Improving a Deep Learning Temperature-Forecasting Model of a 3-Axis Precision Machine with Domain Randomized Thermal Simulation Data

E. Boos[1(✉)], X. Thiem[1], H. Wiemer[1], and S. Ihlenfeldt[1,2]

[1] Technische Universität Dresden, 01062 Dresden, Germany
eugen.boos@tu-dresden.de
[2] Fraunhofer Institute for Machine Tools and Forming Technology, Nöthnitzer Strasse 44, 01187 Dresden, Germany

Abstract. With the continuous rise of industry 4.0 applications, artificial intelligence and data driven monitoring systems for machine tools proved themselves as highly capable alternatives to classical analytical approaches. However, their precision is limited to a number of crucial aspects. One of the main aspects revolves around the lack of meaningful data, which leads to imprecise and false model predictions. This issue is closely linked to production processes and machine tools in production engineering, as the available amount of meaningful real data is strongly limited. The usage of simulation models to acquire additional synthetic data is able to fill this lack. This work looks into improving the prediction accuracy of a deep learning model for temperature forecasting of a 3-axis precision machine by combing and comparing real process data with domain randomized simulation data. The used thermal simulation model is based on finite element models of the machine assemblies. Model order reduction techniques were applied to the FE models to reduce the computational effort, increasing the simulation-to-reality gap. The approach is evaluated on unseen real data, demonstrating the underlying potential of the inclusion of synthetic data from simulation models of machine behavior.

Keywords: Condition monitoring · Synthetic data · Domain randomization · Deep learning forecasting · Digital twins

1 Introduction

In machine tool engineering as well as robotics, the correct positioning of the tool center point (TCP) is at most importance for ensuring a functioning workflow, as well as meeting component quality standards. This is especially the case for high precision tasks such as milling and grinding. However, despite regular calibrations of the TCP, temperature induced deformations during different operation tasks lead to offsets of the TCP [1]. In dependence of the magnitude, these offsets lead to rejections of high precision components. One possible way to counteract is the usage of digital twins such

M. Liewald et al. (Eds.): WGP 2022, LNPE, pp. 574–584, 2023.
https://doi.org/10.1007/978-3-031-18318-8_58

as simulation models, which run in parallel to the machine tool and predict temperature induced displacements based on real load cases [2]. With this information, the TCP can be adjusted in real time. As advantageous as this approach is, it still has shortcomings. In particular, the computational time necessary, for a high precision finite element (FE) thermo-mechanical simulation of a machine tool, limiting the complexity of the model to meet the real time computation requirements. As well as the limitation to solely one prediction step, restricting additional features, which profit from a multi-horizon forecasting.

These shortcomings can be eliminated with the usage of Deep Learning (DL) forecasting models [2]. Two of the currently best performing DL forecasting models are DeepAR [3] and the Temporal Fusion Transformer (TFT) [4]. Both models are based on Recurrent Neural Networks (RNN) and specially designed for probabilistic time series multi-horizon forecasting. However, as powerful as these models can be, they are limited to two crucial aspects: the historic data itself, and the available amount of data [5]. As the prediction is limited to past events, previously unknown process behaviors and errors will not be detected. Whereas, not enough data of the overall process and possible error cases leads to imprecise and false predictions. These issues are closely linked to production processes and machine tools in production engineering. Machine tools are mostly individual or custom constructions—even if structurally identical, they often differ in features, equipment level and purpose. Therefore, their production behaviours are not fully comparable and data acquisition must happen at the individual machine [6]. In total, the availability and accessibility of enough high-quality data of production processes and machine tools for Artificial Intelligence (AI) based applications is not given. The collaboration of a physical simulation based digital twin to synthetically generate data for the AI model can fill the lack of accessible real data [7]. Unfortunately, discrepancies between simulations and the real world make the transfer challenging. This reality gap [8] postulates that even the most complex simulation is not able to represent reality in all its details, due to inaccurate parametrization, model simplifications and unmodeled physical effects, for instance such as wear-and-tear and local environmental influences. The synthetic data source does not represent reality well enough to be used in a direct equivalence with real world data [9]. Bridging this gap is crucial in the expansion of available machine tool data sets and filling the existing lack of data. In robotics and robotic learning, Domain Randomization (DR) techniques are used to enable a comparability of graphical simulated image data with real data to further expand the learning data sets [8]. The goal of these approaches is to adjust the data distribution of simulated data sets to either match the distribution of real data [10] or let the real data distribution be a subset of the synthetic data distribution [11].

This study looks into the transfer of DR techniques to improve the temperature-forecasting accuracy of a TFT model with the inclusion of synthetic data from a thermal simulation model of a 3-axis precision machine. It serves as groundwork and a proof of concept for transferring this study's findings to the TCP offset correction.

2 Related Work

Mapping real world interactions and behaviors within simulated environments is a well-known problem solving strategy. Especially the FE method offers a tool for engineers

to recreate complex mechanical and thermal relationships to further improve insight. However, when it comes to transferring behaviors from simulated environments into the real world, results are taken with caution due to discrepancies between the two. Model simplifications, inaccurate parametrization and unmodeled physical effects make transferring behaviors from simulations challenging [8]. These differences are known as the reality gap. One possible approach to make the simulation closely match the physical reality are further system identifications and high-detailed simulations. However, AI-based approaches are in need of high amounts of data. High quality simulations might be helpful, but do not fully solve the issue at hand. Nevertheless, including data from high-quality graphical simulation with other approaches can reduce the number of real labeled samples [12].

When it comes to low level simulations, adding noise to the system parameters [13] can improve the realism of the simulated results. However, it also adds another layer of difficult to evaluate uncertainty, as well as might lead to unfeasible results, which both are unacceptable in production engineering. Approximate Bayesian Computation (ABC) [14] has been one of the main methods to leverage simulation uncertainties. Hereby, parameter settings are accepted or rejected if they are within a certain specified range determined by real observations. The set of accepted parameters approximates the posterior for the real parameters. However, ABC methods have a high computational effort, making them almost unusable in combination with complex FE simulations.

Simulation randomization and transfer learning techniques such as Domain Adaption (DA) have been a powerful tool in robotics and Reinforcement Learning (RL) [15]. With the inclusion of probabilistic simulation parameters, the simulation data distribution is adjusted in a way to fit the real data distribution. Classical optimization routines [16], Bayesian optimization methods [15], as well as the usage of Generative Adversarial Networks [17] have shown promising results. Another form of simulation randomization techniques is the Domain Randomization (DR) (see Sect. 3.2). DR describes the concept of highly randomizing simulation parameters in order to cover the real distribution of the real-world data in spite of the bias between the simulation model and the real world [8]. The goal is, by providing enough simulated variability, to adjust the data distribution in simulated data sets so that the distribution of real data is a subset of it. This follows the assumption, that if an AI-model at training time is capable of working with the randomized simulation data, it is able to generalize the learned data features onto the real data set at testing time. Bayesian Domain Randomization via probabilistic inference showed promising results in robotics simulation environments [18].

Most works in this field focus on computer vision tasks and robotics. For time-series data, transfer learning techniques such as DA and DR have been used for learning temporal latent relationships in health data [19], to perform speech recognition [20] and anomaly detection [21]. However, their effective implementation into AI-applied production engineering problems has be reserved yet.

3 Machine Set Up, Methods and Data Basis

3.1 Temperature Prediction of Demonstrator Machine

The machine in Fig. 1 (left) is selected as the demonstrator for this work. It is a three axes Cartesian serial kinematic in lightweight construction. It consists primarily of aluminum alloy plates which are clamped with the help of tie rods [22]. This machine has the special ability to compensate errors in six degrees of freedom. Three ball screw axes drive the z-slide. Flexure bearings connect the ball screw nuts and the guide carriages to the z-slide. Therefore, the z-slide can be tilted around X- and Y-direction in small angles up to 10 mrad. Two parallel linear direct drives actuate the y-slide. Flexure bearings connect the y-slide to the guide carriages of the linear guideways at the z-slide. In this way, the y-slide can be tilted around the z-direction up to 1.6 mrad. The rotatory axes are named as virtual since they are no rotatory axes in the usual sense [23]. The corresponding simulation model is physically based (see Fig. 1(b)). The temperature field of the machine is calculated using a reduced FE model. The finite element model of the machine is divided in submodels. These submodels do not include relative movement between machine components (e.g. between x-slide and y-slide). The systems of the submodels are reduced by model order reduction methods (Krylov subspace [24]) to obtain a fast computational model. Technological load data, e.g. axis positions, velocities, motor current from the machine control are used as input data for the model. These data are used to calculate the heat sources, due to friction and power loss in various machine components. In addition, the thermal conduction at the interfaces between the submodels and to the environment is calculated. The environmental temperature is measured with a Pt100 resistance temperature sensor. Empirical models are used to calculate the heat sources and thermal conductions.

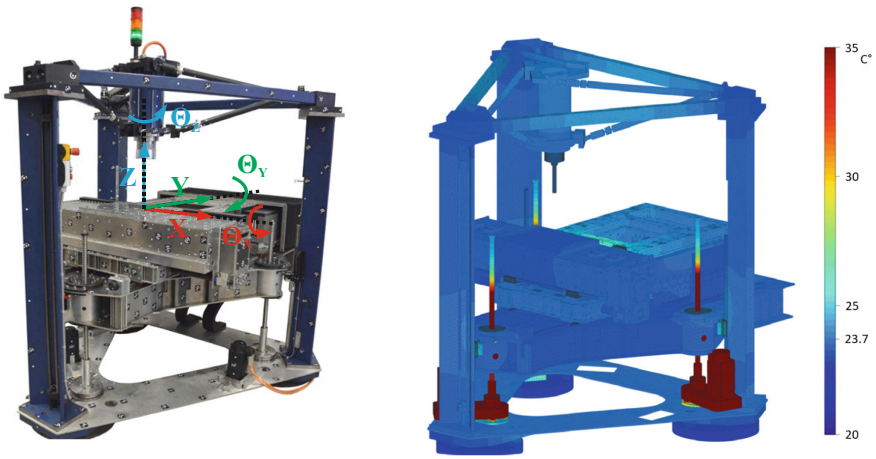

Fig. 1. Machine tool set up (left), and corresponding simulation model (right).

3.2 Applied Domain Randomization

In robotic vision tasks such as object localization, object detection, pose estimation, and semantic segmentation, DR is mainly applied to *domain features.* These are features which describe the domain in which the robot has to perform in, rather than key features of its main task. Lighting exposure, number of light sources, textures, color, noise and field of view of the camera are examples for possible domain features from vision tasks [8]. Therefore, visual DR provides enough simulated domain variability of visual parameters to better adjust to real world key parameters. Transferring this concept onto time series data and physically simulation models, such as FE models, is however, challenging as domain-based features do not exist in the same sense. Each parameter of a physical model is linked to a static, dynamic, linear or nonlinear physical relation to the real world and hence randomizing those means randomizing key parameters, which are mapped onto a time series. The simulation model used in this work has a total of roughly 240 adjustable key parameters, such as material, engine and profile linear guide parameters, as well as emission coefficients, scaling factors for convection and radiation to environment and many more. For this proof of concept, out of the total adjustable parameters, 12 different scaling factors were chosen and randomized uniformly by $\pm$ 20% of their initial value. The scaling factors for the power loss approaches of the machine components bearings, profile rail guides, ball screw spindles to nuts, synchronous motors and linear direct drives were selected. They are often recommendations by manufacturers, which offer simplified model adjustments. Hence, their level of uncertainty is equivalently high and therefore suitable for randomizing.

3.3 Real and Synthetic Data Basis

The data categories used in this work can be split into two spate sources—real and synthetic data sources. The real data was collected from experimental setups, where the machine was analyzed during test runs with different load cases. All in all experimental data of 5 days was available, each day with testing times of roughly 10 h. The machine data—axis positions, velocities and motor current of all 5 engines, was collected with a sampling rate of 100 Hz. The target value is the temperature development within the machine. To completely map the temperature development of the machine, it is equipped with overall 48 Pt100 temperature sensors with a sampling rate of 0.1 Hz. Due to redundancies in the temperature curve development, of the 48 build in sensors the 2 with the most unique temperature development where chosen as the target prediction value of the TFT model—one (X1) in the center of the x-slide, and the other one (Z1) in the center of the z-slide. Sensor X1 is placed rather closely to the engine of the x-slide and highly correlates with its usage. Whereas, the substantial size of the z-slide, in comparison to the x and y-slide, buffers the heat development of the Z1 sensor. The synthetic data was collected from the simulation model (see Sect. 3.1). Hereby, real historic machine data was used as input for the simulation model. Due to the limited amount of actual real historic machine data, its usage as input values was applied multiple times, however, with different randomized scaling factors to generate deviating temperature predictions (see Sect. 3.2). As a consequence of the different sampling rates between machine and temperature data, a direct mapping between input and target data was not feasible. The

used AI-based forecasting model (see Sect. 3.4) requires input and target to be the same sampling rate. Therefore, for the usage with the AI-model the machine data was cut to a sampling rate of 1 Hz, and the temperature data was stretched to 1 Hz. This turned the temperature curve into a floor and ceiling function-like curve. Additionally, the simulation based temperature curve is more continuous due to the higher resolution in comparison to the Pt100 sensors. Further obstructing the uniformity between machine and temperature data, as well as real and synthetic data. To tackle this uniformity the real and simulated temperature curves were transformed into a B-spline, improving the continuous curvature of the temperature, and hence the comparability of the sampling rates between input and target data.

3.4 Model Architecture and Learning Strategy

The core DL model used in this work is specialized on forecasting. The inputs are time series sequences (machine data) and the output of the model is the next possible time step of the target time series (temperature). The deviation between the prediction and the actual event allows evaluating current and possible future states of the machine. DeepAR [3] and the Temporal Fusion Transformer (TFT) [4] are currently the best performing probabilistic forecasting models in regards to tabular or time series data. Within this work the TFT model was used, as it outperformance DeepAR in regards to multi-horizon time series forecasting [4]. The TFT uses recurrent layers for local processing to learn temporal relationships, and interpretable self-attention layers for long-term dependencies.

Table 1. Teaching strategies and data source distribution

	TFT (real data)	Amount [samples]	TFT (simulation data)	Amount [samples]
Training	Real data	67.802	Simulation data	13.536.968*
Validation	Real data	34.501	Real data	34.501
Testing	Real data	34.501	Real data	34.501

The TFT models used in this work were designed to evaluate the last 720 time steps (in seconds) of machine data as input to predict the next 180 time steps of the temperature development of two selected temperature sensors (see Sect. 3.3). The loss function used in this work is the quantile loss, with quantile values for q_2, q_{10}, q_{25}, q_{50}, q_{75}, q_{90} and q_{98}. Based on these input and prediction time windows a hyperparameter optimization was done with the usage of solely real data. Additionally, an early stoppage was implemented to finish the learning process automatically, if the validation accuracy does not change over a set number of epochs. This set up was used to train and compare two separate TFT models—one trained, validated and tested only on real data, and the one trained on simulation data, but also validated and tested on the same real data samples (see Table 1). Hence, a comparison between the two TFT models is possible, where one was trained on

limited amounts of real data, in contrast to one which was trained on significantly more simulation data. For teaching the real data TFT overall 67.802 samples of real data were used. In contrast, for the simulation data TFT roughly 13.5 million samples of simulated data existed, where for reach training epoch 128.000 samples were randomly selected. The real data samples for validating and testing both models were identical to further support a fair comparison.

4 Experimental Results

Additionally to the forecasting of the absolute temperature values, separately trained TFT models for forecasting the temperature derivative were introduced. The reasoning behind this addition is due to the simplicity of the temperature curve, which can be approximated with the 1st order delay elements. To further demonstrate the potential of the proposed approach, as well adjust the complexity of the forecast to the application on the TCP offset correction, the temperature derivate curve was included. The curve itself is the derivative of the temperature B-Spline curve (see Sect. 3.3).

4.1 Absolute Temperature Forecasting Performance

Tables 2 and 3 show the performance results of the two different TFT models respectively for each of the two different sensors. Both models, trained with either real or simulation data, reach high performance results for the MAE, MSE as well as the R_2 values of the corresponding next time step forecast, as well as for the 180th time step prediction (see Sect. 3.4). Although the simulation data TFT does score better, the performance difference to the real data TFT is small. The same observation applies to the evaluation of the quantile values shown in Table 2.

Table 2. Evaluation metrics of the temperature forecast

	MAE	MSE	R2–1	R2–180
TFT RD—Z1	2.54e−3	1.70e−5	1.000	1.000
TFT SD—Z1	8.01e−4	1.49e−6	1.000	1.000
TFT RD—X1	1.97e−2	7.34e−4	0.998	0.996
TFT SD—X1	3.06e−3	2.94e−5	1.000	0.998

4.2 Temperature Derivative Forecasting Performance

The forecast of the temperature derivative demonstrates the difference in performance between the two TFT models. Figure 2 shows the prediction curve of the respectively next time step for sensor Z1 and Fig. 3 the results for sensor X1. For both sensors the real data TFT (left) performs in significant deviations from the actual temperature derivative

Table 3. Pinball loss results of the included quantiles of the temperature forecast

	q_2	q10	q25	q75	q90	q98
TFT RD—Z1	1.64e−4	5.60e−4	1.13e−3	1.68e−3	8.46e−4	2.48e−4
TFT SD—Z1	5.47e−5	3.36e−4	4.56e−4	4.27e−4	2.28e−4	6.30e−5
TFT RD—X1	2.53e−3	5.87e−3	8.02e−3	4.63e−3	2.28e−3	5.60e−4
TFT SD—X1	6.13e−5	1.47e−4	1.97e−4	2.09e−4	1.20e−4	4.29e−5

curve. Whereas, the prediction curve of the simulation data TFT (right) clearly overlaps with the actual curve. Additionally the level of uncertainty, displayed by the quantile values, is strictly smaller with the simulated data TFT. This is especially visible between the results of sensor X1. Figure 4 shows the prediction of the next 180 time steps (180 s) from the time point t_0 onwards. This comparison shows another level of performance difference between the two models. Whereas, the simulation data TFT (right) is able to follow the actual curve, as well as predict the next time steps with little uncertainty, the real data TFT (left) is not able to provide a reliable prediction. The evaluation metrics shown in Tables 4 and 5 provide a similar comparison outlook. Especially the R_2 values show a significant performance difference between the real data TFT and the synthetic data TFT.

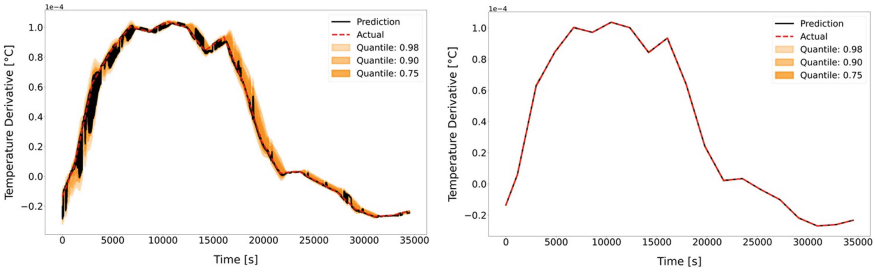

Fig. 2. Complete derivative prediction results for sensor Y1 of the real data TFT (left), and simulation data TFT (right).

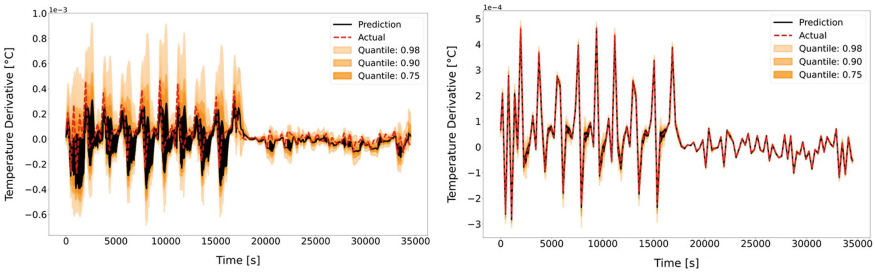

Fig. 3. Complete temperature derivative prediction results for sensor X1 of the real data TFT (left), and simulation data TFT (right).

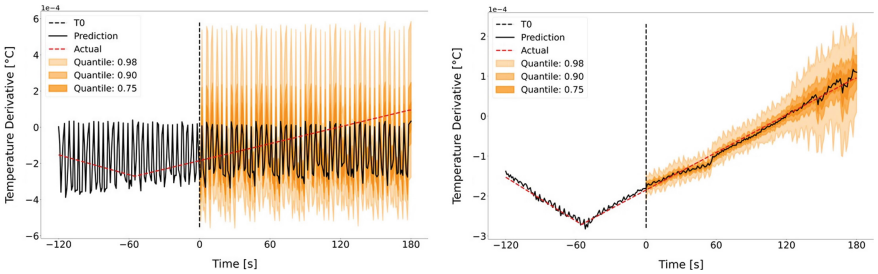

Fig. 4. Temperature derivative prediction of the next 180 time steps for sensor X1 of the real data TFT (left), and simulation data TFT (right).

Table 4. Evaluation metrics of the temperature derivative forecast

	MAE	MSE	R2–1	R2–180
TFT RD—Z1	1.49e−2	7.78e−4	0.997	0.993
TFT SD—Z1	8.32e−4	6.57e−6	1.000	1.000
TFT RD—X1	8.67e−1	1.56e−0	−0.08	−0.581
TFT SD—X1	5.12e−2	1.61e−2	0.999	0.124

Table 5. Pinball Loss results of the included quantiles of the temperature derivative forecast

	q_2	q10	q25	q75	q90	q98
TFT RD—Z1	7.99e−4	3.13e−3	4.17e−3	6.15e−3	3.02e−3	7.85e−4
TFT SD—Z1	4.14e−5	1.01e−4	9.99e−5	1.34e−4	9.46e−5	3.46e−5
TFT RD—X1	3.48e−2	1.45e−1	2.08e−1	3.80e−1	3.22e−1	1.74e−1
TFT SD—X1	6.58e−4	1.56e−3	1.57e−3	2.46e−3	1.78e−3	6.94e−4

5 Conclusion and Outlook

All in all this work demonstrated the capability of domain randomized simulation data from thermal FE-models for DL forecasting tasks. Although a basic machine temperature forecast is possible with solely real data, the forecast of more complex machine behaviors is limited. Besides offering a more accurate deterministic forecast, the inclusion of simulation data also decreases the level of uncertainty. Even though the forecast within this work is limited to the machine temperature development, it shows the opportunity it offers for the usage as a feasible TCP offset correction method. It demonstrates a certain superiority in comparison to the simulation model itself. The DL-forecasting model works faster and is able to predict multiple time steps ahead. These two advantages allow a

more reliable forecast and therefore a potentially improved adjustment of the TCP offset. Additionally, it eliminates the necessity of a real time simulation model, as simulation models are further simplified to meet this requirement. As the computational effort of accumulating machine behavior from simulations is out-sourced from the operational time to the development phase, more complex simulations can be used and hence further improve DL prediction accuracy.

The next step will be the adoption to the TCP correction, as well as a direct performance comparison between simulation based and DL-based models. Future direction to further improve the accuracy of the DL-models include the addition of optimization routines, instead of set uniformly distributed simulation parameters, to further adjust the distribution of the simulated data to the real data distribution.

Domain randomization is a promising research direction for production engineering toward bridging the reality gap between machine behavior and simulation results.

References

1. Großmann, K., et al.: Thermo-Energetische Gestaltung von Werkzeugmaschinen: Eine Übersicht zu Zielen und Vorgehen im SFB/Transregio 96, Zeitschrift für wirtschaftlichen Fabrikbetrieb (2012)
2. Mayr, J., et al.: Thermal issues in machine tools. In: CIRP (2012)
3. Salinas, D., et al.: DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks (2019)
4. Lim, B., et al.: Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting (2020)
5. Kiangala, K.S., Wang, Z.: Initiating predictive maintenance for a conveyor motor in a bottling plant using industry 4.0 concepts. Int. J. Adv. Manuf. Technol. **97**(9–12), 3251–3271 (2018). https://doi.org/10.1007/s00170-018-2093-8
6. Reuß, M., et al.: Ermittlung der Auswirkung des statistischen Verhaltens baugleicher Werkzeugmaschi- nen, Internationales Forum Mechatronik (2011)
7. von Rueden, L., et al.: Combining machine learning and simulation to a hybrid modelling approach: current and future directions. In: Advances in Intelligent Data Analysis XVIII, Lecture Notes in Computer Science (2020)
8. Tobin J., et al.: Domain randomization for transferring deep neural networks from simulation to the real world. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2017)
9. Koos, S., et al.: Crossing the reality gap in evolutionary robotics by promoting transferable controllers. In: GECCO'10 (2010)
10. Lee, S., et al.: StRDAN: synthetic-to-real domain adaptation network for vehicle re-identification. In: IEEE Conference on Computer Vision and Pattern Recognition (2020)
11. Zhao, W., et al.: Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In: IEEE Symposium Series on Computational Intelligence (2020)
12. Richter, S.R., et al.: Playing for data: ground truth from computer games. In: European Conference on Computer Vision. Springer (2016)
13. Tan, J., et al.: Sim-to-real: learning agile locomotion for quadruped robots (2018)
14. Beaumont, M., et al.: Approximate Bayesian computation in population genetics. Genetics (2002)
15. Wilson, G., et al.: A survey of unsupervised deep domain adaptation. ACM Trans. Intell. Syst. Technol. (2020)

16. Chebotar, Y., et al: Closing the Sim-to-Real Loop: Adapting Simulation Randomization with Real World Experience (2019)
17. Rao, K., et al.: RL-CycleGAN: reinforcement learning aware simulation-to-real. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
18. Ramos, F., et al.: BayesSim: adaptive domain randomization via probabilistic inference for robotics simulators (2019)
19. Purushotham, S., et al.: Variational adversarial deep domain adaptation for health care time series analysis. In: International Conference on Learning Representations (2017)
20. Hosseini-Asl, E., et al.: Augmented cyclic adversarial learning for low resource domain adaptation. In: International Conference on Learning Representations (2019)
21. Vercruyssen, V., et al.: Transfer learning for time series anomaly detection. In: Proceedings of the CEUR Workshop (2017)
22. Ihlenfeldt, S., et al.: Innovatives mechatronisches Systemkonzept für eine hochdynamische Werkzeug- maschine. In: Bertram, T., Corves, B., Janschek, K. (eds.) Mechatronik 2017
23. Ihlenfeldt, S., et al.: Simplified Manufacturing of Machine Tools Utilising Mechatronic Solutions on the Example of the Experimental Machine MAX. Springer International Publishing (2020)
24. Freund, R.W.: Model reduction methods based on Krylov subspaces. Acta Numerica (2003)