BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
FACULTY OF MECHANICAL ENGINEERING
DEPARTMENT OF POLYMER ENGINEERING

OPTIMIZATION OF THE FILLING AND HOLDING PHASE OF INJECTION MOLDING TECHNOLOGY USING REINFORCEMENT LEARNING

THESIS BOOK OF PHD DISSERTATION

MADE BY:
RICHÁRD DOMINIK PÁRIZS
MECHANICAL ENGINEER (MSC.)

SUPERVISOR:
DR. DÁNIEL TÖRÖK

2025

The review and the defense report of the doctoral dissertation can be viewed at the Dean's Office of the Faculty of Mechanical Engineering of the Budapest University of Technology and Economics

1. Introduction

A new industrial revolution is taking place in the 21st century, the so-called Industry 4.0. The essence of this revolution, among other things, is to upgrade traditional manufacturing technologies with the use of sensors and big data for communication between machines and software. The aim is to use these techniques for process control by minimizing human intervention. We need new strategies and methods for these goals, which may need to be tailored to some extent for a given manufacturing technology.

Polymer processing is, of course, also affected by this new Industry 4.0 concept. According to certain aspects, additive manufacturing (like 3D printing) is part of the new revolution since new products are made by adding material instead of removing material from the preform (like cutting or milling). Besides, Industry 4.0 also affects melt forming technologies, where the hot melt fills a cavity with a geometry of the product. One of these technologies of polymer processing is injection molding, where the polymer material is heated over its melting temperature, and the melt is injected into a cooled mold with high speed and pressure. Here, the melt cools down and takes the shape of the product.

Injection molding is a relatively complex technology, where the process is divided into several phases (mold closing, filling, plastification, etc.) based on the machine parameters that are important at that phase of production. In addition, the molecular structure of the polymers themselves is complex, and it greatly influences the behavior of the given polymer. Polymers are long-chain molecules with repeating units. These repeating units, the length of the molecules, and the interaction between individual molecules affect the mechanical, geometrical, chemical, and other properties of the final product. Therefore, different polymers must be processed with varying machine settings, which makes setting up the injection molding machine more complicated. Furthermore, mold geometry also significantly impacts the injection molded product. The geometry of the runner system influences the path of the melt, while the geometry of the cooling channels affects heat dissipation at the mold, among others. All of these technological variables and inputs will affect the quality of the product.

It is not surprising that the process optimization of injection molding is a heavily researched area. Nowadays, traditional modeling techniques (like fitting linear models or analyzing factor effects) are increasingly being replaced with the so-called machine learning methods. The reason behind this tendency is that the injection molding technology is very complex. The model fitted for a specific injection molding research does not work universally in all cases. The main effect of research variables may vary significantly between different products, so a new approach was needed. Therefore, recent research is focused on the use of several sensor data, looking for hidden patterns in multidimensional datasets. These tasks are well suited to machine learning methods.

Machine learning methods include several models that are not specified for a given technology but can work with datasets of many dimensions by imitating a natural phenomenon (for example, neighboring data come from the same class). Reinforcement learning algorithms are part of the machine learning methods. These algorithms are well suited to process control tasks, hence their basic concept is that an agent interacts with the environment and learns how to make better decisions through this.

In my dissertation, I aim to use reinforcement learning for injection molding to adjust machine settings and to develop a method that can be used to set up the injection molding machine easily in the spirit of Industry 4.0, without human interaction. Within this framework, I have developed a reinforcement learning–based method, which can adjust two injection molding phases (filling and holding phase) to produce a product with a predefined quality. My goal is also to investigate and analyze the limitations of this method and to use this method with different product geometries and materials. For this reason, I will conduct injection molding and finite-element simulation experiments and use the gathered data to simulate different learning scenarios using the proposed method. Since the quality of an injection molded product depends on many factors, my aim is to develop a method that can learn from data from different sources (from different molds, materials, technological parameters, or finite-element simulations) to adjust the machine, and use this knowledge to set up the machine for a new product.

2. Literature review

The literature I have analyzed sugests that there are many shortcomings, which may pose problems due to the expectations that arise from the current industrial revolution (Industry 4.0). A clear tendency can be observed in the literature that there is an increasing need for sensor and machine data processing methods. In addition, Industry 4.0 is expected to lead to significant changes in jobs in the industry, as it happened in previous industrial revolutions. Presumably, monotonous and, in a sense, more straightforward jobs that require fewer skills in injection molding will be done by machines in the future (such as machine setup).

There are several methods in the literature for injection molding machine optimization, but these methods usually work well for a specific product–material–machine combination. Besides, the authors usually do not provide details about how their method can be used efficiently for products with different materials or geometry (without repeating the whole research). This is a huge gap because, in industrial environments, engineers and technicians usually do not work with only one type of product. In practice, changing the material or geometry of a product is quite common due to changing customer or environmental requirements (for example, the supplier company ceases to exist or they work with different material recipes).

The literature review shows that finite-element simulations can be very useful for machine setup or technological optimization. However, for such useful simulations, the material properties should be accurately measured, just as product and mold geometries must be modeled accurately. Otherwise, the results of the simulations will be different from those of real injection molding experiments. However, the measurement of material properties needs special equipment and knowledge, which are usually not present in an industrial environment. Therefore, these measuring devices should be installed, or the measurements should be bought from a research center.

Reinforcement learning can be a very useful tool for machine setup or process control thanks to its principle of operation. There is currently little research in scientific journals on the use of reinforcement learning for injection molding applications. Only a few of these articles address injection molding machine optimization. In addition, patents about this topic are usually vague on how these methods are used. These documents rarely clarify the specific algorithm the patent covers or several algorithms listed as possible solutions, but the way these algorithms can be used is unclear. These patents protect the method of using reinforcement learning rather than an algorithm idea. Those few documents that give information about the learning algorithm are usually written about Q-learning or actor-critic algorithms with the use of discounted return.

Thanks to its learning principle, the actor-critic algorithm is a very promising method for process control and machine setting adjustment. The learned policy gathered from offline data may be useful for products with different geometries because the effect of machine settings can be somewhat similar for various products; only the optimum point differs usually. For this task, with a relevant state definition, the algorithm may be able to adapt to a slightly different environment and could adjust the injection molding settings. Moreover, the algorithm could even set the machine settings if product quality deviates from the target value due to some outer noise or error. Research on process control usually uses artificial neural networks for policy or state value predictions. According to the literature, optimizing these networks requires a great amount of data, so it is worth using a different approach (for example, state aggregation), which simplifies the prediction.

Based on my literature review, the following tasks could fill most of the research gaps:

- development of an actor-critic–based method for machine setup of the filling and holding phase because these parts of the injection molding process have a great influence on the quality of the product,
- it is important to analyze the effect of preliminary experimental designs on the learning of the new method,
- it is essential to investigate the applicability of the new method for new environments, such as products with different materials or geometry,
- it is highly relevant to examine the use of finite-element simulation data for such a new method, to investigate what changes are needed to use such data.

3. Summary of the research

3.1. Materials and product geometries used

In my research, I used six different product types (with varying product geometries or gate system design) to investigate the filling and holding phase optimization of the injection molding process (Figure 1). These products included simple plates with a nominal dimension of 80x80 mm and thicknesses ranging from 1 mm to 2.5 mm. These products had a film gate („A" and „C" type gate system) or edge gates at the neighboring corners of the product („B" type gate system). There were two more complex plate-like products: a 16-cavity small lid product and a larger lid product. During my experiments, I injection molded products with three different materials: acrylonitrile butadiene styrene (ABS), polypropylene (PP) and polylactic acid (PLA).

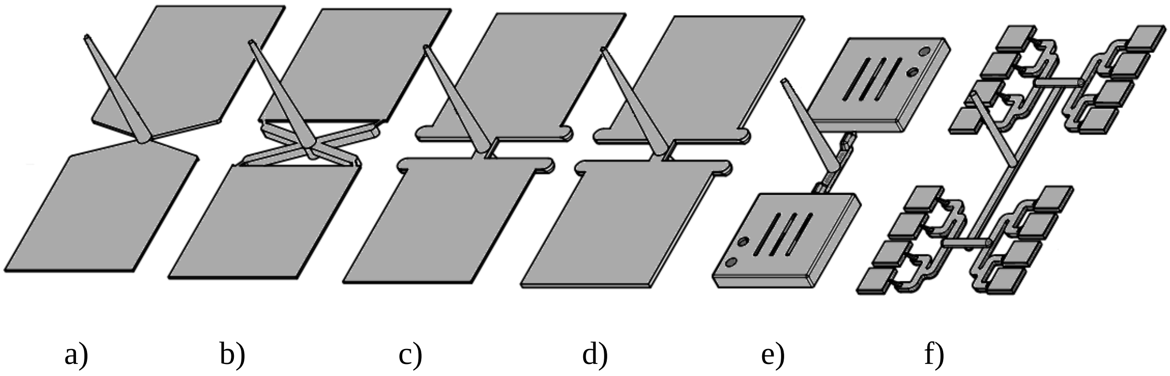


Figure 1 The different type of products used for the research
a) and b) plate with 1.2 mm thickness and with a gate system of „A" and „B", respectively,
c) and d) plate with a gate system of „C" with 1 mm and 2.5 mm, respectively,
e) lid product, f) small lid product

I produced each product type for the filling experiments with different settings of switch-over volumes and injection rates. For the filling optimization, I defined the quality criteria based on the following variables: the maximum pressure in the cavity, the filling time of the cavity, and the image of the product. For each product type, I defined the ideal values of these variables. The image of the product was taken after the opening of the mold, but before ejecting. The quality of a product is the state, which was used by the learning algorithm to learn how to optimize the filling phase. For the holding phase optimization, the quality criteria was product weight. For this optimization, the algorithm learnt how to change the holding pressure and holding time settings.

3.2. The learning algorithm and the analysis of the method

For my investigation, I programmed and modified an actor-critic algorithm. This algorithm uses a softmax function for policy estimation and a bilinear function for state value estimation. Therefore, the action set, state aggregation limits, quality tolerance, the learning rate, and the number of learning steps are all important parameters of the algorithm. Since the algorithm has a stochastic policy, I analyzed different learning scenarios for the learning cases. With each algorithm setting, I used at least 100 learning scenarios to estimate the performance of the algorithm. In addition, I programmed a restart mechanism into the algorithm, which reset the algorithm for the initial machine settings after a given learning step. Another innovation, compared to the original actor-critic algorithm, is that my algorithm can use greedy actions. I defined these greedy actions with the number of best actions and the cumulative probability of those actions. In theory, the algorithm changes the settings of the injection molding machine through actions and learns how the quality of the product changes. The goal of the adjustment is to set up the injection molding machine (for the predefined quality) with as few injection molding cycles as possible.

My optimization method consists of two main parts (see Figure 2). The first part is called pre-learning, where the algorithm learns how to adjust the machine settings based on offline data of a given product. During pre-learning, the algorithm takes 100 learning scenarios (with the same algorithm parameters, and dataset) from which it will have a prior knowledge of the policy and state value estimation parameters.

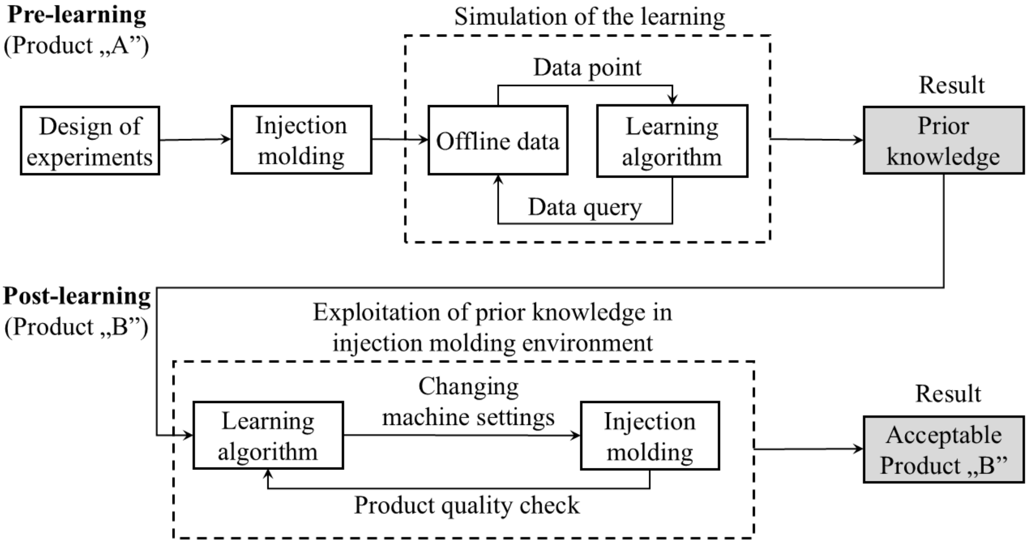
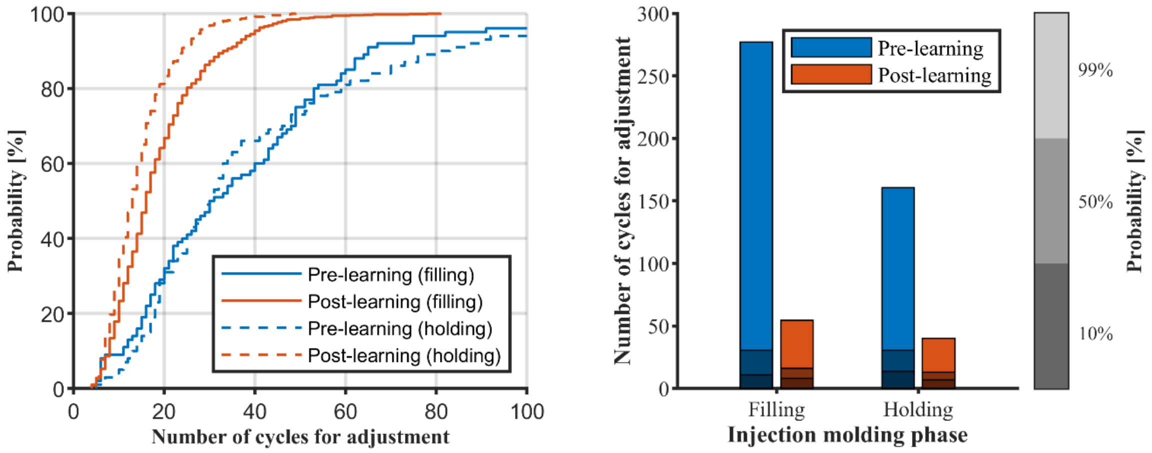


Figure 2 The flow chart of my optimization process control method

The second part is called the post-learning, where the algorithm uses the previously collected prior knowledge for online machine setting adjustments for a new product. I made the post-learning on offline data in my analysis with different learning scenarios. In addition, I conducted experiments for validation, where post-learning was performed with online injection molding data.

3.3. Results

To get a clear understanding of the performance of the algorithm, I made 1000 learning scenarios for post-learning. Since these learning scenarios are somewhat a simulation of the machine setting adjustment, I investigated how many learning steps (or we could say injection molding cycles) are needed to set up the machine in these scenarios. The metric of the performance of the algorithm was the number of required cycles to adjust product quality to the predefined value (with a given tolerance). From the 1000 learning scenarios, I made a cumulative distribution of probability to compare different settings. For the filling and holding phase optimization, these curves can be seen in Figure 3. It is clear that post-learning needs fewer adjustment steps than pre-learning (for both the filling and holding phase optimization).

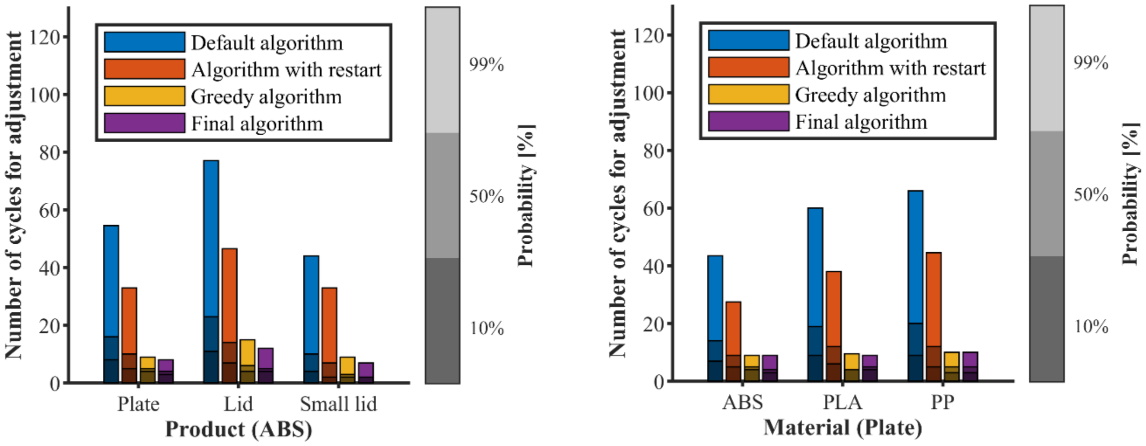


a) b)

Figure 3 a) the cumulative probability of adjustment steps for pre- and post-learning b) the adjustment steps for 10%, 50%, and 99% probability

3.3.1. The effect of hyperparameter optimization

Firstly, I investigated the effect of hyperparameters on my optimization method. Among the optimized parameters were the learning rate, the action set, the number of learning scenarios during pre-learning, and the type of state aggregation. In addition, I investigated how post-learning performance changes if I modify the actor-critic algorithm with the restart mechanism (for pre-learning) and with greedy actions (for post-learning). From the results (Figure 4), it is clear that these modifications greatly improved the performance individually and together. Therefore, the final algorithm includes these modifications, and later, I will use this algorithm for the evaluation.

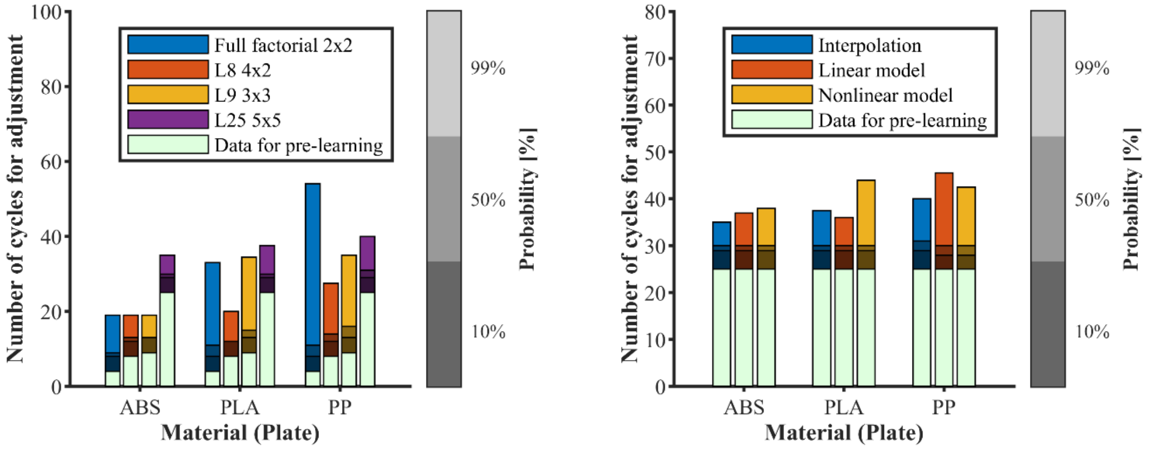


a) b)

Figure 4 The difference between the performance of different algorithms a) for different types of product, b) for different materials

3.3.2. The effect of experimental design and internal point estimation for pre-learning

The dataset used for pre-learning can have a great influence on the performance of the algorithm. Therefore, I made learning scenarios where the prior knowledge (derived from pre-learning) was produced with different experimental designs. A total of 15 different experimental designs were used to teach the algorithm, and each time, post-learning was performed with the (original) full dataset. In addition, I analyzed the effect of different models for the estimation of internal point (data between experimental points). I used linear interpolation, a linear model and a nonlinear model for this task. My results showed that there is no outstandingly good modeling technique or experimental design for this learning task (Figure 5). The effectiveness of post-learning depends on the representation of the target domain (and its neighboring states) during the pre-learning phase. In some cases, the target range was so narrow that the algorithm could not find an action that would produce an acceptable product.



a)

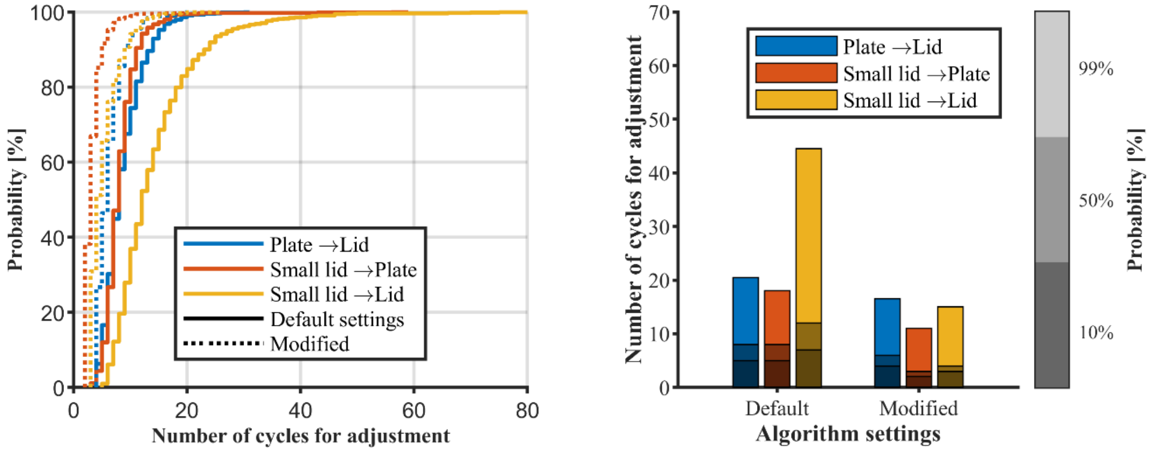
b)

Figure 5 The effect of different a) experimental designs and b) modeling techniques on post-learning with different materials

3.3.3. The effect of learning from a different product

The essence of my method is that the learning algorithm learns (during pre-learning) the effect of technological parameters, whose effects are somewhat similar for different products. Therefore, the algorithm may be able to use this prior knowledge in a new environment (for a new product). Hence, I investigated different learning scenarios where prior knowledge was derived from a different product than the one used for post-learning.

My analysis showed that because the machine parameters have a different effect on a different product, machine adjustment took slightly more cycles for some product combinations. However, in these cases, learning can be improved by the modifications of the inputs of the optimization. I strongly recommend the use of the upload mechanism for post-learning. This way, the effect of some of the differences in the learning environment can be compensated for. My results showed that in the case of large volumetric differences (between the products used for pre-and post-learning), the performance of post-learning can be improved by modifying the switch-over volume actions in proportion to these product volumes. The effect of these actions can greatly depend on the volume of the product, and this phenomenon becomes more significant the larger the volume difference is. It can also be problematic if the tolerance of post-learning (or the tolerance field in the research space) is narrow compared to the possible actions. In such a case, the performance of the algorithm can be improved if the tolerance of pre-learning is narrow, too. The background of this is that the algorithm can learn to use smaller actions near the target region if the tolerance is narrow. In my results, the use of at least one of the previously mentioned modifications can decrease the required cycle number for post-learning (Figure 6).



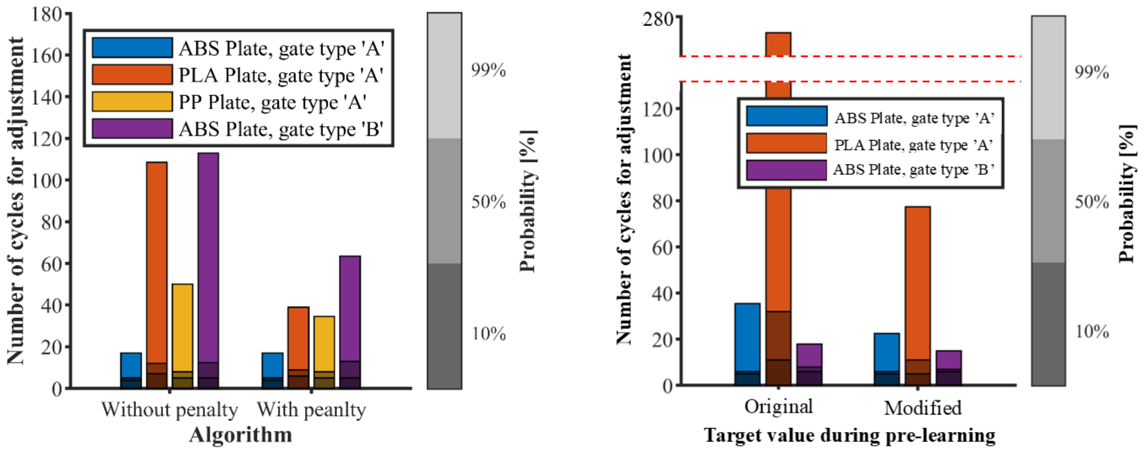
a) b)

Figure 6 The improvement of the algorithm with one of the modifications a) the complete probability distribution, b) the minimum required number of learning steps for 10%, 50%, and 99% of the cases

3.3.4. The effect of pre-learning from finite-element simulation data

The use of finite-element simulations is common in the injection molding industry because these simulations give a good overview of the effect of the process variables. Therefore, I used inaccurate simulations with few boundary conditions to make datasets for different products and materials. The main goal of my investigation was to analyze the impact of these datasets on the prior knowledge and post-learning. For filling optimization, my results showed that the performance of the learning algorithm can be improved by using a penalty for actions that try to go outside the search space. This modification may be necessary because some simulation software cannot overfill the cavity during the filling simulation. Still, this overfilling is common during actual injection molding. With this penalty, the algorithm will find its way from the edge of the search space faster, where the setting combinations of overfilled states can be found.

During the holding phase optimization, there were some cases where the simulated dataset did not contain the target product weight. My results showed that changing the pre-learning target value can be a good solution for such a problem. In this way, the algorithm can learn how to choose actions near the target value, and later, it can use this knowledge to adjust the machine settings. In my results, some of these modifications improved the performance of the algorithm, so it required 200 fewer injection molding cycles to optimize the process (see Figure 7).



a) b)

Figure 7 The results of modificiations in the algorithm using simulation data for pre-learning a) the required cycle number for post-learning during filling phase optimization, and b) holding phase optimization

4. Thesis statements

Thesis statement I

I developed an optimization process control method that can be used to adjust the injection molding parameters for thermoplastic plate-like products for the filling phase based on maximum pressure in the cavity, cavity fill time, and image of the product and the holding phase based on product weight to a value within the tolerance field. This method is based on an actor-critic algorithm modified by me. *The algorithm can modify the technological variables based on the characteristics of the product and using state aggregation and discrete action sets. The essence of the method is that the algorithm can be trained with an offline dataset (pre-learning). The results of this pre-learning is prior knowledge, which can be used to adjust the injection molding settings for a different product or a default machine setting (post-learning). With this method, the machine adjustment requires significantly fewer injection molding cycles [1-5].*

I proved the applicability of this method using six different product or runner geometries and three different materials (ABS-Terluran GP-35, PP-Tipplen H145F, PLA-Ingeo 3100HP). For my measurements, I used four different injection molding machines (Arburg Allrounder 270S 400-170, Arburg Allrounder 320C 400-170, Arburg Allrounder 420C 1000-290, Arburg Allrounder 470A 1000-290). In my analysis, the use of prior knowledge resulted in an average of 195 fewer adjustment cycles (minimum: 34, maximum:427, median: 165) for the filling phase optimization (99% of the learning cases). The use of prior knowledge resulted in an average of 203 fewer adjustment cycles (minimum: 55, maximum: 378, median: 205) for the holding phase optimization (99% of the learning cases).

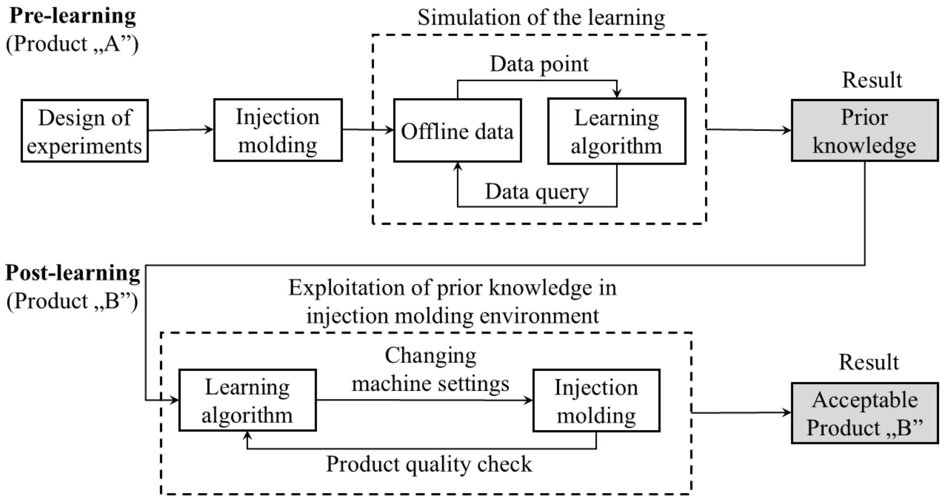


Figure 8 The flow chart of my optimization process control method

Thesis statement II

I proved in the case of various plate-like products and materials, with the simulation of learning and online measurements, that ***in my optimization process control method, the combination of greedy actions at post-learning and the restart mechanism at pre-learning results in a significant reduction of the required adjustment cycles of injection molding.*** *I also confirmed that machine adjustment is the fastest when there are 30 restart times with 100 restart steps, and there are only five greedy actions with a cumulative probability of 90% [1, 2, 4, 5].*

I proved my claim using four product geometries and three materials (ABS-Terluran GP-35, PP-Tipplen H145F, PLA-Ingeo 3100HP). For my measurements, I used four different injection molding machines (Arburg Allrounder 270S 400-170, Arburg Allrounder 320C 400-170, Arburg Allrounder 420C 1000-290, Arburg Allrounder 470A 1000-290). In my analysis, these modifications resulted in an average of 53 fewer adjustment cycles (minimum: 35, maximum:65, median: 54) for the filling phase optimization (99% of the learning cases). The use of these modifications resulted in an average of 40 fewer adjustment cycles (minimum: 13, maximum: 71, median: 38) for the holding phase optimization (99% of the learning cases).

Thesis statement III

I proved in the case of various plate-like products and materials, and with different experimental designs and predictive models, that ***in my optimization process control method, the effectiveness of prior knowledge is based not primarily on the experimental design and type of model for estimating data points between measurement points (during pre-learning), but rather on the extent of the representation of target state and its environment based on these designs and models.*** *The state space used for pre-learning must include the target state in a way that the algorithm can reach it through its actions from any point of the state space. Also, the state space must include at least the neighboring two states (under and above the target state), otherwise the required number of adjustment cycles may increase significantly [1,4-8].*

I proved my claim using four different product or runner geometries and three different materials (ABS-Terluran GP-35, PP-Tipplen H145F, PLA-Ingeo 3100HP). For my measurements, I used four different injection molding machines (Arburg Allrounder 270S 400-

170, Arburg Allrounder 320C 400-170, Arburg Allrounder 420C 1000-290, Arburg Allrounder 470A 1000-290) and Moldflow Insight 2019 finite-element simulations.

Thesis statement IV

I proved in the case of various plate-like products and materials, with the simulation of learning and online measurements, that ***with my optimization process control method, injection molding settings can be adjusted for a new product with prior knowledge derived from a product of different material, or a product with (in some extent) different geometry*** *(plate-like)* ***through nearly as many cycles as with prior knowledge derived from the same product.*** *For this, the following modifications are required:*

- during pre-learning, the target domain needs to be narrowed down (based on the tolerance and stability of the injection molding process),

- the switch-over volume change needs to be modified based on the volume ratio of the two products used for the pre- and post-learning phases,

- the application of the upload mechanism, if it is suspected that the number of aggregated states known at pre-learning is smaller than the number of these states at post-learning [1, 5].

I proved my claim using six product or runner geometries and three materials (ABS-Terluran GP-35, PP-Tipplen H145F, PLA-Ingeo 3100HP). For my measurements, I used four different injection molding machines (Arburg Allrounder 270S 400-170, Arburg Allrounder 320C 400-170, Arburg Allrounder 420C 1000-290, Arburg Allrounder 470A 1000-290). The adjustment for products with different geometry/material required an average of 3 more cycles (minimum:-3, maximum:+10, median:+1) for the filling phase optimization (99% of the learning cases). The adjustment required an average of 4 more cycles (minimum:-4, maximum:+23, median:+2) for the holding phase optimization (99% of the learning cases).

Thesis statement V

***I extended my optimization process control method to use prior knowledge derived from imprecise simulations with few boundary conditions** using finite-element simulations for various plate-like products and materials, with the simulation of learning and online measurements. With the extended method, the filling and holding phase adjustment can be implemented with a similar number of products compared to the case when prior knowledge comes from real injection molding data. This requires the following modifications:*

- The penalty of the actions that go beyond the search space during the filling phase optimization,

- the application of the upload mechanism,

- the modification of the target value and aggregation limits for pre-learning during the holding phase optimization [1, 5].

I proved my claim using three product or runner geometries and three materials (ABS-Terluran GP-35, PP-Tipplen H145F, PLA-Ingeo 3100HP). For my measurements, I used two different injection molding machines (Arburg Allrounder 270S 400-170, Arburg Allrounder 320C 400-170) and Moldflow Insight 2019 finite-element simulations. The adjustment with prior knowledge derived from finite-element simulations required an average of 28 more cycles (minimum: 8, maximum: 52, median: 27) for the filling phase optimization (99% of the learning cases). The adjustment required an average of 4 more cycles (minimum:0, maximum:17, median:1) for the holding phase optimization (99% of the learning cases).

5. Publications

1. Párizs R. D., Török D.: An experimental study on the application of reinforcement learning in injection molding in the spirit of Industry 4.0. Applied Soft Computing, **167**, 112236 (2024). (https://doi.org/10.1016/j.asoc.2024.112236)
2. Párizs R. D., Török D.: Fröccsöntött termékek zsugorodásából adódó alak-deformáció kompenzálása, szerszámüreg-nyomás alapján. Polimerek, **8**(5), 153-160 (2022).
3. Párizs R. D., Török D., Ageyeva T., Kovács J. G.: Machine learning in injection molding: an industry 4.0 method of quality prediction. Sensors, 22(7), 2704 (2022). (https://doi.org/10.3390/s22072704)
4. Párizs R. D., Török D., Ageyeva T., Kovács J. G.: Multiple in-Mold sensors for quality and process control in injection molding. Sensors, **23**(3), 1735 (2023). (https://doi.org/10.3390/s23031735)
5. Párizs R. D., Török D.: How to use prior knowledge for injection molding in industry 4.0. Results in Engineering, **23**, 102667 (2024).
6. Párizs R. D., Török D.: A gyártási paraméterek hatása a fröccsöntött termékek színhomogenitására fűtött csatornás szerszámban. Polimerek, **8**(6),259-264 (2022).
7. Kiss B., Párizs R. D., Tóth C., Török D., Kovács N. K.: Anyagextrúzió alapú additív gyártástechnológiával készült termékek anizotróp viselkedésének elemzése. Polimerek, **9**(5), 155-160 (2023).
8. Tomin M., Lengyel M. Á., Párizs R. D., Kmetty Á.: Measuring and mathematical modeling of cushion curves for polymeric foams. Polymer Testing, **117**, 107837 (2023). (https://doi.org/10.1016/j.polymertesting.2022.107837)