Proceedings are available on SSRN: www.ssrn.com

MIC Procedia (2020) 160–165

20th Machining Innovations Conference for Aerospace Industry 2020 (MIC 2020), December 2nd 2020, Hannover, Germany

Using AutoML to Optimize Shape Error Prediction in Milling Processes

B. Denkena[a], M.-A. Dittrich[a], M. Lindauer[b], J. Mainka[a], L. Stürenburg[a,*]

[a] *Institute of Production Engineering and Machine Tools (IFW), Leibniz Universität Hannover, An der Universität 2, Garbsen, 30823, Germany*
[b] *Institut für Informationsverarbeitung (TNT), Leibniz Universität Hannover, Appelstr. 9A, Hannover, 30167, Germany*

* Corresponding author. Tel.: +49-511-762-18305 Fax: +49-511-762-5115-. *E-mail address:* stuerenburg@ifw.uni-hannover.de

Abstract

Manufacturing of tool molds represents a single part production characterized by varying designs and various different process steps. The associated milling processes require a precise and complex process planning, which subsequently has to be optimized by running-in tests and adaptions to meet the quality specifications. Moreover, high costs of the raw material and the milling tools require a particularly careful and therefore time-consuming choice of process parameters, mainly based on human experience. Often, subsequent rework becomes necessary. This results in additional efforts during the process. For that purpose, machine learning can be used to find correlations between the process parameters in the process planning and the resulting shape error prior to the first cut. Hereby, the choice of the machine learning algorithm and its hyperparameters largely defines the prediction quality. As a disadvantage, finding the optimum of these hyperparameters to model a process with machine learning can be a tedious, time-consuming and error-prone procedure that also highly relies on the experience of the respective user. Automated machine learning (AutoML) offers a method to automatically search for a well-performing set of hyperparameters for a specific machine learning application. This study shows the performance improvements achieved by AutoML to predict shape errors that can occur during milling. For this purpose, a series of experimental investigations was conducted to collect representative data in a varying pocket milling process of cold working steel 1.2842. The design of experiment is supposed to ensure a variety of process parameters. As a novel addition, the machine learning model is incorporating the time-variant behavior such as tool wear. Additionally, the study is making a more realistic approach as it is considering error influences from CAD until the machined part in contrast to other studies. We show that we can achieve substantial improvements in terms of prediction RMSE by using the AutoML tool auto-sklearn; depending on the data between a factor of five and three orders of magnitude compared to plain default settings. This study demonstrates the high potential of using automated machine learning regarding the reduction of efforts in process planning due to improved prediction of shape errors and the ease of using state-of-the-art machine learning.

Keywords: AutoML; Milling; Simulation; Machine Learning; Tool Molds

1. Introduction

Modern production industry changes towards a higher variety of product design combined with shorter development cycles. In addition, the lot sizes of the respective products have decreased over time. Examples for this development can be found especially in the aerospace industry and in die and mold manufacturing.

Due to the conditions described above, the planning and run-in procedure of the manufacturing process is even more relevant than before. For a smaller production lot size, the work preparation represents a higher percentage of the overall production costs in this scenario. Moreover, scrap-parts cannot be tolerated. Thus, improvement of production quality is becoming more and more important.

Generally, in machining, the planning with Computer-Aided Manufacturing methods (CAM-planning) is a work-intensive upstream step necessary to fulfill a sufficient process. The CAM-planning determines the tool path. Moreover, process parameters, like feed rate, material removal rate, chip thickness etc., are defined. These parameters are ordinarily responsible for quality parameters e.g. the resulting shape error [1]. In addition, the state of the tool wear is responsible for an increase in process forces and thus in shape errors. Typically, optimal parameters have to be found by workpiece-specific testing and run-in-procedures, which strongly depend on the expert knowledge of employees. This represents a major financial and time-critical challenge.

Another method to improve part quality is offered by machine learning (ML) approaches. Machine Learning uses data to create a surrogate model that is supposed to represent the underlying correlation of the respective process. Thus, correlations can be made between the process parameters and the resulting quality parameters. For this purpose, varieties of regression models are available from simple linear regression to sophisticated Support Vector Machines. Hyperparameter are responsible for the robustness and effectivity of the learning process of the algorithm.

In this context, simulation of engagement conditions of the tool can give enhanced insights of the process. Implementing the simulated engagement conditions and the state of tool wear, the resulting shape error can be predicted with ML. Using these methods, the efforts in process planning can be reduced. However, in order to use ML algorithms efficiently, it is crucial to choose a well-performing ML algorithm, the underlying ML model class and the corresponding hyperparameters. In this paper, we study the benefits of using automated ML [2], which supports users of ML by determining efficient ML pipelines, incl. algorithm, model class and hyperparameters.

2. State of the Art

Shape errors (i.e. difference between planned and produced shape of workpieces) in milling are unwanted multi-causal effects that consist of systematical and random influences. This results in a partially non-deterministic behavior in milling experiments. Repeated experiments do therefore not necessarily produce the same output. Schmitz et al. [3] conducted a study to determine the isolated causes for resulting geometrical shape errors. They showed that the main driver in milling for that setup are cutting forces. Other causes can be thermal expansion of the workpiece, tool and spindle as well as quasi-static geometric errors, caused by misalignment of the axis. Furthermore, the impact of vibrations in the process is an important factor. The composition and magnitude of the mentioned errors is specific for the respective process setup and engagement conditions. Moreover, errors in the controlling of machine axes can lead to deviations from the planned tool path and thus result in shape errors.

There are several strategies to avoid shape error during milling. One of these strategies is to implement a sensory driven control loop to adapt for unwanted effects during the process. This does usually not require any changes during the process planning phase, but can require extensive physical refurbishment of the machine. For example, Denkena et al. [4] used a dynamometer to collect information about the cutting forces during the machining procedures. These forces are then used to model a resulting shape error via an appropriate model. In-process, a tool path compensation was done according to the calculated shape error.

In contrast to compensation of potential shape errors during the process, a different strategy is offered by considering the erroneous behavior of the process during process planning. For this purpose, analytical, numerical, empirical and machine learning methods can model this behavior.

In analytical approaches, physical laws and correlations model the behavior of the process. This was done by Zeroudi and Fountaine [5]. They described the tool via a modified cylindrical cantilever beam. Subsequently, using the Euler-Bernoulli beam equation to calculate the displacement of the tool. This approach does not consider the remaining error influences.

Saffar et al [6] offer a numerical method. Here, the milling tool and the workpiece are modeled by finite elements. Both, the cutting force and the tool deflection are calculated in this simulation. Hereby the elastic deformation of the workpiece and machine were neglected.

An approach using machine learning was done by Dittrich et al. [7]. Within this study, a Support Vector Machine (SVM) was used to model the shape error in a 5-axis machining process. Gathered knowledge of resulting shape errors and respective engagement conditions were used to train this SVM. Subsequently, a compensation was conducted to minimize the resulting shape deviation of a workpiece. Due to the generation of workpiece independent knowledge this method can also be used for different workpiece geometries. It was shown that shape error compensations can be conducted for new parts using transferable process knowledge. In this experimental setup, the tool is changed after three samples in order to limit the effect of tool wear.

To find enhanced engagement conditions, that are usually not apparent within the CAM-Planning, but useful for ML-approaches, different approaches in simulation are used [8, 9, 10]. For that purpose, both workpiece and tool have to be digitally available.

Although the insight that the correct choice of an algorithm and its hyperparameter for a dataset at hand is quite old [11, 12], only fairly recently the field of automated machine learning (AutoML) found solutions to efficiently make these decisions. Well-known approaches include random search [13], genetic algorithms [14] or Bayesian Optimization [15]. Although, the early AutoML approach considered algorithm selection and hyperparameter optimization as independent problems, the joint optimization to obtain well-performing ML pipelines [16], incl. pre-processing, algorithm selection, hyperparameter tuning and post-processing, is state-of-the-art [17, 18, 19] for tabular, structured data.

This paper presents a novel approach to use AutoML and information about the tool wear for the optimization of milling processes. In contrast to existing methods, the approach uses more efficient ML methods to achieve higher prediction quality and a significantly smaller rooted-mean-square error (RMSE).

3. Approach

To investigate the effectiveness of AutoML regarding quality predictions in machining applications, experimental data of Dittrich et al. [20] is used to achieve comparable results as this study is also investigating the RMSE of ML algorithms.

In addition, to consider tool wear regarding the shape error in milling, both empirical data and simulation-based data will be gathered during a new experimental setup. The simulation-based data is preferred in this context as it is a non-invasive and scalable method. Using a dexel-based cutting simulation (i.e. IFW CutS) [21] in-depth knowledge of the cutting process can be gathered. Hereby, the simulation can calculate time-specific engagement conditions. These cutting conditions can be composed with the associated measured shape error to formulate a regression problem. Moreover, empirical data of the width of flank wear is collected between the processes to present the actual tool wear state. This was done to see if there are any benefits of using this empirical data versus using simulation based data. Thereby, time-variant behavior of the experiment is considered within this study.

We chose one of the state-of-the-art approaches for AutoML on tabular and structured data, namely auto-sklearn [22, 18] that combines Bayesian Optimization [23] with ensembling. Bayesian optimization iteratively fits a predictive surrogate model to all observed settings and corresponding RMSE values such that a trade-off between exploration and exploitation can be achieved via an acquisition function. Auto-sklearn is based on the Bayesian Optimization tool SMAC [24] that uses a random forest as predictive model and expected improvement as acquisition function. Compared to other optimization techniques, Bayesian Optimization is very sampling efficient for expensive black-box problems. In addition, ensembling is an important component of auto-sklearn because (i) ensembling of several machine learning models can help to decrease prediction errors [25] and (ii) auto-sklearn trains and evaluates several machine learning models as part of Bayesian Optimization. Figure 1 shows the general workflow of auto-sklearn without meta-learning. Auto-sklearn is interesting for this study here because it allows the highest level of abstraction for using machine learning; i.e., with only three lines of code, a well-performing machine learning pipeline can be trained. This trades off human expertise in knowing how to apply machine learning efficiently for compute power.

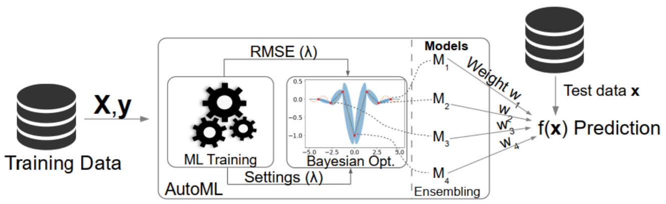

Figure 1: Workflow of AutoSklearn

4. Experimental setup

Previously being used data of Dittrich et al. [20] has been re-evaluated using AutoML. The aim of the re-evaluation is to show the potential of AutoML focusing the resulting RSME. In this context, the feature vector

$$X = \begin{pmatrix} v_f \\ Q_w \\ d_{th} \end{pmatrix} \tag{1}$$

was used. Hereby, X is composed by is the feed velocity v_f, the material removal rate Q_w and the projected distance between surface point to the TCP along tool axis d_{th}. The RMSE shows the average deviation between the predicted and the measured shape error.

In order to study the effects of different approaches using machine learning on this data, we compare: (i) default settings of an SVM with standardization as pre-preprocessing (“SVM”), (ii) a SVM with the settings of “Dittrich et al.”, (iii) optimizing the hyperparameters of the SVM with auto-sklearn in v.0.9.0 (“SVM (HPO)”), (iv) the setting of (iii) and adding the search for preprocessing on top of it (“SVM (HPO & prepro.)”), (iv) the search across all regression models and preprocessing techniques (“AutoML (w/o Ens.)”) and (v) the setting of (iv) and adding ensembling of the best models (“AutoML”). For the data of Dittrich et al. sample no. 9 and n = 50 was used. To run auto-sklearn, we used an inner cross-validation on the training data to estimate the final generalization perform to the test data. For the new data, we used an outer 5-fold cross validation. In all cases, we optimized for RMSE and limited the training and evaluation of a machine learning model to 30 seconds.

The full search space of auto-sklearn (iv + v) consisted of Ada-Boost, ARD Regression, Decision Tree, Extra Trees, Gaussian Process, Gradient Boosting, K-Nearest Neighbor, linear SVM, non-linear SVM, Random Forest, Ridge Regression and SGD from the scikit-learn package. Each of these are parameterized with their according hyperparameters. In addition, auto-sklearn can consider 17 pre-processing techniques, incl. PCA, truncated SVD or random tree embeddings, and its hyperparameters. Overall the full search space of auto-sklearn consisted of 140 hyperparameters. All machine learning experiments were performed on Intel Core i9-9900K CPUs with 3.60 GHz and 65 GB RAM, from which at most 3 GB were used for training and evaluating a machine learning model.

To gain further insights on the effects of tool wear, a new experimental setup was layed out. For this purpose, pocket milling processes (see Figure 2) were carried out at a constant cutting speed v_c = 120 m/min in down milling. The tests were carried out on a CNC machine type DMG HSC 30 using solid carbide milling tools type KENNAMETAL HARVI 1TE with a diameter D = 6.0 mm and a number of teeth z = 4. Cold working steel 1.2842 was used as workpiece material.

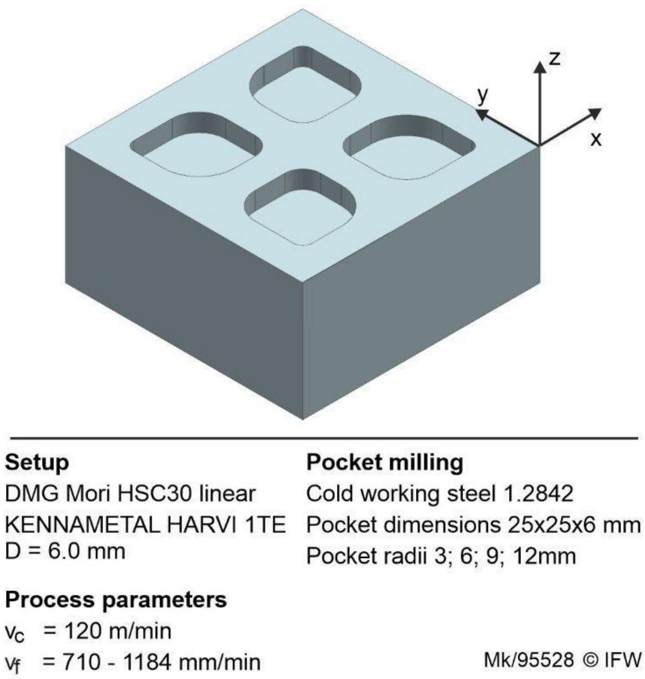

Figure 2: Experimental setup

A machine-integrated tactile probe type BLUM TC52 was used to determine the shape error at defined points. The width of flank wear land VB of the tool was measured optically with a digital microscope type VHX600 from Keyence. The resulting shape error as well as the wear condition of the tool were measured after four milled pockets and a corresponding cooling time of the workpiece. The machine-integrated tactile probe measures 3D points at 32 defined positions of the manufactured workpiece, which are located on the sidewalls of the pockets. These points are merged with the target geometry as defined by the CAD-model. The distance between the corresponding points presents the resulting shape error

$$d_s = \sqrt{(x_m - x_c)^2 + (y_m - y_c)^2} \quad (2)$$

where (x_c, y_c) are the points as designed in CAD and (x_m, y_m) are the measured points on the expected location, respectively.

To detect the influence of the feed velocity on the wear progress the feed speed v_f was varied between 710 and 1184 mm/min. The end of tool life was defined by a width of flank wear land VB ⩾ 75 µm. The material removal simulation IFW CutS is calculating process conditions as for example the material removal rate (Q_w), depth of cut (a_p), width of cut (a_e), feed velocity of the tool (v_f), acceleration of the tool (a_f) for each simulation iteration. For that purpose, actual machine axis-data is used to calculate the simulation via the software library ACCON-AGLink by Delta Logic [8].

To show how time-variant data is benefiting the prediction of shape errors, a splitting of feature sets was done. Feature set 1 consists of following feature vector

$$X = \begin{pmatrix} r \\ v_f \\ a_f \\ Q_w \\ a_e \end{pmatrix} \quad (3)$$

where r is the radius of the respective engagement point.

The feature vector of feature set 2 is complemented by information about the accumulated volume V_w that has been removed by the respective tool. This is done to see if data from simulation regarding the consideration of progressive tool wear and its influence can benefit the predictions. The volume V_w is defined as

$$V_w = \int_{t_0}^{t_s} Q_w \, dt \quad (4)$$

where t_s is the point in time where the respective measuring point was milled and t_0 is the point in time where the experiment started.

To find out what the benefit of using the measured flank wear mark width is, the feature vector has furthermore been supplemented by these 4 variables to create feature set 3.

5. Results

Effectivity of AutoML

As presented in Figure 3 the default Support Vector Machine (SVM) achieves the highest RMSE of all ML algorithms with an RMSE ≈ 100 mm. AutoML and past results from Dittrich et al. show already improved RMSE of predictions by the factor of 10^4. The SVM with Hyper Parameter Optimization (HPO) show both with and without preprocessing an RMSE ≈ 0.001 mm. AutoML without ensembling delivers an even better result of RMSE ≈ 10^{-5} mm and mainly used a surprisingly simple Ridge Regression or a Gaussian Process. AutoML without and with ensembles was able to achieve an even better RMSE value on the inner cross-validation. However, adding ensembles hurt the RMSE on this test data, since ensembling added a bit noise to the predictions which in turn increased the test RMSE.

Hereby, the resulting RMSE is also depending on the invested time for searching for the best combination of preprocessing, regression model and hyperparameters as presented in Figure 4. It can be seen that the RMSE is improving significantly over time and outperforms the previous state of the art of Dittrich et al. after roughly 80 seconds. After roughly 5 min, auto-sklearn achieved its final RMSE.

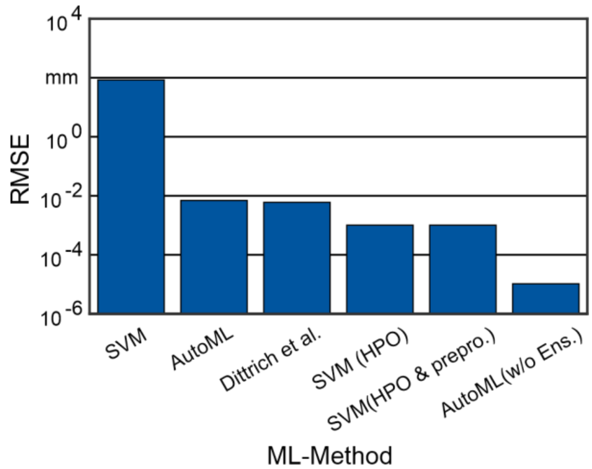

Figure 3: Comparison of RMSE for different ML-Methods

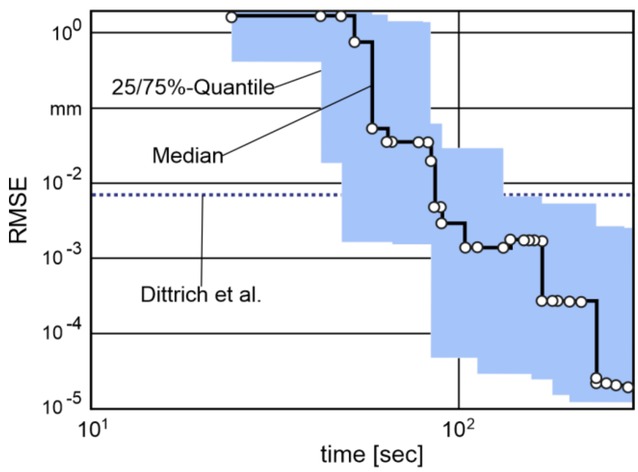

Figure 4: Development of RMSE over time in AutoML

Consideration of tool wear

Regarding the newly conducted experiments with effect of time-variant influences, the difference in the RMSE of predictions is presented in Figure 5. Hereby, it is noticeable that simulation-based data about the material removal volume is significantly improving prediction results. In this case, only the integrated material removal rate was added to formulate feature set 2. This shows that effects of tool wear can already be considered with benefits within process planning. Using additional information about the measured flank wear mark width is barely improving results in this study.

Also in this case, results of RMSE were highly correlated to the invested computation time as seen in Fig 6. Although slightly more expensive compared to the data from Dittrich et al., auto-sklearn was also able to outperform SVM default settings in less than 1 min and needed only few minutes to achieve state of the art.

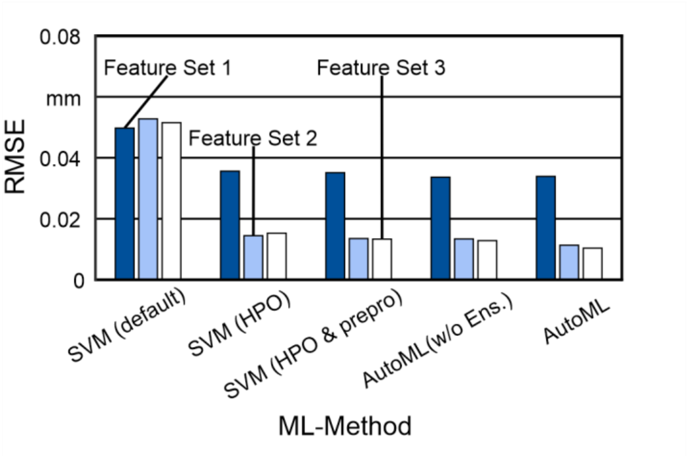

Figure 5: Comparison of RMSE for different ML-Methods

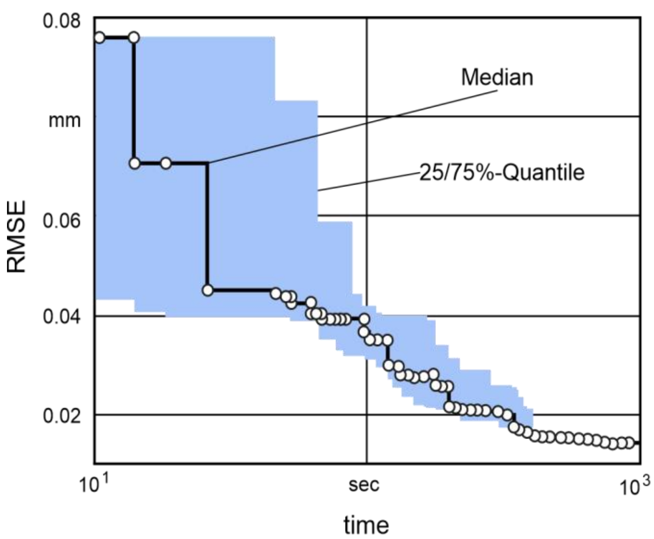

Figure 6: Development of RMSE over time over 10 random seeds for auto-sklearn

6. Conclusion

Previous studies regarding quality predictions in machining were mostly using SVM as a regression method. It was rarely shown how the hyperparameters were chosen or how they could be optimized regarding improved RMSE of predictions. Thus, being unclear how the model selection and hyperparameter optimization would effect applications in machining. Showing that AutoML can decrease prediction errors by a factor of 600 compared to previous studies, this method shows to have a great impact on machine learning applications in production. Since applying AutoML with tools such as auto-sklearn does require a lot less expertise in machine learning compared to manually making all decisions, we recommend to compare against AutoML as a baseline in all further applications of machine learning to predict errors in machining applications.

Regarding the general predictions of shape errors in industry-oriented applications, it was shown that the consideration of variables that can indicate the tool wear have shown improvements in the prediction of shape errors. Hereby, it was sufficient to use information from simulation alone to make robust predictions. For future studies, the extended variation of process parameters and materials will be of interest as well as the combination of transfer learning for other geometries and tool wear consideration regarding shape errors. Moreover, the implementation of predicted shape errors with regard to tool path compensation will be focus of future research.

Acknowledgements

Funded by the Lower Saxony Ministry of Science and Culture under grant number ZN3489 within the Lower Saxony “Vorab“ of the Volkswagen Foundation and supported by the Center for Digital Innovations (ZDIN). The transfer project T10 "Simulation-based Displacement Monitoring" from the Collaborative Research Center 653 is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – SFB 653 T10 – , for which the authors are grateful. We are thankful for the help with evaluating auto-sklearn by Matthias Feurer.

References

[1] Michael Fontain, Arnaud Devillez, Abdelhadi Moufki, Daniel Dudzinski, Modelling of cutting forces in ball-end milling with tool–surface inclination: Part II. Influence of cutting conditions, run-out, ploughing and inclination angle. Journal of Materials Processing Technology, 2007, 189/1-3:85-96

[2] Frank Hutter, Lars Kotthoff, Joaquin Vanschoren, Automated Machine Learning, Springer, 2019

[3] Tony L. Schmitz, John C. Ziegert, J. Suzanne Canning, Raul Zapata, Case study: A comparison of error sources in high-speed milling, Precision Engineering, 2008, 32/2:126-133

[4] Berend Denkena, Haythem Boujnah, Feeling machines for online detection and compensation of tool deflection in milling, CIRP Annals, 2018, 67/1:423-426

[4] Nasreddine Zeroudi, Michaël Fontaine, Prediction of tool deflection and tool path compensation in ball-end milling, Journal of Intelligent Manufacturing, 2015, 26/3:425-445

[5] Kenji Shimana, Eiji Kondo, Hiroko Karashima, Mitsuhiro Nakao, Shunichi Yamashita, An approach to real-time compensation of machining error using deflection of tool estimated from cutting forces in end-milling process, Journal of Advanced Mechanical Design, Systems, and Manufacturing, 2016, 10/2:22-32

[6] Reza Jalili Saffar, Mohammad Reza Razfar, O.Zarei, Elaheh Ghassemieh, Simulation of three-dimension cutting force and tool deflection in the end milling operation based on finite element method, Simulation Modelling Practice and Theory, 2008, 16/10:1677-1688

[7] Marc-André Dittrich, Florian Uhlich, Berend Denkena, Self-optimizing tool path generation for 5-axis machining processes, Journal of Manufacturing Science and Technology, 2018, 24:49-54

[8] Klaus Weinert, Shangjian Du, Patrick Damm, Marc Stautner, Swept volume generation for the simulation of machining processes, International Journal of Machine Tools and Manufacture, 2004, 44/6:617-628

[9] L. Taner Tunc, Erhan Budak, Extraction of 5-axis milling conditions from CAM data for process simulation, International Journal of Advanced Manufacturing Technology, 2009, 43:538-550

[10] Klaus Weinert, Tobias Surmann, Geometric Simulation of the milling process for free formed surfaces, Kolloquium - Forschergruppe FreiFormFlächen, 2003, http://www-isf.maschinenbau.uni-dortmund.de/content/documents/publications/articles/attachments/349.pdf

[11] John R. Rice, The Algorithm Selection Problem, Advances in Computers, 1976, 15:56-118

[12] Geoffrey F. Miller, Peter M. Todd, Shailesh U. Hegde, Designing Neural Networks using Genetic Algorithms. Proceedings of the third international conference on Genetic algorithms, 1989, 379-384

[13] James Begstra, Yoshua Bengio, Random search for hyper-parameter optimization, The Journal of Machine Learning Research, 2012, 13/1:307-361

[14] Randal S. Olson, Jason H. Moore, TPOT: A Tree-based Pipeline Optimization Tool for Automating Machine Learning, Proceedings of the Workshop on Automatic Machine Learning, 2016, 64:66-74

[15] Jasper Snoek, Hugo Larochelle, Ryan P. Adams, Practical Bayesian Optimization of Machine Learning Algorithms, Advances in Neural Information Processing Systems, 2012, 25

[16] Chris Thornton, Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms, Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, 847-855

[17] Lucas Zimmer, Marius Lindauer, Frank Hutter, Auto-PyTorch Tabular: Multi-Fidelity MetaLearning for Efficient and Robust AutoDL, 2020, arXiV:2006.13799

[18] Matthias Feurer, Katharina Eggensperger, Stefan Falkner, Marius Lindauer, Frank Hutter, Auto-Sklearn 2.0: The Next Generation, 2020, arXiv.2007.04074

[19] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, Alexander Smola, AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data, 2020, arXiv.2003.06505

[20] Florian Uhlich, Marc-André Dittrich, Berend Denkena, Data For: Self-optimizing Tool Path Generation for 5-axis Machining Processes, Mendeley Data, 2018, http://dx.doi.org/10.17632/smyg6cfwpk.1

[21] Volker Böß, Berend Denkena, Bernd Breidenstein, Marc-André Dittrich, Hai Nam Nguyen, Improving Technological Machining Mimulation by Tailored Workpiece Models and Kinematics, 17th CIRP Conference on Modeling of Machining Operations, 2019, 82:224-230.

[22] Matthias Feurer, Aaron Klein, Katharina Eggensperger, Jost Springenberg, Manuel Blum, Frank Hutter, Efficient and Robust Automated Machine Learning, Advances in Neural Information Processing Systems, 2015, 28

[23] Jonas Mockus, Bayesian Approach to Global Optimization, Springer, 1989

[24] Marius Lindauer, Katharina Eggensperger, Matthias Feurer, Stefan Falkner, André Biedenkapp, Frank Hutter, SMAC v3: Algorithm Configuration in Python, 2017, https://github.com/automl/SMAC3

[25] Rich Caruana, Alexandru Niculescu-Mizil, Geoff Crew, Alex Ksikes, Ensemble selection from libraries of models, Proceedings of the 21st international conference on Machine learning, 2004, pp 18