Advanced automatic pass schedule design for hot rolling by coupling reinforcement learning with a fast rolling model

IDZIK Christian[1,a *], GERLACH Jannik[1,b], BAILLY David[1,c] and HIRT Gerhard[1,d]

[1]Institute of Metal Forming (IBF), RWTH Aachen University, Aachen, Germany

[a]christian.idzik@ibf.rwth-aachen.de, [b]jannik.gerlach@ibf.rwth-aachen.de, [c]david.bailly@ibf.rwth-aachen.de, [d]gerhard.hirt@ibf.rwth-aachen.de

Keywords: Hot Rolling, Reinforcement Learning, Multi-Objective Optimization

Abstract. Rolling is a well-established forming process for producing finished or semi-finished products in various industries. Although highly automated, most rolling processes are designed manually by experts based on their knowledge, highly specialized heuristics and analytical process models or numerical simulations. This manual design approach does not lead to an optimization accounting for multiple objectives. Previous work [1] has shown the potential of coupling reinforcement learning (RL) with fast analytical rolling models (FRM) to optimize hot rolling processes. However, the designed pass schedules do not robustly reach the desired final height within typical industrial tolerances. Therefore, in this paper the existing approach of coupling RL with an FRM is extended by dynamically ranges for height reductions. This extension guarantees that the target height is always reached exactly. In addition to the height reduction, the RL algorithm can determine the inter-pass time, initial slab temperature and rolling velocity. For the optimization, an objective function, called reward function, considering all relevant optimization objectives such as the final grain size and energy consumption, was developed. An exemplary training was performed for a defined starting (140 mm) and final height (25 mm). The resulting, automatically designed pass schedules reach the target height and fulfill all defined optimization objective including the required average austenite grain size.

Introduction

Hot rolling of flat products is a widely-used metalworking process, in which the material is heated above its recrystallization temperature and then passed through an arrangement of rolls to reduce its thickness. It is typically used for processing cast slabs or ingots into heavy plates or sheets with the desired geometry and material properties for various industries such as the automotive and construction industry. In fact, a large proportion of all steel (> 90%) and aluminum (> 60%) products are rolled at some point during their manufacture, as Allwood et al. [2] pointed out.

With an annual global crude steel production of over 1,900 million tons in 2021 [3], it is clear that even small process improvements such as energy savings are of significant importance. It is therefore logical to further optimize the process parameters in hot rolling e.g. the height reduction, so that desired product properties, such as final geometry, grain size and corresponding mechanical properties, are achieved more efficiently thus minimizing the energy consumption.

The product properties are determined by the initial material state, material properties like flow stress, heat transfer, microstructure evolution as well as the process parameters of each pass. These include height reduction, rolling velocity and inter-pass time among others. The challenge lies in the fact that (hot) rolling consists of several subsequent and coupled passes with the process parameters summarized in one overarching pass schedule. These dependencies result in a change of properties during each pass, therefore indirectly influencing every subsequent pass.

Due to the complex interactions between the process parameters, the material properties as well as the processing constraints, the design and optimization of the pass schedule represents a complex multi-objective optimization problem. Despite great progress, detailed finite element simulations as well as the use of classical optimization algorithms or specialized heuristics for process design are time consuming and hence not yet applicable for process optimization in the industrial context. Therefore, process design is often only based on experience of long-time rolling mill operators or fast process models. Previous work by Scheiderer et al. [1] showed that coupling RL with a FRM is very promising. RL is applied, since after a successful training process it is capable of solving given problems, e.g. to identify strategies in terms of a pass schedule, within seconds. During the training process, the learning part of the RL algorithm, which is called agent, learns through goal oriented interaction with a virtual process environment. Within the presented work, the required data for the RL-agent is provided by the FRM. However, the final height is not robustly reached within typical industrial tolerances.

To tackle this challenge, this paper presents a workflow to ensure that the final target height is always reached exactly within the tolerance. After a brief overview of the current pass schedule design approaches and the application of RL for process optimization, the coupled approach is shown and explained. The focus lies on the newly developed reward function and workflow for robust attainment of the target height. Finally, an exemplary training is presented and discussed.

Pass schedule Design and Optimization

Various approaches to design and optimize pass schedules exist, the first objective of which is always to guarantee the attainment of a target height. Known approaches to distribute the height reduction to several passes are, for example, to set the height reduction per pass to the maximum permitted by the rolling mill limitations [4] or to aim for a uniform distribution of the rolling forces between the individual passes [5]. These approaches are usually used to design a first version of a pass schedule, which is then further optimized.

As, Özgur et al. [6] pointed out in an extensive literature review, these pass schedules are mostly designed based on expert knowledge, analytical rolling models, finite element (FE) simulations and heuristics, and are therefore rarely optimized for several objectives simultaneously. These different approaches are often specialized for specific use cases, which makes a comparison or transferability almost impossible. A similar conclusion was reached by Pandey et al. [7] after their literature review demonstrating that, despite numerous methods, expertise is always needed.

In recent years, evolutionary algorithms have been increasingly used to design and optimize pass schedules, as shown by the work of Wu et al. [8] and Hernandez et al. [9]. However, these algorithms do not learn relationships between process parameters and target variables, thus a separate design process has to be performed for each pass schedule. With the usage of RL in contrast, knowledge can be (partially) transferred to other pass schedules.

Fast Rolling Models

Fast rolling models (FRM) are able to calculate rolling forces and microstructure development for complete rolling processes within a few seconds. For this purpose, they usually consist of simplified, analytical models using semi-empirical equations to describe the material behavior. Numerous fast rolling models with different emphases can be found in the literature.

Beynon and Sellars [10] present a rolling model called SLIMMER, which is able to describe the microstructure evolution and predict the rolling force and torque during hot rolling. Inspired by their work, Lohmar et al. [11] extended a similar model to include height resolution within the workpiece and considered the influence of shear during deformation. In addition to these classical approaches, data-driven methods have also been increasingly used in recent years to model the hot rolling process, for example by Shen et al. [12] to predict rolling forces.

Reinforcement Learning for Process Optimization

Reinforcement Learning (RL) is a branch of Machine Learning, which attempts to imitate natural learning behavior through reward and punishment. It consists of an iterative learning approach, in which it learns by mapping states to actions while trying to maximize a numerical reward [13].

In recent years, RL applications sprung up across the manufacturing field with exponential publication growth year by year as Li et al. [14] stated. The authors analyzed 264 different publications between 2013 and October 2022 and found, that optimizing energy consumption as well as costs and reducing reliance on expert knowledge as the main objectives of these applications. Esteso et al [15] came to similar conclusions and add that the great majority of RL applications in production technology utilize simplified (virtual) environments with discrete action spaces. Panzer and Bender [16] also conducted a literature review and conclude that in numerous applications RL outperforms previously used heuristics or algorithms. However, they are still not yet applied in real production.

There are already several successful applications of RL for the optimization of forming processes, such as in open die forging [17] and wire hot rolling [18]. Reinisch et al. [17] coupled a process model with a RL algorithm to design and optimize pass schedules for open-die forging in terms of final ingot geometry, press force and process duration. The designed pass schedules led to executable forging processes. Moreover, Gamal et al. [18] demonstrated that RL in combination with process data identifies model parameters and thus improve model predictions for bar and wire hot rolling processes. For hot rolling, Scheiderer et al [1] published a RL approach, which can design pass schedules considering several optimization objectives. The authors used a database of simulation data to train the RL algorithm. However, the target height could not be exactly achieved, which in any case is necessary for industrial application. Therefore, in this paper, based on this previous work, the coupling is extended to such an extent that the target height is always reached exactly. For this purpose, the previously statically defined limits of the height reduction are dynamically adjusted so that they prevent the target height from being undershot.

Advanced Coupling of RL with FRM

In this chapter, the advanced coupling between a FRM and a RL algorithm is presented. Fig. 1 schematically shows the structure of the coupled approach. The approach consists of an RL agent, representing the learning part, an environment, representing the problem and a set of selectable actions within defined ranges. At each discrete time step, the RL agent perceives the current state of the environment and performs actions, which result in changes of the environment. In this concrete use case, the RL agent chooses the process parameters of the next pass based on the current slab geometry and temperature. These process parameters lead to a change of the geometry and the temperature of the slab and therefore the state of the material. These changes are calculated by the FRM based on the chosen parameters, the previous material state and the material properties such as the flow stress. The RL agent designs the pass schedule pass by pass, thus feedback can be given to the agent directly after each pass

Based on the FRM results, a numerical reward is calculated evaluating the quality of the chosen process parameters (actions). It can be either positive or negative, representing a reward or a punishment, respectively. The reward as well as the current state are passed to the RL agent, which stores the information. Based on all stored information, neural networks aim to learn relationships between the previous states, the process parameters, the resulting new state and the reward.

The goal lies in identifying a sequence of process parameters (pass schedule), which result in a maximized reward. The described steps are repeated until the desired goal, here the target height, is reached. As soon as this is the case, the pass schedule design is completed, an iteration is finished and the material state is reset to the initial state. Afterwards, another iteration of the training process is performed. The training is carried out until the designed pass schedule converges. This means that the pass schedule does not really change from iteration to iteration.

Each further iteration provides the RL algorithm with new information and therefore helps improving the choice of process parameters in further iterations. To guarantee continuous improvement of the designed process, the chosen reward function is of essential importance. Here, it e.g. tracks and rewards, whether the final grain size matches the desired one, or punishes the exceeding of the maximum rolling force of the rolling mill.

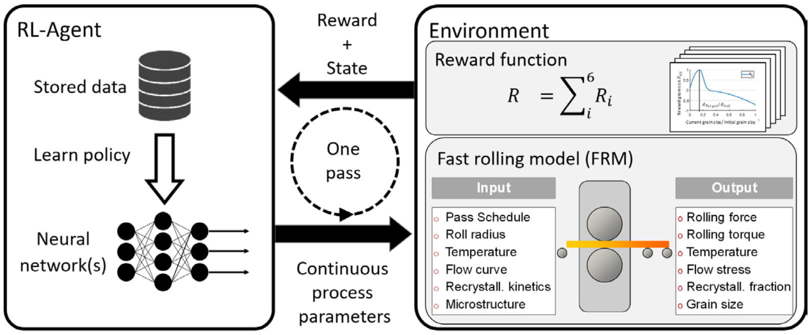

Fig. 1. Schematic depiction of the coupling of the RL algorithm with the FRM.

As previously mentioned, the pass schedule design is performed pass by pass. Hence, it follows that the reward is also calculated directly after each pass instead of only receiving feedback after a complete pass schedule. However, the RL algorithm can only process a single numerical value to evaluate all defined optimization objectives. Thus, each objective first has to be evaluated separately, before being summarized to one numerical value. Table **1** lists all considered optimization objectives.

Table 1. Overview of the optimization objectives.

Optimization objective	**Goals**
Grain size R_d	Final average austenite grain size after rolling
Force/torque R_F / R_T	No exceeding of the rolling mill limits (80% of max optimal)
Energy consumption R_E	Minimization of the energy consumption
Process time R_{time}	Minimization of the process duration
Height reduction R_{deltaH}	Minimization of total number of passes

In this paper, the reward R for each pass is composed as a weighted sum of six components R_i as shown in Eq. 1. The prioritization here lies in achieving a desired grain size as it corresponds to mechanical properties. Therefore, this reward component is provided with a higher weighting than the others. An evaluation of the height is not necessary as the target height is always achieved exactly by dynamic adjustment of the height reduction, which will be described later.

$$R = \sum_i^6 w_i \cdot R_i = 3 \cdot R_d + R_F + R_T + R_E + R_{time} + R_{deltaH} \quad (1)$$

For the definition of the individual reward components R_i, continuous functions were chosen, which scale between -1 and 1 to ensure no prioritization only by function design. To prevent the accumulation of positive rewards by simply adding a large number of passes, most reward components are defined negatively. Selected reward functions for individual objectives are described in more detail below.

During optimization, some goals are to be exactly reached ($R_d/R_F/R_T$), while others are to be minimized (R_E/R_{time}) or maximized (R_{deltaH}). Fig. **2** (left) shows the reward for the rolling force (R_F). The definition for the rolling torque (R_T) is equivalent to the one for the force. In both cases, it is important not to exceed the rolling mill limits while still using enough of the available supplies. Therefore, the reward increases up to an optimal working point (F_{opt}) sitting at 80% of the maximum rolling mill force. This ensures that unintentional exceeding of the rolling stand limits is prevented even if process deviations or material fluctuations occur. If the defined operating point is exceeded, the reward decreases and reaches a penalty of -1 at the maximum rolling mill force.

Fig. 2 (right) shows the reward for the austenite grain size (R_d). The punishment increases the further the current grain size deviates from the desired target grain size (d_{Target}). Once the grain size lies within the desired tolerance, the reward (R_d) becomes positive.

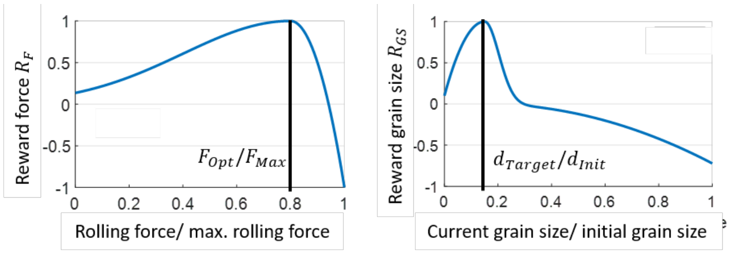

Fig. 2. Reward functions for rolling force (left) and grain size (right).

For components requiring the minimization or maximization of a given process parameter, parabolas as shown in Fig. 3 on the left for energy consumption (R_E) and on the right for height reduction (R_{deltaH}) are used. These reward components are only defined between 0 and -1 so that the agent is forced to minimize the punishment. The energy consumption is calculated considering the rolling torque, rolling speed, roll radius and the process duration for each pass.

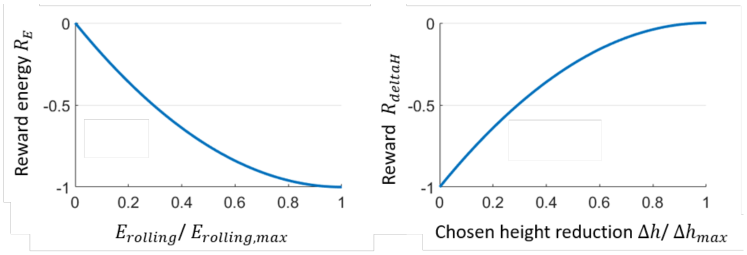

Fig. 3. Reward functions for energy consumption (left) and for height reduction (right).

In addition to the aspects described so far, the advanced coupled approach also has to ensure that the final height is exactly reached. Seeming to be particularly easy using human knowledge in process design, this has proven to be difficult for the RL algorithm in the past. In the work of Scheiderer et al. [1], the limits defined for the height reduction were the same absolute values for each pass. Therefore, the RL agent had to learn how to reach the target height, which did not always turn out to be successful. As soon as the maximum possible height reduction Δh was greater than the remaining difference between the initial height h_0 of the current pass and the target height h_{Target}, there was a risk that the pass schedule leads to heights below the target height. To prevent this, dynamic adjustments of the height reduction limits $[\Delta h_{min}, \Delta h_{max}]$ are implemented.

For the maximum permissible height reduction (Δh_{max}), two process-technological limits need to be respected. On the one hand, there is a maximum possible height reduction ensuring that no damage like cracking occurs. This limit ($\Delta h_{max,1}$) is material-dependent and typically lies at around 40 % of the initial height h_0 of the regarded pass. On the other hand, the bite condition during rolling has be taken into account. Depending on the friction coefficient μ and the roll radius r, it defines the maximum possible height reduction ($\Delta h_{max,2}$), for which the material is still drawn into the roll gap. However, since there are uncertainties regarding the friction coefficient, here, 85 % of the possible maximum height reduction is used. This ensures rollable pass. The corresponding developed method is shown in Fig. 4. Based on the two calculated maximum possible height reductions ($\Delta h_{max,1}$, $\Delta h_{max,2}$), the smaller one is selected, so that both limitations are always fulfilled. In the next step, it is checked whether the chosen height reduction is larger than the remaining height difference between the current height and the target height h_{Target}. If so, the maximum permissible height reduction Δh_{max} is lowered to this difference.

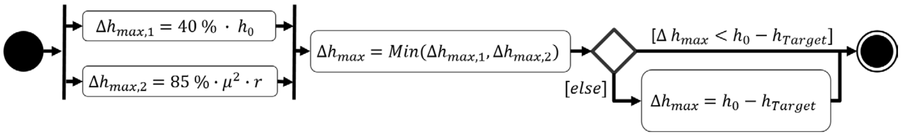
$\Delta h_{max,1} = 40\ \% \cdot h_0$
$\Delta h_{max,2} = 85\ \% \cdot \mu^2 \cdot r$
$\Delta h_{max} = Min(\Delta h_{max,1}, \Delta h_{max,2})$
$[\Delta\ h_{max} < h_0 - h_{Target}]$
$[else]$
$\Delta h_{max} = h_0 - h_{Target}$

Fig. 4. Calculation of maximum height reduction Δh_{max} to guarantee reaching the desired final height.

Regarding the minimum height reduction (Δh_{min}), there are no fixed limits. Small height reductions are principally allowed, see Fig. 5. However, they usually lead to longer process times, which are punished by the reward function as shown in Eq. 1.

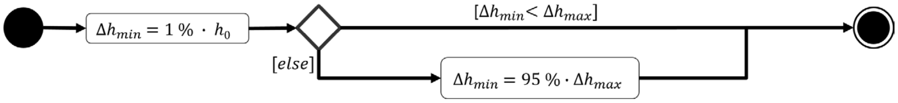
$\Delta h_{min} = 1\ \% \cdot h_0$
$[\Delta h_{min} < \Delta h_{max}]$
$[else]$
$\Delta h_{min} = 95\ \% \cdot \Delta h_{max}$

Fig. 5. Calculation of minimum height reduction Δh_{min} to guarantee reaching the desired final height.

Results: Training for Designing and Optimizing a Pass Schedule

The above described approach was used to design a pass schedule for reversing hot rolling of a S355 steel slab with an initial height of 140 mm to a target height of 25 mm on the universal rolling mill Bühler VRW-400 at the Institute of Metal Forming (IBF). Furthermore, the optimization aimed to reach a final grain size ($d_{Target} = 30\ \mu m$). The rolling mill limits were set to a force ($F_{max} = 4\ MN$) and torque ($F_{max} = 65\ kNm$), while energy consumption and process time were to be minimized. During the training, the RL agent was able to vary the process parameters within the limits shown in Table 2.

For the results shown in the following, an established RL algorithm, the Deep Deterministic Policy Gradient (DDPG) [19] was used. This algorithm uses two neural networks, allowing it to solve problems with continuous action spaces. One network estimates the cumulative long-term reward based on the current state and the chosen actions. The second network tries to learn a suitable strategy based on the reward estimation.

For the coupling with RL, an already existing fast rolling model is used, which was developed and validated at IBF by Lohmar et al [11]. It consists of several modules allowing the prediction of deformation, temperature and austenite grain size evolution as well as rolling forces and torques.

Table 2: Ranges of the RL agent's selectable process parameters.

Process parameter	**Limits**	
	Minimal	Maximum
Height reduction Δh	Δh_{min}	$\Delta h_{max}(\mu = 0.3, r = 205\ mm)$
Initial temperature	1000 [°C]	1200 [°C]
Inter-pass time	5 [s]	30 [s]
Rolling velocity	100 [mm/s]	500 [mm/s]

The presented training was carried out with 20,000 iterations, which corresponds to 20,000 calculated pass schedules. This equals about 24 hours of calculation time (CPU: Intel Xeon E3-1270). Fig. 6 shows the height evolution in black and the grain size evolution in red for the first and the final designed pass schedules. It is evident that the number of passes (from 18 to 8 passes) and the process time have been reduced and that the target grain size has been reached. Furthermore, both pass schedules reach the desired target height, regardless of the training progress. This is ensured by the dynamic adjustment described in Fig. 4 and Fig. 5.

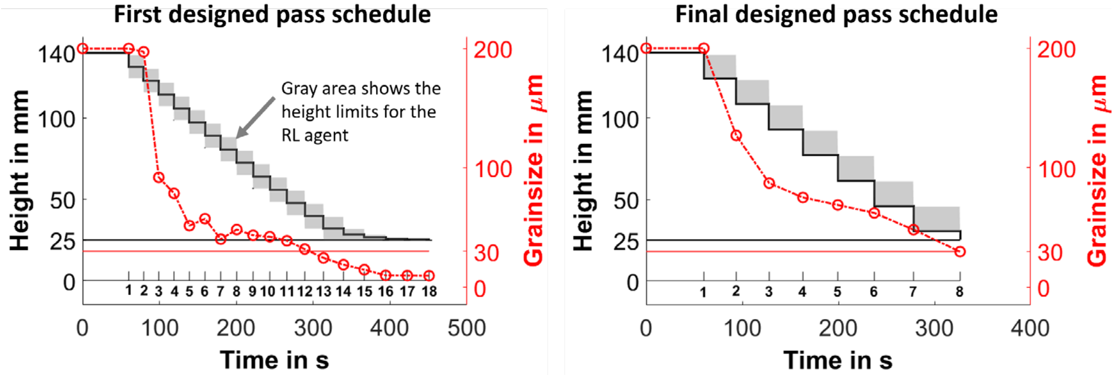

Fig. 6. The first (left) and the final (right) designed pass schedule after 20,000 iterations by the RL agent.

For comparison, Table 3 shows two pass schedules, one designed using static limits for Δh and one designed by the presented extended approach. In both cases, the same optimization objectives and boundary conditions were used. Both lead to similar results, e.g. final grain size.

Table 3. Comparison of the final designed pass schedule with static Δh limits and with the new dynamic approach.

Pass Number	**Height after each pass in [mm]**	
	Static Δh	Dynamic Δh
1	124.32	124.32
2	108.64	108.64
3	92.64	92.95
4	77.27	77.27
5	61.59	61.59
6	46.22	45.91
7	33.31	30.54
8	24.88	25.00

Although the difference between the two pass schedules appears to be negligible, it is of great importance since an undershooting of the final height is irreversible and leads to unsaleable products. The comparison demonstrates that the presented approach successfully supports the RL agent in identifying strategies, which lead to saleable products.

In addition to the resulting pass schedules, taking a closer look at the training progress is of interest, too. Therefore, the results achieved using the dynamic Δh are shown in more detail. Fig. **7** shows the evolution of the total reward R_{total} of the complete pass schedule in blue and the deviation ΔGS between the final average grain size and the target one in red during training. The evolution of the total reward over the iterations clearly shows that the reward increases the most within the first 1,000 iterations. Afterwards, it increases only slightly until it reaches a constant level at about 2,500 iterations.

A similar pattern can also be found for the deviation of the final grain size from the target one (ΔGS). The initial deviation of about 20 µm is significantly reduced in the first 1,000 iterations, so that the deviation at 2,500 iterations lies at only a few µm. Subsequently, the desired grain size is reached. The results show that the training is clearly converging towards a total reward of about -10, at which the target grain size is reached. Comparing the evolutions of the total reward and the deviation from the target grain size, it is noticeable that both correlate well, especially at the beginning and up to about 1,000 iterations. This strongly indicates that at the beginning the RL agent is trying to achieve the target grain size as soon as possible. This results from the prioritization of this component in the reward function. This indicates that weighting of individual components in the overarching reward function can be used to successfully focus on certain optimization objectives while still achieving all other desired objectives.

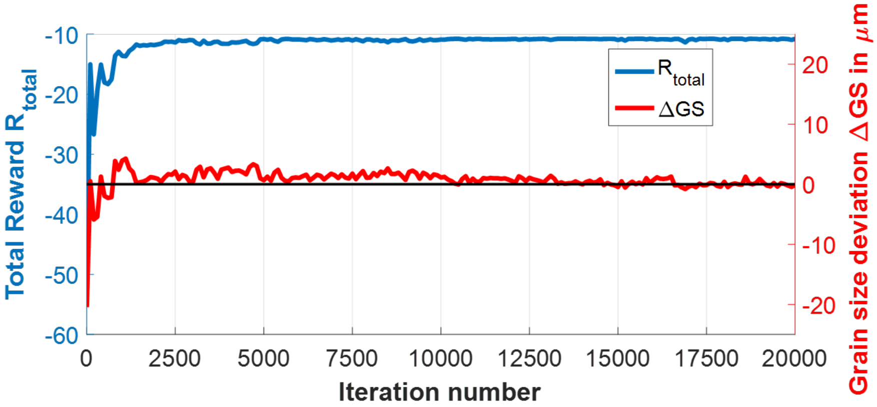

Fig. 7. Evolution of the total reward in blue and the deviation of the average grain size from the target grain size (30 µm) ΔGS in red.

Fig. 8 shows the evolution of the energy consumption as well as the total number of passes of the designed pass schedules during the training. It is noticeable that the energy consumption initially increases significantly, peaks at around 500 iterations at 0.14 GJ and then continuously decreases until it reaches a constant level at about 5,000 iterations with a value of 0.12 GJ.

Another example for the well-defined prioritization in the reward function is the evolution of the number of passes in the pass schedules laid out. The aim was to favor height reductions that are as large as possible so that the required number of passes is as low as possible. A small number of passes usually results in a shorter process time and thus higher efficiency. Here, the number of passes per pass schedule strives very quickly from 18 to the minimum possible number of eight.

In addition to the values shown here, the analysis of the resulting rolling forces and torques showed that the maximum rolling mill limitations were retained at all times during the training. Moreover, the final height is exactly reached by using the dynamic adjustment of the height reduction presented in this work, it can be stated that rollable pass schedules have been laid out successfully after only a few thousand iterations. Moreover, the dynamic adjustment of the height reduction accelerated the training progress significantly compared to the previous work [1].

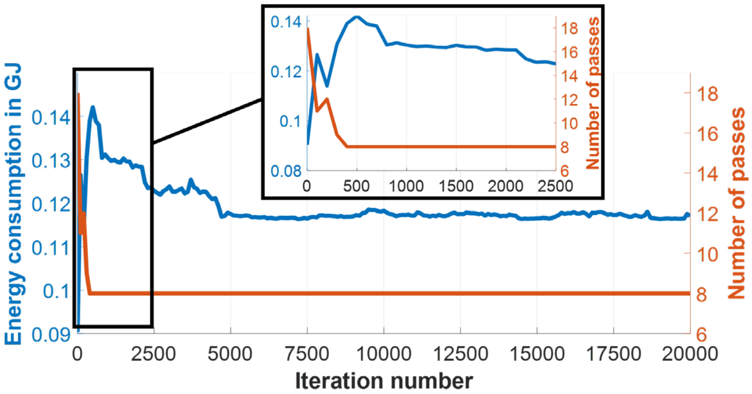

Fig. 8: Evolution of the energy consumption in blue and of the total number of passes in orange of the designed pass schedules during the training.

Summary

The results show that the coupling of RL with an FRM can successfully learn the complex interactions between process parameters, material behavior and the final properties such as the final austenite grain size. The approach does not require explicit knowledge or procedure and can automatically design pass schedules for multiple optimization objectives based on a reward function. The presented extension, the dynamic adjustment of the height reduction, ensures that the target height is always hit exactly. It guarantees automatically designed pass schedule that lead to saleable products. Furthermore, compared to the results of previous work, learning could rather be accelerated. Additionally, the further objectives, especially the target grain size, were successfully achieved. Following next, the designed pass schedules are to be experimentally validated, while the knowledge of the trained RL agents will be transferred to design further pass schedules with varying starting and final geometries.

Acknowledgement

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany´s Excellence Strategy – EXC-2023 Internet of Production – 390621612.

References

[1] C. Scheiderer, T. Thun, C. Idzik, A.F. Posada-Moreno, A. Krämer, J. Lohmar, G. Hirt, T. Meisen, Simulation-as-a-Service for Reinforcement Learning Applications by Example of Heavy Plate Rolling Processes, Procedia Manuf. 51 (2020) 897-903. https://doi.org/10.1016/j.promfg.2020.10.126

[2] J.M. Allwood, J.M. Cullen, M.A. Carruth, Sustainable materials. With both eyes open ; [future buildings, vehicles, products and equipment - made efficiently and made with less new material]. UIT Cambridge, Cambridge, 2012

[3] World Steel Association, 2022 World Steel in Figures. World crude steel production 1950 to 2021. https://worldsteel.org/steel-topics/statistics/world-steel-in-figures-2022/

[4] D.S. Svietlichnyj, M. Pietrzyk, On-Line Model for Control of Hot Plate Rolling. In: Beynon JH (Hrsg) 3rd International Conference on Modelling of Metal Rolling Processes. IOM Communications, London, S (1999) 62-71

[5] M. Schmidtchen, R. Kawalla, Fast Numerical Simulation of Symmetric Flat Rolling Processes for Inhomogeneous Materials Using a Layer Model – Part I. Basic Theory, Steel Res. Int. 87 (2016) 1065-1081. https://doi.org/10.1002/srin.201600047

[6] A. Özgür, Y. Uygun, M.-T. Hütt, A review of planning and scheduling methods for hot rolling mills in steel production, Comput. Ind. Eng. 151 (2021) 106606. https://doi.org/10.1016/j.cie.2020.106606

[7] V. Pandey, P.S. Rao, S. Singh, M. Pandey, A Calculation Procedure and Optimization for Pass Scheduling in Rolling Process. A Review, J. Mater. Sci. Mech. Eng. 5 (2018) 126-130

[8] S. Wu, X. Zhou, J. Ren, G. Cao, Z. Liu, N. Shi, Optimal design of hot rolling process for C-Mn steel by combining industrial data-driven model and multi-objective optimization algorithm. J. Iron Steel Res. Int. 25 (2018) 700–705. https://doi.org/10.1007/s42243-018-0101-8

[9] C.A. Hernández Carreón, J.E. Mancilla Tolama, G. Castilla Valdez, I. Hernández González, Multi-Objective Optimization of the Hot Rolling Scheduling of Steel Using a Genetic Algorithm. MRS Adv. 4 (2019) 3373-3380. https://doi.org/10.1557/adv.2019.436

[10] J.H. Beynon, C.M. Sellars, Modelling Microstructure and Its Effects during Multipass Hot Rolling. Iron Steel Inst. Jap. 32 (1992) 359–367. https://doi.org/10.2355/isijinternational.32.359

[11] J. Lohmar, S. Seuren, M. Bambach, G. Hirt, Design and Application of an Advanced Fast Rolling Model with Through Thickness Resolution for Heavy Plate Rolling. In: J. Guzzoni, M. Manning (Hrsg) 2nd International Conference on Ingot Casting Rolling Forging, ICRF 2014

[12] S. Shen, D. Guye, X. Ma, S. Yue, N. Armanfard, Multistep networks for roll force prediction in hot strip rolling mill, Machine Learning with Applications 7 (2022) 100245. https://doi.org/10.1016/j.mlwa.2021.100245

[13] R.S. Sutton, A. Barto, Reinforcement learning. An introduction. Adaptive computation and machine learning, The MIT Press, Cambridge, MA, London, 2018

[14] C. Li, P. Zheng, Y. Yin, B. Wang, L .Wang, Deep Reinforcement Learning in Smart Manufacturing: A Review and Prospects, CIRP J. Manuf. Sci. Technol. 40 (2022) 75-101. https://doi.org/10.1016/j.cirpj.2022.11.003

[15] A. Esteso, D. Peidro, J. Mula, M. Díaz-Madroñero, Reinforcement learning applied to production planning and control, Int. J. Prod. Res. (2022) 1-18. https://doi.org/10.1080/00207543.2022.2104180

[16] M. Panzer, B. Bender, Deep reinforcement learning in production systems: a systematic literature review, Int. J. Prod. Res. 60 (2022) 4316-4341. https://doi.org/10.1080/00207543.2021.1973138

[17] N. Reinisch, F. Rudolph, S. Günther, D. Bailly, G. Hirt, Successful Pass Schedule Design in Open-Die Forging Using Double Deep Q-Learning, Processes 9 (2021) 1084. https://doi.org/10.3390/pr9071084

[18] O. Gamal, M.I.P. Mohamed, C.G. Patel, H. Roth, Data-Driven Model-Free Intelligent Roll Gap Control of Bar and Wire Hot Rolling Process Using Reinforcement Learning, IJMERR 10 (2021) 349–356. https://doi.org/10.18178/ijmerr.10.7.349-356

[19] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, M. Riedmiller, Deterministic Policy Gradient Algorithms, Proceedings of the 31 st International Conference on Machine Learning, 32. Aufl, Beijing, China, 2014, 387-395