FMI (Functional Mock-up Interface) standard [107]. FMU models have a predefined set of inputs and outputs that are set by the model creator. This allows for a simple and transparent way of connecting multiple models. An issue of having a distributed simulation, though, is that the dynamics of one model might affect another one. Time delays between the simulation models result in reactions of one model to past states of another one. This interaction can introduce a self-reinforcing feedback loop and cause the simulation to become unstable. The stability of this interaction depends on the whole system dynamics, and not only on each separate model. This is a challenge for a digital twin model, which is to be generic and valid regardless of the simulated system it is placed in, since it is not possible to guarantee that the digital twin will be stable in all interactions [108].

The digital twin is furthermore a hybrid model. Tao *et al.* state that the digital twin includes behavior models and rule models [31]. Behaviour models may be, for instance, physics simulations. Such simulations are models of a continuous state space, that is, state variables are real-valued. Rule models, on the other hand, are of discrete nature in the sense that their state variables change value if certain conditions are met. An example of this would be *if* `button = pressed` *then* `machine state` $\leftarrow$ `off`, where the state variable may also take discrete values. A combination of continuous behavior models and discrete rule models leads to a combined model that is piece-wise continuous, also known as a hybrid system.

In addition, the physical system will be subject to disturbances, such as faults

(e.g. [28], [98]). Vice versa, the digital twin may also be used to predict faults and other events before they occur [20]. To that end, probabilistic models in the digital twin are asked for [31], [104]. These may be used to compute Bayesian probabilities of future failures and mitigate undesired behaviours [20].

In general, a model is the mathematical description of a system in the physical space. Models of dynamic systems are often expressed in state-based form as a parameterized function that relates the current system state and its control input to the next state or the derivative of the state. A digital twin differentiates itself from just any type of (simulation) model by the existence of a corresponding part in the physical space [27], [108]. Such a one-to-one relation between physical space and virtual space *"can be considered as the core elements of a digital twin"* [6]. Moreover, while simulation models are developed usually for exploring what-if scenarios, digital twins are deployed to capture what is currently happening [27]. However, digital twin models of such high fidelity as commonly envisioned are generally still too computationally expensive for a real-time execution in parallel to the physical system. Wright and Davidson [108] thus state that:

> *"In general, a model for a digital twin should be: sufficiently physics-based that updating parameters within the model based on measurement data is a meaningful thing to do; sufficiently accurate that the updated parameter values will be useful for the application of interest; and sufficiently quick to run that decisions about the application can be made within the required timescale."*

Also, Huang *et al.* [109] call for lightweight – and yet – high-fidelity models that can be incorporated as prior knowledge in data-driven approaches in the context of digital twins. Such models are thought to be able to transcend the limitation of conventional and computationally-expensive models in the context of manufacturing processes [109].

Indeed, a data-driven convergence of digital and physical space is key to digital twin-based smart manufacturing [30]. It is achieved – both – by disturbance-rejection control of the physical space based on digital twin models, as well as model-calibration in case of inconsistencies in the virtual space [30]. This requires *"technologies including iterative optimization, self-learning, self-organization and self-adaption mechanisms"* [30]. Once convergence is achieved, evolution of the digital twin alongside the physical system is a second adaptive characteristic of the digital twin [105]. Evolution implies a time-varying dynamic of the system, as for instance reported in [98], where wear and tear in the process is named as reason

for drifting process parameters. If that is to be captured in a parameterized model, such as the digital twin model, its parameters need to be identifiable. A parameter is said to be identifiable if its value can be uniquely determined eventually from the history of control inputs and system outputs. Identifiability is assumed when convergence and evolution of the digital and physical space is desired in the sense of model calibration, such as in [30], [98] for instance. Adaptive systems are inherently non-linear, limiting the choice in optimization and control methods [110]. However, non-linearity of the digital twin is unavoidable if convergence and evolution of virtual and physical space are desired.

Since convergence and co-evolution of digital twin and physical system are frequently listed as research challenge (e.g. [20], [28], [30], [98]), its main principles are briefly outlined here. Assuming a particular parameterized model structure is given, the convergence of digital and physical space is the problem of assigning the right values to the parameters. A good choice of parameter values would ideally result in a small error between true outputs of the system and outputs predicted by the digital twin model. Finding such parameter values can be expressed as a minimization problem of some loss function of the prediction error over a set of observed data. This minimization of the loss function is typically achieved through a numerical search method such as gradient descent or the Gauss-Newton method [111]. In that, the particular structure of the model is not necessarily relevant. For instance, artificial neural networks are a potent class of universal function approximators. As such, they can be used in principle as model structure, or as model structure for the residual errors of another (digital twin) model. *Supervised learning* of neural networks occurs then in the same way through a minimization of some loss function of the prediction error.

While the views on what represents a suitable level of modelling fidelity and modularity seem to vary among researchers, the adaptive property of digital twins is frequently reported as crucial to the concept. It inspired thus the research question **RQ1** posed in Chapter 1. However, parameter identification, as described above, is complicated by the property of the digital twin that is described in the following.

Optimally Controlling

The information flow from virtual to physical space consists of control inputs computed by the digital twin and sent to the physical system [6]. For instance, *"the virtual model can analyze, evaluate, and optimize a scheduling scheme through self-organizing and self-learning"* [26] that is then applied in the physical space. Several

authors (e.g. [27], [28], [98], [104]) highlight such direct and autonomous feedback control from the digital twin to the physical system as an important development direction for smart manufacturing. A system is said to be autonomous if its control input is a function of the system's state. Further, the digital twin may be operated in a model-predictive control style, in which the digital model is used to forecast possible system trajectories [20], [28], [31], [98]. These forecasts are then used to derive autonomously and "intelligently" [28], [104] the optimal control input. To be able to select the better of two possible system trajectories, one needs to evaluate the trajectories' performance based on an objective or cost function. The digital twin, hence, requires a performance criterion that maps input and output to a scalar value – for instance a monetary figure – that is to be optimized.

Optimization of dynamical systems can be approached in two ways: as a static problem or as a dynamic problem [112]. Static optimization is also known as parametric optimization, because it optimizes the parameters of the control function. Those parameters remain static while controlling the system. In a dynamic formulation, the control input is optimized dependent on the state of the system. For both problem formulations, model-based and model-free algorithms exist. Model-based algorithms have access to a model of the system to be optimized, and are generally preferable, since they can exploit the structure of the problem to solve it efficiently [113]. Given the structure of the problem, these algorithms combine methods of search, inference and relaxation to arrive at a solution quickly [114].

Optimally controlling a system based on its digital twin model is complicated, though, by the adaptive property of the digital twin, described in the previous section. The research field of adaptive – or real-time – optimization deals with optimal control under model-system mismatches. Chachuat *et al.* [115] divide adaptation strategies for real-time optimization into three categories: the two-step approach, the modifier approach, and direct input adaptation. In the two-step approach model parameters are repeatedly identified through parameter identification methods and the estimated model is then used to determine suitable control inputs, often based on optimization strategies. In the modifier approach, the actual system model is left as is. Instead, an error model of the mismatch is identified, such that the optimization over system model and error model results in system-optimal control inputs. In addition to being applicable to the system model and its objective function, both these approaches can also be applied to any (other) constraints in the optimization problem. Direct input adaptation turns the optimization problem into a feedback control problem, in which the control inputs are optimized through an online search in the system (see

for instance [55] for a comparative study of various search strategies). Each of these adaptation strategies is subject to the dual control problem, which is to describe the trade-off between selecting control inputs that help to identify an accurate model of the system and inputs that are optimal to the model.

3.3 Digital Twins Refined

To summarize the preceding subsections, we refine the digital twin as a collection of digital artefacts, comprised of a modular, parameterized, identifiable, adaptive, stochastic, and non-linear hybrid dynamics model, an event-driven sensing function capable of fully observing the state of the digital twin's physical counterpart, and an autonomous control function that minimizes a given cost function under constraints of the modeled system dynamics. While our definition tries to capture the properties of the digital twin in all its theoretical generality, in practice, digital twin implementations may not posses all of those properties. The digital model might not be modular, stochastic or hybrid, but monolithic, deterministic, or simply continuous or discrete. These alternative properties can be seen as special cases of their more generic counterpart. Similarly, a sampled sensor function is a special case of an event-driven one. True optimality of the autonomous control function might also be difficult to achieve in practice. On the other hand, we can conclude that identifiability of the digital model's parameters is a must for the desired adaptive characteristic, resulting in a time-varying, non-linear digital twin. Furthermore, observability and autonomy seem to be non-optional to the digital twin.

For the above reasons, the digital twin requires besides a digital model:

1. a state observer that reconstructs the state of the physical space from the history of control inputs and system outputs,
2. a mechanism to identify the model parameters,
3. an optimization mechanism for solving the optimal control problem.

These three rather obvious functionalities were recently summarized by the authors in [116]. They resemble the three key characteristics defined by Tao *et al.* [31]: (1) real-time reflection, (2) interaction and convergence, and (3) self-evolution. More recently, Ma *et al.* [117] arrived at similar conclusions and raised *"control issues and interaction methods* (i.e. model parameter updating methods in the context of [117]) *in AI-enhanced shop floor digital twins"* as a future research challenge.

3.4 Challenges for Digital Twin-Based Intelligent Automation

Digital twins are envisioned as digital models of complex systems, such as manufacturing systems, extended by functionality to fully observe the system state at all times, to adapt themselves to the system, and to optimally control it. As such, they include a representation of (a part of) the world, interpret sensory input to the digital twin to construct its current state, improve their system representation from data, and plan sequences of optimal actions based on that representation. By that, they contain most elements of artificial intelligence outlined by Minsky [19]. If they are used in the automation of manufacturing processes, one can understand digital twins, therefore, as model-based intelligent automation.

However, underlying each of the three required functionalities to turn a digital model into a digital twin, is an optimization problem that needs to be solved in a timely fashion. Each additional variable in the system state makes the problem of state observation, parameter identification and optimal control in principle harder and more time-consuming to solve. Yet, for many digital twin applications, a rapid solution of these problems is crucial [108]. This calls for research into domain-specific approaches that leverage specific system properties and heuristics to enable adaptation and autonomous optimal control in the digital twin.

Indeed, the problems of state observation, parameter identification and optimal control have already been worked on in the field of control theory for several decades. The main difference to the research on digital twins is the lower dimensionality of most control applications to enable exact solutions with the available methods. Realistic models of large, complex systems that are synchronized, updated, and optimized in real-time are at the edge of the currently possible. To that end, the digital twin faces two fundamental questions: Is it possible to get such accurate models as frequently desired? And is it possible to compute and optimize them quickly enough? The former is a question of epistemological belief, the later an open research question. Its answer may very well lay in the the control theorist's approach of using lower-fidelity models and mitigating the model-system mismatch through feedback and adaptation. However, to fully realize the digital twin concept, its complex model-based challenges must be overcome in novel ways – for instance by data-driven learning.

CHAPTER 4

Reinforcement Learning

In Chapter 3, the digital twin has been introduced as the model-based approach to intelligent manufacturing automation, in which adaptive system models are used to optimally control the manufacturing system. Optimal control problems are, beside tracking problems, a major class of control problems. The goal of optimal control is to optimize some performance criterion of the system. Accurate and high-fidelity models of the system's dynamics are helpful for that purpose. When such models are not available in the required quality, adaptive control techniques may be applied. Two main classes of adaptive control exist: direct and indirect adaptive control. Indirect methods continuously estimate a model of the system and derive a controller from the current model. Direct methods estimate the controller directly from system interactions. Conventional reinforcement learning can thus be considered a form of direct adaptive optimal control [118]. Sparked by breakthroughs in deep artificial neural networks, reinforcement learning is increasingly seen as a promising approach to optimally controlling complex systems purely from sampled data, i.e. in the direct adaptive control manner. A particular focus of the second half of this chapter is, however, on model-based RL, sim-to-real learning, and learning with *"expert"* controllers, such as established model-based control methods.

4.1 Foundations of Reinforcement Learning

Underlying any reinforcement learning algorithm is the assumption that the system to be controlled can be seen, and interacted with, as an *Markov Decision Process* (MDP). An MDP is a tuple $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \rho, f, \gamma \rangle$, where $\mathcal{X}$ is the set of states (*state space*), $\mathcal{U}$ is the set of actions (*action space*), $\rho(x,u) : \mathcal{X} \times \mathcal{U} \mapsto \mathscr{P}(\mathbb{R})$ is the stochastic reward function for each $(x,u) \in \mathcal{X} \times \mathcal{U}$, $f(x,u) : \mathcal{X} \times \mathcal{U} \mapsto \mathscr{P}(\mathcal{X})$ is the stochastic transition function for each $(x,u) \in \mathcal{X} \times \mathcal{U}$ (i.e. *dynamics model* $x_{k+1} = f(x_k, u_k)$), γ is a discount factor (alternatively: planning horizon H). In the MDP model, time advances in discrete steps $k = 0, 1, 2, \ldots$. Each state has the Markov property, meaning that the current state vector together with the control input contains all information necessary to predict the associated reward r_{k+1} and next state x_{k+1}. In other words, the state is representative of the control history. If that assumption does not hold, e.g. because the state of the system is only partially observable, the state definition must be the history of observations to enable optimal control. This significantly increases the reinforcement learning complexity, because no such history state is visited more than once [119]. In the following, we generally assume the underlying state to be directly observable and that the Markov property is thus satisfied.

The goal in RL is then to maximize the expected return R (that is the expected sum of discounted future rewards), i.e.:

$$\text{maximize} \quad R = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{k+1} \right] \tag{4.1}$$

$$\text{subject to} \quad x_{k+1} = f(x_k, u_k) \quad \forall k \in [0, \infty) \,, \tag{4.2}$$

by learning a control policy $\pi(x) : \mathcal{X} \mapsto \mathcal{U}$ for all $x \in \mathcal{X}$ that fulfills this goal. For that purpose, it is helpful to define the value functions of an MDP. The state-value function $V^\pi(x)$ of a control policy $\pi(x)$ is the expected return R if following π from state x. The state-action-value function $Q^\pi(x,u)$ of $\pi(x)$ is the expected return R if taking action u in state x and following π from the next state x' onwards. Value function define a partial ordering over policies. Policies that achieve the maximal expected return in all states are called the optimal policies π^* and yield the optimal value functions. Those can be defined recursively by the Bellman optimality equations:

$$V^*(x) = \max_{u \in \mathcal{U}} \sum_{x'} p^u_{xx'} \left[r^u_{xx'} + \gamma V^*(x') \right] , \tag{4.3}$$

$$Q^*(x,u) = \sum_{x'} p^u_{xx'} \left[r^u_{xx'} + \gamma \max_{u \in \mathcal{U}} Q^*(x',u') \right], \tag{4.4}$$

where x' is the next state, $p^u_{xx'}$ is the probability of transitioning to x' from x after applying u, and $r^u_{xx'}$ is the reward received in that transition. These equations are at the core of most RL algorithms [120]. Note here that these standard RL equations are based on the expected, i.e. the average, future reward. In controlling a system, this average reward need not to exist. For instance, a particular control action may result in a reward of 0 in $90\,\%$ of the time and 100 in the remaining $10\,\%$. The control action's reward expectation of 10 will never be observed though. Risk-sensitive reinforcement learning [121] and distributional reinforcement learning [122] address this issue.

RL algorithms can be primarily categorized into model-based (such as Dynamic Programming) and model-free methods that learn purely based on data sampled from the MDP (e.g., the physical system to be controlled). An exact (tabular) and an approximate ('deep') version exists for most algorithms. Model-free methods can be further divided into temporal-difference (TD-learning) methods (such as Q-learning, see Algorithm 1) that exploit the estimation errors in the value functions, and policy optimization methods (such as REINFORCE, see Algorithm 2) that tune the parameters of the policy directly. Actor-Critic methods combine policy optimization and TD-learning. Further information can be found in [120], [123] and [124].

Algorithm 1 Q-learning

Initialize $Q(x,u)$ arbitrarily
for all episodes **do**
  Initialize x
  for all steps of episode **do**
    Choose u from x using policy derived from Q
    Take action u, observe r, x'
    $Q(x,u) \leftarrow Q(x,u) + \alpha\left[r + \gamma \max_{u'} Q(x',u') - Q(x,u)\right]$
    $x \leftarrow x'$
  end for
end for
return $\pi(u) = \arg\max_u Q(x,u)$

Algorithm 2 REINFORCE

Require: A differentiable policy parameterization $\pi(u|x,\theta), \forall u \in \mathcal{U}, x \in \mathcal{X}, \theta \in \mathbb{R}^d$
  Initialize π with random weights θ
  repeat
    Generate trajectory $x_0, u_0, r_1, \ldots, x_{K-1}, u_{K-1}, r_K$ following π
    for all steps of episode $k = 0, \ldots, K-1$ **do**
      $G \leftarrow$ return from step k
      $\theta \leftarrow \theta + \alpha\gamma^k G \nabla_\theta \log \pi(u_k|x_k,\theta)$
    end for
  until $\mathbb{F}$
  return $\pi(\cdot|\cdot,\theta)$

4.2 Multi-Agent Reinforcement Learning

In larger systems, it can make sense to distribute its control across multiple actors. Three distinct communication structures can be chosen for that purpose [125]. In a centralized control approach, actors (or agents) communicate with a central actor that aggregates state information across agents and determines control actions for each agent. In a distributed approach, each single agent interacts with the system independently of other agents. In a networked approach, control is distributed across agents, but the agents can exchange information between each other that can be used to determine the optimal action of each single agent. In such distributed control approaches, it is further differentiated whether agents cooperate, compete with each other, or largely pursue goals independent of each other [125]. In the context of intelligent manufacturing automation of large-scale systems, the distributed control with networked and cooperating agents may be most favourable.

However, the networked and cooperative setting may not necessarily be most practical, since single agents may be developed by different vendors without knowledge of the complete system and all interactions between its sub-systems. In such a situation, each RL-agent may try to solve its own MDP-view of the systems. The dynamics of that MDP are potentially changing, however, due to the learning of other agents. That means the stationarity assumption of standard RL algorithms does not hold [126]. Hernandez-Leal *et al.* [126] identified five strategies in the literature to address this issue: (1) ignore the non-stationarity, (2) forget earlier data and continuously re-train on the latest one, (3) act *"robustly"* by accounting for all possible changes, (4) learn a model of the non-stationarity, or (5) optimize against other agents' behavior while knowing that they will do the same. The practical simplicity

in designing such large-scale distributed systems may thus be counterbalanced by a decrease in optimality and/or sample-efficiency.

4.3 Challenges of Reinforcement Learning-Based Intelligent Automation

The generality of the Markov decision problem formulation allows for the application of reinforcement learning to a wide range of systems, including manufacturing automation systems. And despite the relative simplicity of most reinforcement learning algorithms, impressive results have been achieved in various domains. As such, reinforcement learning is a promising research avenue towards intelligent automation. However, this generality comes at a price. The following outlines a few issues of reinforcement learning that need to be considered not only in the manufacturing automation context, but in most real-world applications of RL.

Like many other optimization problem formulation and solution methods, RL also suffers from the curse of dimensionality. States of the MDP are usually described using state features (e.g. $\mathbf{x} \in \mathbb{R}^d$). The size of the state space $|\mathcal{X}|$ is exponential in the dimensionality d of the state features. This is especially acute, since the MDP formulation generally does not assume any structure on the state or action space that could be exploited by the learning algorithm. The data sample complexity of model-free RL algorithms, hence, becomes worse with each additional state feature that is added to the problem formulation. Exact (i.e., tabular) RL algorithms, thus become quickly intractable when the problem size grows beyond a few state features.

Since a model of the transition dynamics is usually not given, the reinforcement learning controller faces the exploration/exploitation dilemma. The controller must explore the system to acquire new information, but it must also exploit what it knows to behave optimally. The exploration/exploitation dilemma is a *fundamental* problem in RL. Even with the best possible exploration strategy, model-free RL algorithms will always need a minimum amount of exploration, during which the controller acts sub-optimally.

In physical systems, such as manufacturing systems, the transient control performance during the exploration phase may be, though, more crucial than the final optimal performance. In the RL literature, however, algorithms are often judged on their final performance after having sampled millions of data points. In a real-world setting, the transient performance, while learning, might be more important, since sampling data from a physical system is often expensive and subject to safety con-

straints [127], [128]. Simplifying the learning problem by selecting a reward function design that provides frequent informative feedback is thus tempting. However, reward function design is difficult for many applications [129]. A poorly specified reward function might cause the RL algorithm to game the reward function instead of solving the intended problem [128], which is also known as *reward hacking*. This may result in unexpected and undesired – even unsafe – *"optimal"* control behaviors that may not immediately be distinguishable from the transient performance during learning.

Naive exploration strategies are, in addition, often favoured over "smart" exploration strategies because of their simplicity in implementation and their elegant theoretical properties. A common naive exploration strategy is ϵ-*greedy*, which selects a random action with probability ϵ and acts greedily otherwise. "Smarter" exploration strategies that make use of Bayesian confidence bounds are common in the multi-armed bandit (i.e., the state- and dynamics-free RL setting) literature. For instance, Auer *et al.* [130] introduce the UCB1 exploration method for multi-armed bandits. UCB1 estimates an upper confidence bound on the mean of the reward of an arm based on the current sample mean and an exploration bonus based on the number of times an arm has been played. A well explored arm will have an exploration bonus that tends to 0, whereas an arm that has not been played as often obtains an exploration bonus that increases with every round the arm has not been played. Such approaches are only slowly extended to the full MDP context. Tijsma *et al.* [131], for example, add weighted UCB1 terms to the Q-values when deriving the policy from the Q-function. This method shows to be relatively easy to tune and performs better than ϵ-greedy in experiments. However, the exploration bonus in this formulation is only considered in the one-step action selection, but not in the Q-function itself. Such count-based techniques are even harder to extend to large state spaces common to deep reinforcement learning problems. A potential solution is proposed by Tang *et al.* [132], who employ static hashing functions to transfer count-based exploration to the deep learning context. The hash function abstracts the high-dimensional state vector into a low-dimensional representation for which a state visitation count can be maintained. It is shown that this technique produces efficient exploration and near state-of-the-art performance in multiple continuous control tasks as well as in Atari video games. Smart exploration strategies are, nevertheless, still an important research avenue, especially for real-world applications, such as in intelligent manufacturing automation, in which data samples often come at a cost.

When the state-action space of the learning problem is large, function approxima-

tion for the value and/or control policy functions must be used. Much of reinforcement learning's recent success in controlling complex systems can be attributed to the ability of artificial deep neural network function approximation to learn lower-dimensional representations of the state space. A controller in form of a deep neural network is, however, hard to interpret or analyze. Moreover, the combination of function approximation (e.g. neural nets), bootstrapping (e.g. TD-learning), and off-policy learning (training on data collected under a different policy) is known as the deadly triad [120]. All three elements are used in most deep RL algorithms, but theoretically can lead to divergence of the learning process. In practice, a less severe form of divergence occurs, in which state and action values are severely overestimated but likely maintain their relative ordering [133]. Such overestimated values loose their informative value over expected future control performance, which might be of interest in real-world applications.

Lastly, results achieved by RL algorithms and reported in the literature, can be hard to reproduce due to their sensitivity to for example: random seeds, hyperparameter settings, implementation details, reward design or local optima [134]. So, even if a suitable algorithm has been selected, considerable time and data may be needed for experimentation before the desired results materialize. In summary, reinforcement learning requires often a considerable amount of data for learning, during which certain performance aspects like optimality and safety may not be guaranteed. If function approximators like neural networks are used, explainability, in addition, may not be given.

4.4 Solution Approaches

Several sub-fields within reinforcement learning research aim to address the practical shortcomings of standard reinforcement learning: sample-complexity, optimality, and safety. The approaches discussed in this section are summarized in Table 4.1.

Reinforcement Learning from Demonstrations

When reinforcement learning is used to replace the controller of an existing system, data of the system under the previous control often exist. This data may be used to initialize the RL policy [135]. For that purpose, an off-policy algorithm, such as Q-learning, must be chosen, which can learn from data generated by a different policy than its current own one. In the deep RL scheme, algorithms like for instance

Table 4.1: An overview of the presented solution approaches.

Approach	Applied To	Prior Knowledge	Issues Addressed
RL from Demonstrations	Policy & Value Functions	Sampled Transition Data	Data Sample Complexity, (Safety)
Apprenticeship Learning	Reward Function	Sampled Transition Data	Reward Hacking, (Safety)
Imitation Learning	Policy	Expert Policy Function	Data Sample Complexity, (Safety)
Model-based RL	Policy & Value Functions	Transition & Reward Functions	Data Sample Complexity, Safety

Deep Q-learning from Demonstrations [136] or Deep Deterministic Policy Gradient from Demonstrations [137] pre-train the neural network in a supervised manner on demonstrated state-action-reward-next-state data samples. When the RL controller then takes control of the system, it is thereby hoped that its policy performs reasonably well from the start. Ideally, a costly and near random sampling at the beginning of learning can be skipped by such methods. The RL controller then tries to improve its performance based on standard RL techniques. Additional modifications of the RL algorithms are, however, often needed to prevent the RL agent from *"catastrophically"* forgetting the demonstrated performance.

Alternatively, the demonstration data may be used for inverse reinforcement learning. Inverse reinforcement learning concerns identifying an unknown reward function given demonstrations of corresponding optimal behavior. It is therefore the dual to reinforcement learning [138]. An issue in inverse reinforcement learning is *degeneracy*, i.e. the issue that the observed policy may be optimal for many reward functions [139]. To address this issue, most algorithms either aim to infer a reward function that makes the demonstrated policy by a margin better than alternatives or to find a distribution of reward functions with maximum entropy that explains the demonstrations, such that as few as possible assumptions are made [140]. In a subsequent step, the learned reward function may be used for learning a policy through reinforcement learning, also known as apprenticeship learning [141]. Apprenticeship learning circumvents some of the challenges associated with reward function engineering (such as reward hacking) but may lead to policies that do not generalize to unseen parts of the state space [140].

Learning with Expert Controllers

In contrast to learning from demonstrations, learning-with-expert approaches assume that a sufficiently good *"expert"* control policy is available instead of demonstrated system trajectories only. One such approach is imitation learning [37]. The goal of imitation learning is to train a novice controller to imitate the behaviour of an expert, which, for example, may be a human or a computationally expensive algorithm.

A popular benchmark method in imitation learning is, because of its simplicity, the Dataset Aggregation (DAgger) algorithm [142]. DAgger trains a novice controller online by applying a supervised learning algorithm to an aggregated set of state observations, each of which is labelled with the expert's control action. The expert's control of the task is annealed and the novice is given successively more control. The expert then labels all observations with the optimal control action in retrospect, which is used to update the learner. Such imitation learning aims for more robust learned policies compared to policies learned in a supervised manner from demonstrations (compare to the previous Subsection 4.4 – RL from Demonstrations). Safety and expert query-efficiency during training and deployment are crucial in that and have been addressed in extensions such as [143]–[146]. For instance, a Bayesian neural network is used in [146] to estimate potential errors of the novice policy, and by that safety and query-efficiency is improved.

Often, the performance of policies learned through imitation learning algorithms is upper bounded by the performance of the expert policy [147]. Algorithms such as [148] and [149] combine, for that reason, imitation learning with subsequent reinforcement learning. By that, a robust initial policy is learned in a relatively safe and sample-efficient way before RL is used to optimize the policy further.

Model-based Reinforcement Learning

In model-based RL, the reinforcement learning makes use of models of the system dynamics (i.e. its transition function) and its reward function. These models can either be given as prior knowledge or learned from previous interactions with the system. In the DYNA framework [150], [151], for instance, a dynamics model is estimated from recorded data. The reinforcement learning then trains simultaneously on observed and generated data. Such approaches have the potential to reduce data sample complexity [152]. They can furthermore improve estimates of value functions [153]. Improved value function estimates have been key to the success of AlphaGo [25], where a combination of given rule models and estimated transition

probabilities (i.e. "self-play" in Monte Carlo Tree Search) supported the RL agent in its decision making. Prior knowledge in form of given models can additionally enable safe reinforcement learning in previously unknown situations [154], [155].

Fitting high-capacity models for complex dynamics based on small data sample sizes is challenging though [156]. In such situations, the risk of over-fitting the model is significant. The resulting model errors, biases, and uncertainties can be detrimental for the final performance of model-based RL algorithms [157]. The sim-to-real approach in reinforcement learning addresses the model-system mismatch in multiple ways. On the one hand, minimal-mismatch simulation models are strived for that enable any RL algorithm to learn a control policy in simulation that can be directly transferred to the real system. On the other hand, domain randomization techniques for the sensory input or the system dynamics aim for robust policies by generating data distributions from imperfect simulation models that contain data observed in the real system [36]. Robust reinforcement learning algorithms, moreover, explicitly consider the model-system mismatch in the learning procedure [36]. Robustness of the control policy interferes, however, often with control performance.

An alternative approach has been recently introduced by Gros and Zanon [158], who showed that economic non-linear model predictive control (ENMPC) formulations [159], [160] may be used as function approximators in RL. Here, the value function $V(x)$ is the optimal solution to the ENMPC problem. Similarly, the state-action value $Q(x, u)$ is the solution of the ENMPC problem, if the first control u_0 in the ENMPC is constrained to equal u. When using function approximation in RL, such as in deep Q-learning, parameter updates frequently take the form:

$$\theta \leftarrow \theta + \alpha \tau_k \nabla_\theta Q_\theta(x_k, u_k) \; , \tag{4.5}$$

where α is the learning rate and the temporal difference error is

$$\tau_k = \rho(x_k, u_k, x_{k+1}) + \gamma V_\theta(x_{k+1}) - Q_\theta(x_k, u_k) \; . \tag{4.6}$$

The gradient ∇ of Q with respect to its function parameters θ is then the gradient of the optimal solution to the Lagrangian relaxation $\mathcal{L}$ of the ENMPC scheme:

$$\nabla_\theta Q_\theta(x, u) = \nabla_\theta \mathcal{L}_\theta(x, u, \lambda^*) \; , \tag{4.7}$$

where λ^* are the optimal Lagrangian multipliers [158]. This approach offers thus a principled way to incorporate rich prior knowledge and has been evaluated in [97].

CHAPTER 5

Summary of Included Papers

This chapter provides a summary of the included papers that aim to answer the two research questions of this thesis. The research questions are: **RQ1** – *How can machine learning be used to mitigate the model-system mismatch in digital twins, while minimally altering the provided solution?* and **RQ2** – *How can prior model-based knowledge be introduced in reinforcement learning to improve sample-efficiency, explainability, and safety of the learned control policy?*.

5.1 Paper A

Constantin Cronrath, and Bengt Lennartson
How Useful is Learning in Mitigating Mismatch between Digital Twins and Physical Systems?
Published in IEEE Transactions on Automation Science and Engineering, (Early Access), Dec. 2022.
 DOI: 10.1109/TASE.2022.3231386 .

Given a highly accurate digital twin model and sufficient computational resources, virtually any off-line optimization algorithm could be used to derive optimal control

inputs from the twin to improve the performance of the physical system. For very practical reasons, however, this accuracy may prove elusive, inducing a mismatch between digital twin and physical system and, hence, render such control potentially sub-optimal. In Paper A, we are interested in the mitigation of model-system mismatches through learning-based algorithms in the direct input adaptation scheme (see Section 3.2 or [115]). In an observational study, we juxtapose the usefulness of 16 learning-based and blackbox optimization algorithms for the purpose of restoring optimal control under system-twin mismatches in high-dimensional, continuous, smooth, and non-linear static systems.

More concretely, we base our experiments on a geometry assurance task in the assembly of automotive sheet metal parts, previously described in [48] and Section 2.1. Here, adjustable locators of the assembly fixture position two or more sheet metal parts for subsequent joining through welding or clinching. The parts are depicted in Figure 5.1. The locators of the physical fixture may deviate from their digital twin due to wear and tear, damages, or effects of changing process parameters. We aim to optimize geometric quality by compensating for this system-twin mismatch through direct input adaptation. To that end, we formulated the problem as an online optimization problem with a control architecture as shown in Figure 5.2. In our evaluation, we randomly sample problem instances from this geometry assurance context and measure the usefulness of 16 different algorithms (BOBYQA, Bayesian Optimization, Conjugate Gradients, COBYLA, DDPG, DIRECT-L, Differential Evolution, HOO, L-BFGS-B, MMA, Nelder-Mead, PRAXIS, Powell's Method, SLSQP, SPSA, and Subplex). We denote an algorithm to be more useful than another algorithm based on three aspects: 1) it requires less data samples to reach a desired minimal performance, 2) it achieves better performance for a reasonable number of data samples, and 3) it accumulates less regret.

Our results indicate that local gradient-based blackbox optimization algorithms outperform learning-based algorithms in terms of sample-efficiency, accuracy within a limited sampling budget, and regret, even in the presence of white measurement noise. These local algorithms are most useful for direct input adaption in all respects in the deterministic setting, but perform mostly worse when white output measurement noise is present. However, the stochastic, local algorithm SPSA outranks almost all other algorithms on all criteria in the stochastic setting. Stochastic algorithms generally perform comparably better in the stochastic setting. This applies to the learning-based algorithms too. Yet, learning-based algorithms cannot be considered most useful in neither the stochastic nor the deterministic setting of our

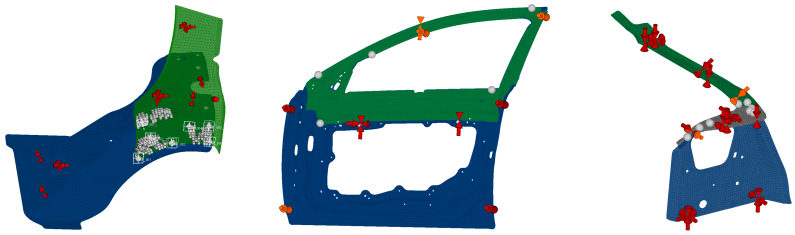


Figure 5.1: The three use cases for our experiments: (a) shows a sub-assembly of a car reinforcement bracket. Its two parts are positioned in space by twelve locators (shown in red). (b) shows the frame of a car door. Its two sheet metal parts are positioned in space by 19 locators. (c) shows a pillar of a car, consisting of three parts held in place by 20 locators. A digital twin computes the geometric quality after the assembly process (positioning, clamping, joining, releasing, scanning) based on several detailed Finite Element Method calculations, including elastic deformations and springback under the assembly forces. The final geometric quality may be influenced by adjusting each single locator along its axis.

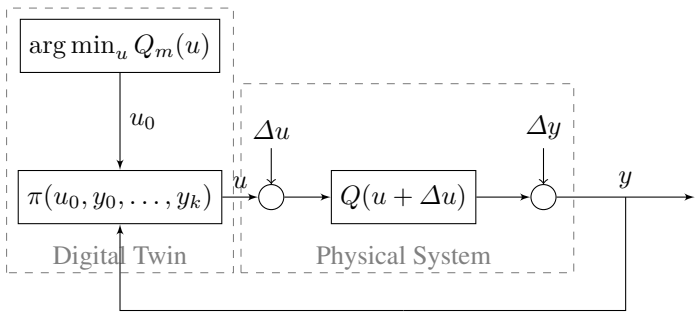


Figure 5.2: A block diagram of the system under consideration. We are interested in control policies π capable of determining a high-dimensional control input u, that minimizes a performance or objective function Q of the physical system. A model, or digital twin, of this function is accessible, that may be queried for its minimizer u_0. While we assume this model to be sufficiently accurate in the neighborhood of the minimum, the model is assumed not to capture input disturbances Δu, and output disturbances Δy.

experiments, since they perform generally worse than other types of blackbox optimizer. An exception is Bayesian Optimization, which is considered a learning-based algorithm in our context, and ranks highly in our comparisons, except for in the comparison on the regret metric.

The usefulness ranking is fairly consistent across algorithmic classes and user-determined tolerance levels in minimal performance and regret metrics. This applies to both the deterministic and the stochastic setting. Beside that, we observe the same dominance of the local optimization algorithms across all values of the sampling budget. The dominance of gradient-based algorithms and Bayesian Optimization may be traced back to a smarter exploration strategy compared to the near "random" strategy of others. Technically, our conclusions can *not* be generalized to the larger set of all existing blackbox and learning-based algorithms, since they result from a fixed factor experiment. A different choice of algorithms may have resulted in a different conclusion. However, this is unlikely since our results indicate general trends. First, we observe that local algorithms are generally more useful than global algorithms in the direct input adaptation context. Second, we note that stochastic algorithms are more competitive in the setting with noise compared to the deterministic setting. A different selection of algorithms is likely to reproduce these trends. Our results may also extend to *static* system-twin mismatches that can be described as linear or nonlinear continuous input our output disturbances. The effect of such disturbances would be a "warping" of the optimization manifold over the search space in u. As long as the local convexity and smoothness property of the optimization manifold is maintained, such a warping would most likely be without significant impact. It is important to emphasise here, though, that Wolpert and Macready [161] proved mathematically that all blackbox optimization algorithms perform in average equally well across all optimization problems. The dominance of quasi-Newton algorithms on the deterministic problems and their complete failure in the presence of output noise is likely the most illustrative example of this no free lunch theorem in the paper at hand. Accordingly, we consider the empirical observation of this theorem as yet another reason to research into adaptive optimisation algorithms in the context of system-twin mismatch that leverage properties of the specific problem class.

In conclusion, we find that gradient-based blackbox optimization is better suited to compensate for system-twin mismatches of high-dimensional, continuous, smooth and static performance functions than learning-based algorithms. However these gradient-based methods do not learn from available data, which can be a drawback in the long run. This highlights the need to extend the search for algorithms, which

can restore optimal control in digital twin governed autonomous systems, beyond generic machine-learning algorithms to include for instance smart exploration and uncertainty estimation methods.

5.2 Paper B

Constantin Cronrath, Abolfazl Rezaei Aderiani, and Bengt Lennartson
Enhancing Digital Twins through Reinforcement Learning
Published in Proceedings of the 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), Vancouver, Canada,
pp. 293–298, Sept. 2019.
 DOI: 10.1109/COASE.2019.8842888 .

Similar to Paper A, we assume the digital twin as given in this paper. Specifically, the digital twin controls the fixture in a sheet metal joining task to improve the geometric quality of each individual assembly. The considered nominal assembly is shown in Figure 5.1 (a). The digital twin and the emulated physical system only differ in a single locator position that was picked to introduce a noticeable mismatch. Different to Paper A, however, individual part geometries are considered in Paper B. In this case study, we are thus interested in developing a direct input adaptation algorithm that mitigates against the mismatch and utilizes the side information given by the part geometries.

We formulate the task as a contextual bandit problem, which is a reduced reinforcement learning formulation without state dynamics, but in which the side information is considered to be the observed state in each round. To account for the manufacturing context, a focus in Paper B is on safe exploration strategies that maintain on average with high probability a given level of performance during learning. Garcia and Fernandez [121] identified *teacher advice* as one common approach to incorporate external knowledge in the exploration process to make the same safer. With the availability of the *digital twin*, we have access to a default policy π_d that can be regarded as such teacher advice. The default policy π_d is the original control policy of the digital twin, before we apply deep learning to compensate for model inaccuracies. This default policy may be sub-optimal, but arguably superior to the agent's policy π_a in the initial learning period. We assume the performance of the digital twin's default policy π_d is known from prior operation of the system. A problem formulation similar to Wu *et al.* [162] then suits our manufacturing case.

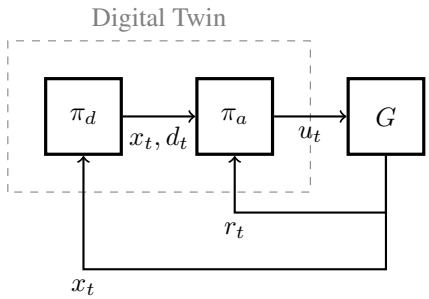


Figure 5.3: The architecture of our EDiT algorithm. The digital twin observes a state x_t and decides on a control action d_t based on its default policy π_d. Our RL algorithm EDiT observes, both, x_t and d_t. It decides then whether to apply d_t or $u_t = \pi_a(x_t)$ to the physical system G. The system then generates a feedback signal (reward) r_t and a next state x_{t+1} that is observed by the digital twin. r_t is used to improve the EDiT policy π_a.

Accordingly, we define a cumulative performance constraint with respect to the digital twin's default performance and constrain exploration further by introducing a per part constraint for additional safety. Since the expected reward of the learning agent's action is unknown *a priori*, we employ a Bayesian approach and compute upper and lower confidence bounds on the expected reward of the action. In each round, our proposed exploration strategy then decides whether the agent's proposed action is likely to improve performance while maintaining the cumulative and per part performance constraints – and if not chooses to apply the digital twin's default policy. The resulting algorithm for Enhancing digital twins is named EDiT and its control architecture is depicted in Figure 5.3.

Our particular test case consists of two sheet metal parts of a car body shown in Figure 5.1 (a). The geometry of the parts is given by their point clouds of $\sim 2.5k$ points each and represent the side information given to our learning algorithm. We evaluate the algorithm on 250 part instances. Each of the fixture's twelve locators are adjustable along their axes and constitute the action space. In our experiments we observe an overall improvement in performance compared to the digital twin's default policy as listed in Table 5.1. In the best case, this improvement is realized just after a few rounds. In the worst case, the learning requires up to $2k$ rounds of exploration until improving upon the default policy. In average, though, we see an improvement of about 0.25% over $10k$ rounds. This is likely due to the particular

Table 5.1: Performance of our proposed EDiT algorithm over 10 Repetitions of $10k$ Rounds.

Measure	Mean	Range
Mean Reward r_u	1.0025	$[0.385 - 1.682]$
Number of Constraint Violations		
Per Part ($\geq 95\,\%$ of π_d)	207	$[187 - 229]$
Cumulative Performance ($\geq 99.5\,\%$ of π_d)	2	$[0 - 10]$

mismatch that was introduced in our test case and because of the high-dimensional state-action space. We expect the algorithm to take many more rounds in this case, before the optimal policy is fully learned. We further notice a number of violations of the safety constraints. Future research directions may include extensions of the algorithm to improve safety and sample-efficiency. The behaviour of deep neural network estimates can be unpredictable while learning. Although we employ a performance constraint and Bayesian neural networks to estimate uncertainty, we see further safety guarantees needed for the application of deep reinforcement learning in industry.

To conclude, we have introduced the learning algorithm EDiT for enhancing the control policy of digital twins in continuous domains, based on a contextual bandit formulation. It utilizes the digital twin as safety policy to maintain constraints imposed on the learners performance. While this formulation has been shown suitable for the manufacturing context, the behaviour of the learning algorithm in the direct input adaptation scheme may be unpredictable in that the performance constraints may nevertheless be violated occasionally.

5.3 Paper C

Anders Sjöberg, Magnus Önnheim, Otto Frost, **Constantin Cronrath**, Emil Gustavsson, Bengt Lennartson, and Mats Jirstrand
Online Geometry Assurance in Individualized Production by Feedback Control and Model Calibration of Digital Twins
Published in Journal of Manufacturing Systems,
vol. 66, pp. 71–81, Jan. 2023.
DOI: 10.1016/j.jmsy.2022.11.011 .

As in Paper A and B, a digital twin controls the fixture in a sheet metal joining

task to improve the geometric quality of each individual assembly. The considered nominal assemblies are shown in Figure 5.1 (a) and (b). In this paper, a drifting mismatch is introduced for one, as well as three, locators, and individual part geometries are considered. A modifier adaptation approach is chosen here to learn a model of the drifting mismatch.

More specifically, we re-frame the mismatch problem between digital twin and system as a state observation problem. A dynamic modifier model for the drifting mismatch is introduced and adapted by a parameterized controller that balances *exploration* (tracking ability of the drifting mismatch) and *exploitation* (single-assembly quality improvements). An Unscented Kalman Filter (UKF) is used in that as a state estimator, providing a state estimate of the drifting mismatch along with an uncertainty measure in the form of the posterior covariance of the UKF. Our proposed control method has access to the current assembly, along with the control signal computed by the digital twin, the state estimate of the mismatch and the uncertainty measure of the UKF. Using those data, the controller is implemented as a one-step look-ahead minimizer of a weighted combination of the expected next-step quality and the next-step uncertainty of the UKF-estimator, denoted as the exploitation and exploration loss, respectively. The exploration loss is designed to promote the estimation of the mismatch, as measured by the (future) state covariance of the UKF, in order to maximize the expected average quality of future assemblies. We denote this control scheme as the w-controllers, due to the weighting parameter w that balances exploration and exploitation. In our digital twin context, the w-controller adjusts the control signal computed by the digital twin, which optimizes fixture locators for each individual assembly. After welding, the assembly is scanned to measure the geometric quality and that is in turn fed back to our controller. The controller thus incorporates both the digital twin and the feedback signals of the physical system. This control concept is depicted in Figure 5.4.

In our simulated experiments, we evaluate the exploration-exploitation trade-off on the individual geometry assurance task depicted in Figure 5.1 (a) and (b), and show that significant quality improvements are possible with our proposed approach. To acquire an evaluation environment with ground truth, i.e., known discrepancy between the physical system and the digital twin, we also simulate the physical system equivalently as our digital twin. In the first case (Figure 5.1 (a)), the w-controllers demonstrate significant gains in quality of the produced assemblies, while in the second case (Figure 5.1 (b)) they show negligible to small improvements. The second case is, however, rather insensitive to mismatches, which enables only small gains.

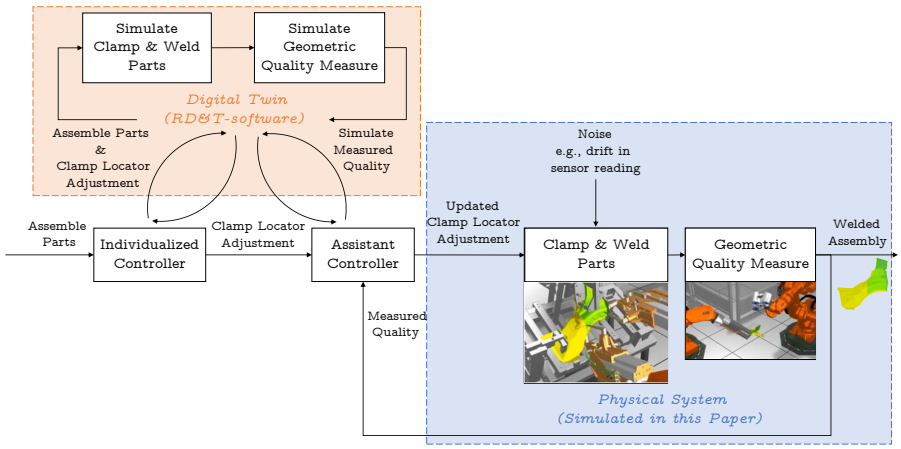


Figure 5.4: An overview of the whole chain of events. Given a set of assembly parts, which are about to be welded, the individualized controller computes the optimal clamp locator adjustment with help of the digital twin, i.e., the RD&T software, and proposes that as a control signal. Due to noise, e.g., drift in sensor reading, in the welding cell the clamp locator might need to be adjusted according to that offset, which is the purpose of our proposed (assistant) controller. After welding, the assembly is scanned to measure the geometric quality and that is in turn fed back to our controller. This controller incorporates both the digital twin and the feedback signals of welded assemblies' quality. In our numerical evaluation of the proposed control scheme, the physical system is simulated in accordance with the digital twin.

As expected, higher exploration results in better estimations. However, adequate estimations come with a cost, since the quality is negatively affected for higher values of exploration rate. On the other hand, by only optimizing the expected next-assembly quality, the model loses track of the offset. That in turn may result in extremely poor quality and often ultimately in the controller diverging. Thus, we see a clear trade-off between exploration and exploitation in the quality.

Often a combination of exploration and exploitation is most beneficial. However, it is not a trivial task how to choose the weight w. The exploitation limit, where controllers start to diverge, differs depending on which locators are considered. This is likely in part due to an equal weighting of the uncertainty corresponding to each component of the mismatch regardless of how easy or hard that component is to esti-

mate. Furthermore, the proposed w-controllers come with significant computational overhead, especially as the dimension of the state-space increases, which may limit the practical usefulness of our approach. It does not scale well due to the curse of the dimensionality – both as a consequence of the increased number of Sigma points needed in the UKF and the increased number of evaluation points needed to approximate them.

In summary, we re-framed the mismatch problem between digital twin and system as a state observation problem. A dynamic modifier model for the drifting mismatch was introduced and adapted by the w-controller method that balances exploitation and exploration. While our evaluations illustrated the difficulty of balancing exploitation and exploration, they also showed that this method can significantly improve performance under drifting model-system mismatches.

5.4 Paper D

Constantin Cronrath, Tom P. Huck, Christoph Ledermann, Torsten Kröger, and Bengt Lennartson
Relevant Safety Falsification by Automata Constrained Reinforcement Learning
Published in Proceedings of the 2022 IEEE 18th International Conference on Automation Science and Engineering (CASE), Mexico City, Mexico,
pp. 2273–2280, Aug. 2022.
 DOI: 10.1109/CASE49997.2022.9926460 .

In Paper D, the aim is to introduce model-based knowledge into reinforcement learning. For that purpose, a safety falsification application in the context of collaborative robotics is investigated. Simulation-based falsification is a testing method for uncovering safety hazards of complex safety-critical cyber-physical systems, such as collaborative robots. One particular challenge in reinforcement learning-based falsification is that it should identify scenarios which are *safety-critical* and *relevant* at the same time. These two goals do not always go hand in hand, in some cases, they may even be opposed to each other. One approach to address this is to use different reward components to encourage both high risk and relevance. Yet, this raises the questions of (1) how to define and prescribe what is “relevant" and (2) how to balance the reward components in a principled manner. This paper proposes automata constrained reinforcement learning, in which rewards for relevant behavior are tuned via Lagrangian relaxation.

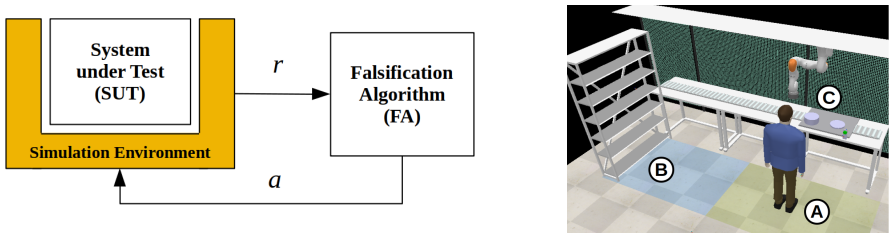


Figure 5.5: (a) The principle of simulation-based falsification: the falsification algorithm (FA) chooses actions a to adapt the simulation environment which excites the system under test (SUT) and receives rewards r (e.g. risk or coverage metrics). (b) The SUT used for testing the proposed automata constrained reinforcement learning: the nominal operating conditions are to walk from A to B to pick parts from the shelf, walk back, and assemble the parts in collaboration with the robot (C).

More specifically, automata specifications are introduced in this paper as prior knowledge to describe the *nominal* operating conditions of the System Under Test (SUT). These specifications are run synchronously with the SUT and issue rewards when the falsification algorithm operates close to the nominal conditions (the basic principle of algorithmic falsification is shown in Figure 5.5 (a)). The underlying assumption here is that the discovery of hazards close to the nominal conditions is more relevant, since these are more likely to occur in practice. Additional rewards are given for high-risk behavior. To balance the reward components, the technique of Lagrangian relaxation is used. In our approach, the reward function of the specification is tuned in the dual problem of the Lagrangian optimization by SPSA [163], while reinforcement learning is used in the primal problem to learn a relevant falsification policy, which considers both MDP rewards and specification rewards. In that, the specifications help to guide the exploration process, but are tuned in such a way that they only alter the learned behaviour as little as required.

The proposed method is demonstrated in an application example from the domain of Human-Robot Collaboration (HRC), where the objective is to identify potentially safety-critical human errors in a collaborative assembly task, while avoiding to deviate unrealistically far from the nominal assembly sequence. This use-case is illustrated in Figure 5.5 (b). Compared to random sampling and conventional approximate Q-learning, we show that the proposed method generates equally hazardous, but at the same time more relevant testing conditions that expose safety flaws.

While the presented use-case study has clearly indicated the benefits of the newly proposed method, it fundamentally relies on a trial-and-error approach that necessitates violating the specifications to learn about them. In addition, the SUT in this case had relatively weak safety measures, meaning that even the unsophisticated random sampling found a considerable amount of hazardous situations. It would be interesting to compare the approaches in test scenarios with stronger safety measures, where unsafe situations are more rare. Another issue to consider is that safety analyses are typically performed iteratively, meaning that after a hazardous situation is found, new safety measures are introduced and the search is then repeated on a slightly modified SUT. Investigating how different algorithms cope with such modifications is also a possible area for future research.

To sum up, we presented a principled method for introducing behavioral specifications into the RL performance criterion to guide and restrict the exploration process. By introducing additional reward components on the basis of a nominal behavior specification, we introduced an incentive for the reinforcement learning algorithm to not only identify safety-critical behaviors, but also remain in a certain vicinity of the nominal behavior, thus making the results more relevant and more valuable from a practical safety analysis standpoint. As is often the case with multiple reward components, finding an appropriate balance between the components can present a challenge. The naïve approach of using a weighted sum of reward components raises the problem of choosing appropriate weights, which are difficult to determine a priori. The Lagrangian relaxation approach demonstrated in this paper provides a more principled approach to balancing the reward components.

5.5 Paper E

Mattias Hovgard, **Constantin Cronrath**, Kristofer Bengtsson, and Bengt Lennartson
Adaptive Energy Optimization of Flexible Robot Stations
Revised version submitted to IEEE Transactions on Automation Science and Engineering, Mar. 2023 .

The aim in Paper E is to use additional prior model-based knowledge in reinforcement learning that goes beyond behavioral specifications. To that end, it has recently been shown by Gros and Zanon [158] that Economic Non-linear Model Predictive Controllers (ENMPC) can be used as function approximators in reinforcement learning. In Paper E, we investigate the application of this promising method in the context

of energy optimization of flexible robot stations. For that purpose, we formulate an approximate model of the optimization task, select an appropriate model-based optimal control method, and adapt the control scheme by reinforcement learning. In that, we extend the method by event-driven online scheduling.

In specific, a generic and adaptive method for energy optimization of flexible robot systems is proposed. The method takes practical requirements into considerations when minimizing energy use by including performance constraints (i.e. meeting deadlines) and by learning unknown system parameters from measurements. For that purpose, the energy optimization problem has been decomposed into: (1) its integer part, for which a graph search algorithm determines the operation sequence that maximizes the capacity for energy reduction based on an approximate extended finite automaton system model; (2) its non-linear part, for which an event-driven model predictive controller tunes the motion parameters of the sequenced operations online to minimize energy use, while ensuring to meet the deadlines; and (3) its online adaptation, for which a reinforcement learning algorithm continuously estimates unknown parameters in the optimization model. This decomposition is depicted in Figure 5.6. In that, the sequence and timing optimizations act as model class of the function approximator for the Q-function in the reinforcement learning. By that, prior knowledge about the reward function, system dynamics, and behavioral specifications can be introduced in the learning. The proposed approach may be also understood as a converging digital twin for energy optimization, in which feedback and learning reduce the model-system mismatch. This is, because a dynamic model of the system is used to predict future states and to optimize performance of the system based on its current state, while the parameters of that model are adapted to reduce any mismatch.

The method is evaluated in a numerical example based on a robotic kitting station (see Figure 5.7), which contains several difficulties commonly found in practical applications, such as stochastic variations, hard deadline constraints and operation failures. In the numerical evaluation, the method works as expected for all tested experimental settings; it is able to quickly and significantly reduce the energy use compared to the unoptimized case, while fulfilling the performance constraint. When comparing optimization with true parameter values to optimization with learned parameter values, we observe only small differences in energy use and no significant differences in probability. This shows that the reinforcement learning finds suitable parameter values, which enables the optimum to be found in the energy optimization. Note here, that the learned model parametrizations are fairly accurate but do

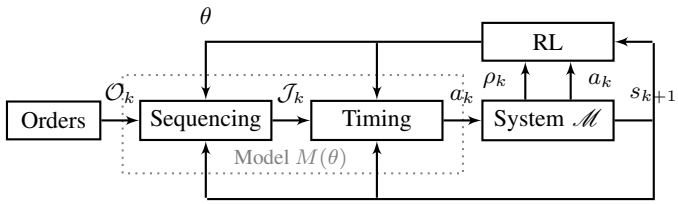


Figure 5.6: Block diagram of the control architecture. Robot operations for dispatched orders are first scheduled to determine the necessary sequence of operations. Then, a model-predictive controller (MPC) optimizes robot motion parameters of each operation to minimize energy use. A reinforcement learning (RL) algorithm monitors the performance criterion ρ (energy use and productivity penalties) of the system and adapts parameters θ of the MPC from that information.

not completely coincide with the true ones. This is because, the goal of the parameter learning is not to find the true values, but to find values that allows the minimum energy use to be achieved.

However, several properties of the proposed control architecture may lead to sub-optimal control actions. For instance, the anytime property of the used scheduling algorithm ensures that a solution is always available when needed, but feasibility is prioritized over optimality in that. In real-time systems, there is often a trade-off between solution time and goodness of solution, such that a good enough solution in time is preferred over the optimal solution too late. In addition, inaccurate model parameters and exploration during learning inevitably mean sub-optimal control actions. As the learning progresses, sub-optimality due to these aspects will decrease. On the other hand, it is tempting to select the most expressive available system model for optimizing energy use, such as high-fidelity digital twin models or flexible universal function approximators. However, ENMPC formulations, that include some form of predictive system model, can be computed quickly, and can be adapted by reinforcement learning, may prove to be a practical compromise between expressive models and flexible function approximation. In [116], the authors argue that a digital twin should be capable of supporting such a formulation in order to fulfill the desired properties of prediction, convergence, and co-evolution.

In conclusion, adaptive economic non-linear model predictive control seems a promising method for combining model-based and data-driven control. The method, investigated in this paper, uses a system model to enforce behavioral constraints while identifying system parameters through reinforcement learning. The result

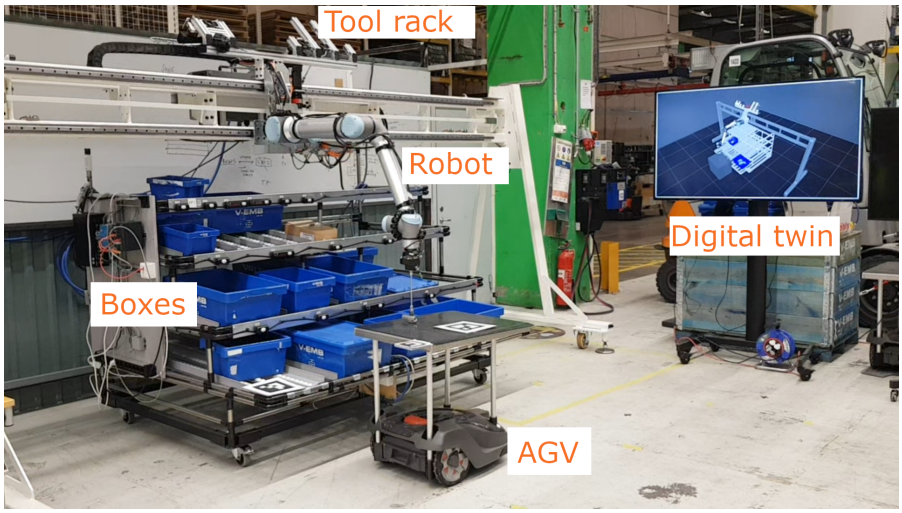


Figure 5.7: An example of a robotic kitting station. The collaborative robot is mounted on a horizontal gantry to be able to pick parts from all areas of the rack. A camera is used to identify the position of the rack and parts within each box. The configuration and state of the system are mirrored at all times in its digital twin.

showed that the system performance was optimized, the unknown parameters were effectively estimated, and the constraints were fulfilled. These results indicate, furthermore, how convergence and co-evolution of digital twin and physical system could be achieved in autonomous and optimal control of manufacturing systems.

CHAPTER 6

Concluding Remarks and Future Work

This thesis is based on the premise that manufacturing automation must become more intelligent to meet current demands on production systems in terms of flexibility, speed, quality, and cost. Two distinct methodical approaches to intelligent automation have been identified in the scientific literature: (1) the model-based digital twin approach, and (2) the data-driven reinforcement learning approach. A review of both approaches in previous chapters of this thesis found that: (1) the digital twin approach faces the model-system mismatch challenge, while (2) the reinforcement learning approach must overcome the performance challenge in small data regimes common to manufacturing. The research presented in this thesis was thus guided by the idea to incorporate principles of either approach into the other to leverage their respective advantages.

In **Paper A** – *How Useful is Learning in Mitigating Mismatch between Digital Twins and Physical Systems?*, we, hence, compared the usefulness of learning for restoring optimal control under model-system mismatches in terms of sample efficiency, best performance within a limited sampling budget, and regret. In the context of direct input adaptation for static, high-dimensional, continuous, and smooth performance functions, it was shown that standard reinforcement learning is easily outperformed by gradient-based blackbox optimizers and Bayesian Optimization, which

explore the system in a smarter way.

In **Paper B** – *Enhancing Digital Twins through Reinforcement Learning*, we, accordingly, proposed a deep RL algorithm to adapt the digital twin's control policy derived from an erroneous model. Key feature of the algorithm is a smart exploration method that uses Bayesian artificial neural networks, which estimate uncertainties about the system's performance criterion to address safety and quality concerns.

In **Paper C** – *Online geometry assurance in individualized production by feedback control and model calibration of digital twins*, we re-framed the mismatch problem between digital twin and system as a state observation problem. A dynamic modifier model for the drifting mismatch was introduced and adapted by a novel Kalman filtering-based method that balances exploitation and exploration. While our evaluations illustrate the difficulty of balancing exploitation and exploration, they also show that this method can significantly improve performance under drifting model-system mismatches.

In **Paper D** – *Relevant Safety Falsification by Automata Constrained Reinforcement Learning*, we presented a principled method for introducing behavioral specifications into the RL performance criterion to guide and restrict the exploration process. Although, the proposed method leads to control policies that eventually satisfy the specifications, it fundamentally relies on a trial-and-error approach that necessitates violating the specifications to learn about them.

In **Paper E** – *Adaptive Energy Optimization of Flexible Robot Stations*, we demonstrated a promising state-of-the-art method for combining model-based and data-driven control in the context of energy optimization under static model-system mismatches. This method uses a system model to enforce behavioral constraints while identifying system parameters through reinforcement learning.

6.1 Conclusions

To concisely answer **RQ1** – *How can machine learning be used to mitigate the model-system mismatch in digital twins, while minimally altering the provided solution?*, we recapitulate the three possible strategies found in the literature: for (1) direct input adaptation, (2) modifier learning, and (3) model parameter identification. Direct input adaptation and modifier learning may be implemented as extensions of the digital twin model, which may be advantageous under some circumstances. Our research suggests that learning-based direct input adaptation is of limited usefulness, though, in the systemic mismatch case (Paper A), but that it might have some merit in in-

dividualized production control (Paper B). Modifier learning appears promising for individualized production control under dynamically drifting mismatches (Paper C). Model parameter identification, in contrast, requires the mismatch mitigation mechanism to have access to adjustable parameters within the digital twin model. In Paper E, we indicate the connection between the digital twin concept and a recent adaptive MPC method, that uses RL to identify model parameters that result in a good control performance. Judging from our research, this method seems promising.

In turn, our answer to **RQ2** – *How can prior model-based knowledge be introduced in reinforcement learning to improve sample-efficiency, explainability, and safety of the learned control policy?* is twofold in this thesis: (1) as specification, and (2) as model class in function approximation. In Paper D, we presented a principled method for incorporating automata specifications into RL. Note here that this method also extends to specifications given in temporal logic, if they can be converted into an automaton by any available method. While our application in Paper D benefits from the trial-and-error nature of our proposed approach, in general, this method may quickly show its limitations in physical systems such as in manufacturing, due to the possibly required repeated violations of the specification during learning. The method used in Paper E, in contrast, is capable of incorporating prior model-based knowledge about the system's dynamics, constraints, and its performance function as model class in RL function approximation. Our research highlights the benefit of this method in a new application area.

In summary, the adaptive model predictive control method of Paper E seems promising for both the model-based and the learning-based approach to intelligent automation. It is in order to emphasize here that these research findings have been the outcome of a research approach that has predominantly relied on case studies as research design. The presented findings may thus possess only limited generalization power. This research design appeared appropriate, however, for exploring the identified research gap of this thesis more qualitatively and has led to the development of methods that are believed to be generic to a large extend. Still, the findings are provisional to a more comparative evaluation of all approaches on a representative set of engineering problems within intelligent automation. This could be understood as the outstanding final descriptive study II phase in the design research methodology by Blessing and Chakrabarti [47]. Its outcome may very well be – as so often in design research – that the right tool needs to be chosen for the right task. Based on the author's understanding at the time of writing, though, the adaptive model predictive control method may be the right tool for many tasks within intelligent automation.

6.2 Future Work

Beside the previously mentioned comparative evaluation of the presented approaches in the style of a descriptive study II (see [47]), three other areas for future work are evident.

Firstly, the convergence and co-evolution of modular digital twins may be an interesting area for future work. Digital twins are comprehensive models of complex systems, and it is thus often proposed in the literature to build digital twins as collections of modular components. When several of these components pursue an adaptive or learning approach to "twinning", the system appears as time-varying and drifting for each single learning module. This poses an additional challenge to the learning algorithm. The adaptation of a modular digital under model-system mismatches may thus be an interesting multi-agent control problem to explore further.

Secondly, further research may be aimed at developing practical guidelines as to when to use which mismatch mitigation approach. This thesis touched upon three mitigation strategies: direct input adaptation, modifier learning, and model parameter identification. The adaptive MPC method within model parameter identification may be considered the method truest to the digital twin concept. However, this method requires the computation of gradients of the digital twin. This may be a challenge to some digital twin implementations. Direct input adaptation methods, in contrast, can be implemented independently of the digital twin, but it is not clear how they may extend to dynamic systems or varying mismatch scenarios. The modifier learning approach seems well suited for varying mismatch scenarios, but may be too computationally costly in static mismatch scenarios. These considerations may be further investigated and elaborated on.

And thirdly, hybrid architectures might be further explored that combine the deliberative advantages of a model-based approach with the quick inference capabilities of data-driven function approximation. Research suggests that this is prevalent also in human decision making [164]. The model-based part in that may be, for instance, an expressive, but computationally heavy, adaptive model predictive controller (e.g. a digital twin), whereas the data-driven inference could be a deep policy network, trained, for example, in the imitation learning flavour presented in [146]. Such an approach could give good control performance, while gradually and safely reducing the computational burden over time.

References

[1] Y. Koren, *The global manufacturing revolution: product-process-business integration and reconfigurable systems*. John Wiley & Sons, 2010.

[2] S. J. Hu, J. Ko, L. Weyand, H. A. ElMaraghy, T. K. Lien, Y. Koren, H. Bley, G. Chryssolouris, N. Nasr, and M. Shpitalni, "Assembly system design and operations for product variety," *CIRP Annals*, vol. 60, no. 2, pp. 715–733, 2011.

[3] United Nations, Department of Economic and Social Affairs, Population Division, *World population prospects 2022*, `https://population.un.org/wpp/Download/`, Accessed: 2022-09-21, 2022.

[4] W. Lutz, G. Amran, A. Belanger, A. Conte, N. Gailey, D. Ghio, E. Grapsa, K. Jensen, E. Loichinger, G. Marois, R. Muttarak, M. Potančoková, P. Sabourin, and M. Stonawski, *Demographic scenarios for the EU: migration, population and education*. Publications Office of the European Union, 2019.

[5] L. D. Xu, E. L. Xu, and L. Li, "Industry 4.0: State of the art and future trends," *International journal of production research*, vol. 56, no. 8, pp. 2941–2962, 2018.

[6] F. Tao, Q. Qi, L. Wang, and A. Nee, "Digital twins and cyber–physical systems toward smart manufacturing and Industry 4.0: Correlation and comparison," *Engineering*, vol. 5, no. 4, pp. 653–661, 2019.

[7] J. Krüger, T. K. Lien, and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP Annals*, vol. 58, no. 2, pp. 628–646, 2009.

[8] R. Y. Zhong, X. Xu, E. Klotz, and S. T. Newman, "Intelligent manufacturing in the context of Industry 4.0: A review," *Engineering*, vol. 3, no. 5, pp. 616–630, 2017.

[9] L. Sanneman, C. Fourie, and J. A. Shah, "The state of industrial robotics: Emerging technologies, challenges, and key research directions," *Foundations and Trends® in Robotics*, vol. 8, no. 3, pp. 225–306, 2021.

[10] S. Kumar, C. Savur, and F. Sahin, "Survey of human–robot collaboration in industrial settings: Awareness, intelligence, and compliance," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 280–297, 2020.

[11] European Commission. Directorate-General for Research and Innovation, M. Breque, L. De Nul, and A. Petridis, *Industry 5.0 : towards a sustainable, human-centric and resilient European industry*. Publications Office of the European Union, 2021.

[12] X. Xu, Y. Lu, B. Vogel-Heuser, and L. Wang, "Industry 4.0 and Industry 5.0—inception, conception and perception," *Journal of Manufacturing Systems*, vol. 61, pp. 530–535, 2021.

[13] European Commission. Directorate-General for Research and Innovation, *Industry 5.0: Human-centric, Sustainable and Resilient*. Publications Office of the European Union, 2020, ISBN: 9789276215844.

[14] A. M. Turing, "Computing machinery and intelligence," *Mind*, vol. 59, no. 236, pp. 433–460, 1950.

[15] C. E. Shannon, "Programming a Computer for Playing Chess," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 41, no. 314, pp. 256–275, 1950.

[16] J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon, "A proposal for the Dartmouth summer research project on artificial intelligence, August 31, 1955," *AI Magazine*, vol. 27, no. 4, pp. 12–12, 2006.

[17] J. McCarthy and P. Hayes, "Some philosophical problems from the standpoint of artificial intelligence," in *Machine Intelligence 4*, Edinburgh University Press, 1969, pp. 463–502.

[18] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd Edition. Pearson Education, Inc., 2010.

[19] M. Minsky, "Steps toward artificial intelligence," *Proceedings of the IRE*, vol. 49, no. 1, pp. 8–30, 1961.

[20] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems*, Springer, 2017, pp. 85–113.

[21] S. Boschert and R. Rosen, "Digital Twin - The Simulation Aspect," in *Mechatronic Futures*, Springer International Publishing, 2016, pp. 59–74.

[22] T. Lechler, E. Fischer, M. Metzner, A. Mayr, and J. Franke, "Virtual commissioning–scientific review and exploratory use cases in advanced production systems," *Procedia CIRP*, vol. 81, pp. 1125–1130, 2019.

[23] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[24] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[25] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[26] F. Tao, H. Zhang, A. Liu, and A. Y. Nee, "Digital twin in industry: State-of-the-art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2018.

[27] Y. Lu, C. Liu, I. Kevin, K. Wang, H. Huang, and X. Xu, "Digital twin-driven smart manufacturing: Connotation, reference model, applications and research issues," *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101 837, 2020.

[28] C. Zhuang, J. Liu, and H. Xiong, "Digital twin-based smart production management and control framework for the complex product assembly shop-floor," *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 1, pp. 1149–1163, 2018.

[29] B. Wang, F. Tao, X. Fang, C. Liu, Y. Liu, and T. Freiheit, "Smart manufacturing and intelligent manufacturing: A comparative review," *Engineering*, vol. 7, no. 6, pp. 738–757, 2021.

[30] F. Tao and M. Zhang, "Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing," *IEEE Access*, vol. 5, pp. 20 418–20 427, 2017.

[31] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *The International Journal of Advanced Manufacturing Technology*, vol. 94, no. 9, pp. 3563–3576, 2018.

[32] G. Zhou, C. Zhang, Z. Li, K. Ding, and C. Wang, "Knowledge-driven digital twin manufacturing cell towards intelligent manufacturing," *International Journal of Production Research*, vol. 58, no. 4, pp. 1034–1051, 2020.

[33] Y. Lu, X. Xu, and L. Wang, "Smart manufacturing process and system automation–a critical review of the standards and envisioned scenarios," *Journal of Manufacturing Systems*, vol. 56, pp. 312–325, 2020.

[34] A. Kuhnle, M. C. May, L. Schaefer, and G. Lanza, "Explainable reinforcement learning in production control of job shop manufacturing system," *International Journal of Production Research*, pp. 1–23, 2021.

[35] J. Arents and M. Greitans, "Smart industrial robot control trends, challenges and opportunities within manufacturing," *Applied Sciences*, vol. 12, no. 2, p. 937, 2022.

[36] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: A survey," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, 2020, pp. 737–744.

[37] C. Celemin, R. Pérez-Dattari, E. Chisari, G. Franzese, L. de Souza Rosa, R. Prakash, Z. Ajanović, M. Ferraz, A. Valada, and J. Kober, "Interactive imitation learning in robotics: A survey," *Foundations and Trends® in Robotics*, vol. 10, no. 1-2, pp. 1–197, 2022.

[38] J. Li, D. Pang, Y. Zheng, X. Guan, and X. Le, "A flexible manufacturing assembly system with deep reinforcement learning," *Control Engineering Practice*, vol. 118, p. 104 957, 2022.

[39] C. Li, P. Zheng, Y. Yin, B. Wang, and L. Wang, "Deep reinforcement learning in smart manufacturing: A review and prospects," *CIRP Journal of Manufacturing Science and Technology*, vol. 40, pp. 75–101, 2023.

[40] M. Panzer and B. Bender, "Deep reinforcement learning in production systems: A systematic literature review," *International Journal of Production Research*, pp. 1–26, 2021.

[41] L. Wang, Z. Pan, and J. Wang, "A review of reinforcement learning based intelligent optimization for manufacturing scheduling," *Complex System Modeling and Simulation*, vol. 1, no. 4, pp. 257–270, 2021.

[42] B. M. Kayhan and G. Yildiz, "Reinforcement learning applications to machine scheduling problems: A comprehensive literature review," *Journal of Intelligent Manufacturing*, pp. 1–25, 2021.

[43] E. Bell, A. Bryman, and B. Harley, *Business research methods*. Oxford University Press, 2022.

[44] A. F. Chalmers, *What is this thing called science?* Hackett Publishing, 2013.

[45] P. Pruzan, *Research methodology: the aims, practices and ethics of science*. Springer, 2016.

[46] K. Säfsten and M. Gustavsson, *Research methodology: For engineers and other problem-solvers*. Studentlitteratur AB, 2020.

[47] L. T. Blessing and A. Chakrabarti, *DRM: A design reseach methodology*. Springer, 2009.

[48] R. Söderberg, K. Wärmefjord, J. S. Carlson, and L. Lindkvist, "Toward a digital twin for real-time geometry assurance in individualized production," *CIRP Annals*, vol. 66, no. 1, pp. 137–140, 2017.

[49] K. Wärmefjord, R. Söderberg, B. Schleich, and H. Wang, "Digital twin for variation management: A general framework and identification of industrial challenges related to the implementation," *Applied Sciences*, vol. 10, no. 10, p. 3342, 2020.

[50] R. Bohlin, J. Hagmar, K. Bengtsson, L. Lindkvist, J. Carlson, and R. Söderberg, "Data flow and communication framework supporting digital twin for geometry assurance," in *ASME International Mechanical Engineering Congress and Exposition*, American Society of Mechanical Engineers, vol. 58356, 2017, V002T02A110.

[51] K. Wärmefjord, R. Söderberg, L. Lindkvist, B. Lindau, and J. S. Carlson, "Inspection data to support a digital twin for geometry assurance," in *ASME International Mechanical Engineering Congress and Exposition*, American Society of Mechanical Engineers, vol. 58356, 2017, V002T02A101.

[52] K. Wärmefjord, R. Söderberg, B. Lindau, L. Lindkvist, and S. Lorin, "Joining in nonrigid variation simulation," in *Computer-aided Technologies–Applications in Engineering and Medicine*, InTech Rijeka, Croatia, 2016.

[53] E. Jorge, L. Brynte, C. Cronrath, O. Wigström, K. Bengtsson, E. Gustavsson, B. Lennartson, and M. Jirstrand, "Reinforcement learning in real-time geometry assurance," *Procedia CIRP*, vol. 72, pp. 1073–1078, 2018.

[54] C. Cronrath, A. R. Aderiani, and B. Lennartson, "Enhancing digital twins through reinforcement learning," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, IEEE, 2019, pp. 293–298.

[55] C. Cronrath and B. Lennartson, "How useful is learning in mitigating mismatch between digital twins and physical systems?" *IEEE Transactions on Automation Science and Engineering*, 2022.

[56] A. Sjöberg, M. Önnheim, O. Frost, C. Cronrath, E. Gustavsson, B. Lennartson, and M. Jirstrand, "Online geometry assurance in individualized production by feedback control and model calibration of digital twins," *Journal of Manufacturing Systems*, vol. 66, pp. 71–81, 2023.

[57] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

[58] A. Mörtl, M. Lawitzky, A. Kucukyilmaz, M. Sezgin, C. Basdogan, and S. Hirche, "The role of roles: Physical cooperation between humans and robots," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1656–1674, 2012.

[59] M. A. Goodrich and A. C. Schultz, "Human–robot interaction: A survey," *Foundations and Trends® in Human–Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2008.

[60] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human–robot collaboration," *Autonomous Robots*, vol. 42, no. 5, pp. 957–975, 2018.

[61] A. Hanna, S. Larsson, P.-L. Götvall, and K. Bengtsson, "Deliberative safety for industrial intelligent human–robot collaboration: Regulatory challenges and solutions for taking the next step towards Industry 4.0," *Robotics and Computer-Integrated Manufacturing*, vol. 78, p. 102 386, 2022.

[62] P. A. Lasota, T. Fong, and J. A. Shah, "A survey of methods for safe human-robot interaction," *Foundations and Trendsin Robotics*, vol. 5, no. 4, pp. 261–349, 2017.

[63] T. Arai, R. Kato, and M. Fujita, "Assessment of operator stress induced by robot collaboration in assembly," *CIRP Annals*, vol. 59, no. 1, pp. 5–8, 2010.

[64] A. D. Dragan, K. C. Lee, and S. S. Srinivasa, "Legibility and predictability of robot motion," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2013, pp. 301–308.

[65] A. Dragan and S. Srinivasa, "Generating legible motion," in *Robotics: Science and Systems*, 2013.

[66] ISO, *ISO 12100:2011: Safety of machinery - General principles for design - Risk assessment and risk reduction*, 2011.

[67] IEC, *IEC 61508-1:2010-1 Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 1: General requirements*, International Electrotechnical Commission, 2006.

[68] L. Hornung and C. Wurll, "Human-robot collaboration: A survey on the state of the art focusing on risk assessment," in *Berichte aus der Robotik - Robotix-Academy Conference for Industrial Robotics (RACIR) 2021*, Sep. 2021, pp. 10–17.

[69] N. Leveson, *Engineering a safer world: Systems thinking applied to safety*. MIT Press, 2011.

[70] M. Askarpour, D. Mandrioli, M. Rossi, and F. Vicentini, "SAFER-HRC: Safety analysis through formal verification in human-robot collaboration," in *International Conference on Computer Safety, Reliability, and Security*, Springer, 2016, pp. 283–295.

[71] ——, "Modeling operator behavior in the safety analysis of collaborative robotic applications," in *International Conference on Computer Safety, Reliability, and Security*, Springer, 2017, pp. 89–104.

[72] F. Vicentini, M. Askarpour, M. G. Rossi, and D. Mandrioli, "Safety assessment of collaborative robotics through automated formal verification," *IEEE Transactions on Robotics*, vol. 36, no. 1, 2019.

[73] M. Askarpour, M. Rossi, and O. Tiryakiler, "Co-simulation of human-robot collaboration: From temporal logic to 3D simulation," in *1st Workshop on Agents and Robots for Reliable Engineered Autonomy, AREA 2020*, Open Publishing Association, vol. 319, 2020, pp. 1–8.

[74] M. Rathmair, C. Luckeneder, T. Haspl, B. Reiterer, R. Hoch, M. Hofbaur, and H. Kaindl, "Formal verification of safety properties of collaborative robotic applications including variability," in *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, IEEE, 2021, pp. 1283–1288.

[75] D. Araiza-Illan, A. G. Pipe, and K. Eder, "Intelligent agent-based stimulation for testing robotic software in human-robot interactions," in *Proceedings of the 3rd Workshop on Model-Driven Robot Software Engineering*, 2016, pp. 9–16.

[76] P. Bobka, T. Germann, J. K. Heyn, R. Gerbers, F. Dietrich, and K. Dröder, "Simulation platform to investigate safe operation of human-robot collaboration systems," in *6th CIRP Conference on Assembly Technologies and Systems (CATS)*, vol. 44, 2016, pp. 187–192.

[77] T. Huck, C. Ledermann, and T. Kröger, "Virtual adversarial humans finding hazards in robot workplaces," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021.

[78] D. L. Dill, "What's between simulation and formal verification?" In *Proceedings 1998 Design and Automation Conference. 35th DAC.*, IEEE, 1998, pp. 328–329.

[79] T. P. Huck, Y. Selvaraj, C. Cronrath, C. Ledermann, M. Fabian, B. Lennartson, and T. Kröger, "Hazard analysis of collaborative automation systems: A two-layer approach based on supervisory control and simulation," *arXiv preprint arXiv:2209.12560*, 2022.

[80] A. Löcklin, M. Müller, T. Jung, N. Jazdi, D. White, and M. Weyrich, "Digital twin for verification and validation of industrial automation systems–a survey," in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, vol. 1, 2020, pp. 851–858.

[81] C. Cronrath, T. P. Huck, C. Ledermann, T. Kröger, and B. Lennartson, "Relevant safety falsification by automata constrained reinforcement learning," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, IEEE, 2022, pp. 2273–2280.

[82] World Commission on Environment and Development, *Our common future*. Oxford University Press, 1989.

[83] A. Sartal, R. Bellas, A. M. Mejías, and A. García-Collado, "The sustainable manufacturing concept, evolution and opportunities within Industry 4.0: A literature review," *Advances in Mechanical Engineering*, vol. 12, no. 5, p. 1 687 814 020 925 232, 2020.

[84] C. Cronrath, B. Lennartson, and M. Lemessi, "Energy reduction in paint shops through energy-sensitive on-off control," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, IEEE, 2016, pp. 1282–1288.

[85] G. Carabin, E. Wehrle, and R. Vidoni, "A review on energy-saving optimization methods for robotic and automatic systems," *Robotics*, vol. 6, no. 4, p. 39, 2017, ISSN: 2218-6581.

[86] J. M. Rödger, N. Bey, and L. Alting, "The Sustainability Cone – A holistic framework to integrate sustainability thinking into manufacturing," *CIRP Annals - Manufacturing Technology*, vol. 65, no. 1, pp. 1–4, 2016, ISSN: 17260604.

[87] C. Gahm, F. Denz, M. Dirr, and A. Tuma, "Energy-efficient scheduling in manufacturing companies: A review and research framework," *European Journal of Operational Research*, vol. 248, no. 3, pp. 744–757, 2016.

[88] R. Menghi, A. Papetti, M. Germani, and M. Marconi, "Energy efficiency of manufacturing systems: A review of energy assessment methods and tools," *Journal of Cleaner Production*, vol. 240, p. 118 276, 2019.

[89] IEA, "Energy efficiency 2019," IEA, 2019.

[90] M. Hovgard, B. Lennartson, and K. Bengtsson, "Applied energy optimization of multi-robot systems through motion parameter tuning," *CIRP Journal of Manufacturing Science and Technology*, vol. 35, pp. 422–430, 2021.

[91] D. Meike, M. Pellicciari, and G. Berselli, "Energy efficient use of multirobot production lines in the automotive industry: detailed system modeling and optimization," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 3, pp. 798–809, 2014, ISSN: 15455955.

[92] S. Riazi, O. Wigström, K. Bengtsson, and B. Lennartson, "Energy and peak power optimization of time-bounded robot trajectories," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 646–657, 2017.

[93] M. Hovgard, B. Lennartson, and K. Bengtsson, "Energy-optimal timing of stochastic robot stations in automotive production lines," in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, 2022, pp. 1–7.

[94] P. Rohdin and P. Thollander, "Barriers to and driving forces for energy efficiency in the non-energy intensive manufacturing industry in Sweden," *Energy*, vol. 31, no. 12, pp. 1836–1844, 2006.

[95] N. Sundström, O. Wigström, and B. Lennartson, "Conflict between energy, stability, and robustness in production schedules," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 658–668, 2017.

[96] A. Giret, D. Trentesaux, and V. Prabhu, "Sustainability in manufacturing operations scheduling: A state of the art review," *Journal of Manufacturing Systems*, vol. 37, pp. 126–140, 2015.

[97] M. Hovgard, C. Cronrath, and B. Lennartson, "Adaptive energy optimization of flexible robot stations," *Revised version submitted to IEEE Transactions on Automation Science and Engineering*, 2023.

[98] R. Rosen, G. Von Wichert, G. Lo, and K. D. Bettenhausen, "About the importance of autonomy and digital twins for the future of manufacturing," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 567–572, 2015.

[99] M. Shafto, M. Conroy, R. Doyle, E. Glaessgen, C. Kemp, J. LeMoigne, and L. Wang, "Modeling, simulation, information technology & processing roadmap," *National Aeronautics and Space Administration*, vol. 32, pp. 1–38, 2012.

[100] "Automation systems and integration — Digital twin framework for manufacturing," International Organization for Standardization, Geneva, CH, Standard, Oct. 2021.

[101] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018.

[102] C. Cimino, E. Negri, and L. Fumagalli, "Review of digital twin applications in manufacturing," *Computers in Industry*, vol. 113, p. 103 130, 2019.

[103] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: Enabling technologies, challenges and open research," *IEEE Access*, vol. 8, pp. 108 952–108 971, 2020.

[104] K. Ding, F. T. Chan, X. Zhang, G. Zhou, and F. Zhang, "Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors," *International Journal of Production Research*, vol. 57, no. 20, pp. 6315–6334, 2019.

[105] A. Sharma, E. Kosasih, J. Zhang, A. Brintrup, and A. Calinescu, "Digital twins: State of the art theory and practice, challenges, and open research questions," *Journal of Industrial Information Integration*, p. 100 383, 2022.

[106] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE Access*, vol. 5, pp. 2050–2062, 2017.

[107] Modelica Association. (2020). "Functional Mock-up Interface," [Online]. Available: `https://fmi-standard.org/` (visited on 02/17/2020).

[108] L. Wright and S. Davidson, "How to tell the difference between a model and a digital twin," *Advanced Modeling and Simulation in Engineering Sciences*, vol. 7, no. 1, pp. 1–13, 2020.

[109] Z. Huang, Y. Shen, J. Li, M. Fey, and C. Brecher, "A survey on AI-driven digital twins in Industry 4.0: Smart manufacturing and advanced robotics," *Sensors*, vol. 21, no. 19, p. 6340, 2021.

[110] K. J. Åström and B. Wittenmark, *Adaptive control*. Courier Corporation, 2013.

[111] L. Ljung, *System Identification : Theory for the User*. 2. ed. Prentice Hall, 1999, ISBN: 0136566952.

[112] A. Gosavi, *Simulation-Based Optimization*. Boston, MA: Springer US, 2015, vol. 55, ISBN: 978-1-4899-7490-7.

[113] C. Audet and W. Hare, *Derivative-free and Blackbox Optimization*. Springer, 2017.

[114] J. N. Hooker, *Integrated Methods for Optimization*. Boston, MA: Springer US, 2012, vol. 170, pp. 1–640, ISBN: 978-1-4614-1899-3.

[115] B. Chachuat, B. Srinivasan, and D. Bonvin, "Adaptation strategies for real-time optimization," *Computers & Chemical Engineering*, vol. 33, no. 10, pp. 1557–1567, 2009.

[116] C. Cronrath, L. Ekström, and B. Lennartson, "Formal properties of the digital twin–implications for learning, optimization, and control," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, IEEE, 2020, pp. 679–684.

[117] X. Ma, J. Cheng, Q. Qi, and F. Tao, "Artificial intelligence enhanced interaction in digital twin shop-floor," *Procedia CIRP*, vol. 100, pp. 858–863, 2021.

[118] R. S. Sutton, A. G. Barto, and R. J. Williams, "Reinforcement learning is direct adaptive optimal control," *IEEE Control Systems Magazine*, vol. 12, no. 2, pp. 19–22, 1992.

[119] K. Åström, "Optimal control of Markov processes with incomplete state information," *Journal of Mathematical Analysis and Applications*, vol. 10, no. 1, pp. 174–205, Feb. 1965, ISSN: 0022-247X.

[120] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, 2nd. MIT Press, 2018, p. 552, ISBN: 9780262039246.

[121] J. Garcia and F. Fernandez, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1437–1480, 2015.

[122] M. G. Bellemare, W. Dabney, and R. Munos, "A distributional perspective on reinforcement learning," in *International Conference on Machine Learning*, PMLR, 2017, pp. 449–458.

[123] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[124] D. P. Bertsekas, *Reinforcement Learning and Optimal Control*. Athena Scientific, 2019, p. 388, ISBN: 978-1-886529-39-7.

[125] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Handbook of Reinforcement Learning and Control*, pp. 321–384, 2021.

[126] P. Hernandez-Leal, M. Kaisers, T. Baarslag, and E. M. de Cote, "A survey of learning in multiagent environments: Dealing with non-stationarity," *arXiv preprint arXiv:1707.09183*, 2017.

[127] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, Sep. 2013, ISSN: 0278-3649.

[128] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in AI safety," *arXiv preprint arXiv:1606.06565*, 2016.

[129] A. Irpan, *Deep Reinforcement Learning Doesn't Work Yet*, `https://www.alexirpan.com/2018/02/14/rl-hard.html`, 2018.

[130] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2, pp. 235–256, 2002.

[131] A. D. Tijsma, M. M. Drugan, and M. A. Wiering, "Comparing exploration strategies for Q-learning in random stochastic mazes," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, 2016, pp. 1–8.

[132] H. Tang, R. Houthooft, D. Foote, A. Stooke, O. Xi Chen, Y. Duan, J. Schulman, F. DeTurck, and P. Abbeel, "#exploration: A study of count-based exploration for deep reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[133] H. Van Hasselt, Y. Doron, F. Strub, M. Hessel, N. Sonnerat, and J. Modayil, "Deep reinforcement learning and the deadly triad," *arXiv preprint arXiv:1812.02648*, 2018.

[134] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[135] J. Ramírez, W. Yu, and A. Perrusquía, "Model-free reinforcement learning from expert demonstrations: A survey," *Artificial Intelligence Review*, vol. 55, no. 4, pp. 3213–3241, 2022.

[136] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband, *et al.*, "Deep Q-learning from demonstrations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[137] M. Vecerik, T. Hester, J. Scholz, F. Wang, O. Pietquin, B. Piot, N. Heess, T. Rothörl, T. Lampe, and M. Riedmiller, "Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards," *arXiv preprint arXiv:1707.08817*, 2017.

[138] S. Russell, "Learning agents for uncertain environments," in *Proceedings of the 11th Annual Conference on Computational Learning Theory*, 1998, pp. 101–103.

[139] A. Y. Ng and S. Russell, "Algorithms for inverse reinforcement learning," in *ICML*, vol. 1, 2000, p. 2.

[140] S. Arora and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress," *Artificial Intelligence*, vol. 297, p. 103 500, 2021.

[141] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the 21st International Conference on Machine Learning*, 2004, p. 1.

[142] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[143] M. Laskey, S. Staszak, W. Y.-S. Hsieh, J. Mahler, F. T. Pokorny, A. D. Dragan, and K. Goldberg, "SHIV: Reducing supervisor burden in DAgger using support vectors for efficient learning from demonstrations in high dimensional state spaces," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 462–469.

[144] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg, "DART: Noise injection for robust imitation learning," in *Conference on Robot Learning*, PMLR, 2017, pp. 143–156.

[145] J. Zhang and K. Cho, "Query-efficient imitation learning for end-to-end simulated driving," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[146] C. Cronrath, E. Jorge, J. Moberg, M. Jirstrand, and B. Lennartson, "BAgger: A bayesian algorithm for safe and query-efficient imitation learning," in *Machine Learning in Robot Motion Planning–IROS 2018 Workshop*.

[147] B. Zheng, S. Verma, J. Zhou, I. W. Tsang, and F. Chen, "Imitation learning: Progress, taxonomies and challenges," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–16, 2022.

[148] W. Sun, J. A. Bagnell, and B. Boots, "Truncated horizon policy search: Combining reinforcement learning & imitation learning," *arXiv preprint arXiv:1805.11240*, 2018.

[149] C.-A. Cheng, X. Yan, N. Wagener, and B. Boots, "Fast policy learning through imitation and reinforcement," *arXiv preprint arXiv:1805.10413*, 2018.

[150] R. S. Sutton, "Integrated architectures for learning, planning, and reacting based on approximating dynamic programming," in *Machine Learning Proceedings 1990*, Elsevier, 1990, pp. 216–224.

[151] ——, "Dyna, an integrated architecture for learning, planning, and reacting," *ACM Sigart Bulletin*, vol. 2, no. 4, pp. 160–163, 1991.

[152] T. Wang, X. Bao, I. Clavera, J. Hoang, Y. Wen, E. Langlois, S. Zhang, G. Zhang, P. Abbeel, and J. Ba, "Benchmarking model-based reinforcement learning," *arXiv preprint arXiv:1907.02057*, 2019.

[153] V. Feinberg, A. Wan, I. Stoica, M. I. Jordan, J. E. Gonzalez, and S. Levine, "Model-based value estimation for efficient model-free reinforcement learning," *arXiv preprint arXiv:1803.00101*, 2018.

[154] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[155] M. Zanon and S. Gros, "Safe reinforcement learning using robust MPC," *IEEE Transactions on Automatic Control*, vol. 66, no. 8, pp. 3638–3652, 2020.

[156] A. Plaat, W. Kosters, and M. Preuss, "High-accuracy model-based reinforcement learning, a survey," *arXiv preprint arXiv:2107.08241*, 2021.

[157] A. S. Polydoros and L. Nalpantidis, "Survey of model-based reinforcement learning: Applications on robotics," *Journal of Intelligent & Robotic Systems*, vol. 86, no. 2, pp. 153–173, 2017.

[158] S. Gros and M. Zanon, "Data-driven economic NMPC using reinforcement learning," *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 636–648, 2019.

[159] L. Grüne and J. Pannek, *Nonlinear model predictive control*. Springer, 2017.

[160] T. Faulwasser, L. Grüne, and M. A. Müller, *Economic Nonlinear Model Predictive Control*. Now Foundations and Trends, 2018.

[161] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[162] Y. Wu, R. Shariff, T. Lattimore, and C. Szepesvári, "Conservative bandits," in *International Conference on Machine Learning*, 2016, pp. 1254–1262.

[163] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Transactions on Automatic Control*, vol. 37, no. 3, pp. 332–341, 1992.

[164] D. Kahneman, *Thinking, fast and slow*. Farrar, Straus and Giroux, 2011.