



Self-optimizing process planning of multi-step polishing processes

Berend Denkena^{1,2} · Marc-André Ditttrich^{1,2} · Hai Nam Nguyen¹ · Konrad Bild^{1,2}

Received: 11 December 2020 / Accepted: 25 February 2021 / Published online: 11 March 2021
© The Author(s) 2021

Abstract

Self-optimizing process planning is an essential approach for finding optimum process parameters and reducing ramp-up times in machining processes. For this purpose, polishing is presented as an application example. In conventional polishing processes, the process parameters are selected according to the operator's expertise in order to achieve a high-quality surface in the final production step. By implementing machine learning (ML) models in process planning, a correlation between process parameter and measured surface quality is generated. The application of this knowledge automates the selection of optimal process parameters in computer-aided manufacturing (CAM) and enables a continuous adaptation of the NC-code to changing process conditions. Applying the presented ML-model, the prediction accuracy of 83% will adapt the process parameters to achieve the target roughness of 0.2 μm . The sample efficiency is shown by the decrease in root mean square error from 0.1–0.28 to 0.02–0.07 μm with additional polishing iterations.

Keywords Abrasive polishing · Machine learning · Optimization · Simulation-based planning · Process planning

1 Introduction

In times of “Industry 4.0,” self-optimizing machining systems are in demand which enable a low-cost process with the output of individual high-quality products [1]. Additionally, the development of highly adaptable manufacturing processes is technically advanced to a degree that allows it to be applied in the industrial sector [2, 3]. In this context, process planning via the CAM interface provides an expedient way of achieving self-optimizing machining systems (SOMS) [2]. The extension of production processes through manufacturing simulation modifies the purely geometric and idealized view of conventional CAM software [4]. Machine learning models play an increasingly significant role in processing the understanding between the machining parameters in the NC-code and manufacturing outcome [5]. This eliminates the need for time-consuming experiments and purely experience-based process design. In particular, the

use of self-optimizing process planning has a high potential for cost reduction and quality maximization in the field of precision machining. However, approaches in this field have hardly been researched [6].

In this paper, an approach for self-optimizing process planning is presented in order to redefine the adaptive optimization of process parameters using the challenging example of polishing. With varying process conditions, the correlation between process parameters and the manufactured output changes. A planning approach via machine learning and simulation of the polishing process selects suitable process parameters for the NC-code generation. In the final step, the approach will be validated and the process parameters will be adapted to changing process conditions.

2 State of the art

In a variety of industrial sectors, polishing represents the final and most critical step in the production of high-quality surfaces and geometric accuracy [7]. Application examples for the finishing of high-end products include dental restorations [8, 9], joint prostheses [10, 11], mold inserts [12–14] or components for the implementation in optical or optoelectronic devices [15–17]. In order to meet the desired quality requirements of demanding

✉ Konrad Bild
bild@ifw.uni-hannover.de

¹ Institute of Production Engineering and Machine Tools, An der Universität 2, 30823 Garbsen, Germany

² Cluster of Excellence PhoenixD (Photonics, Optics, and Engineering – Innovation Across Disciplines), Hannover, Germany

applications, the use of appropriate process planning is crucial. The common challenges in the process planning of polishing are choosing the correct tool path, a suitable abrasive tool, and appropriate process parameters [6]. The decision-making is carried out via the expert knowledge of human process planners [18]. Their ability to detect the optimal solution is limited [2]. Due to the shifting interactions which determine the process result, mechanical polishing seems to be highly complex [19]. In summary, there is potential to optimize process planning within common polishing processes in order to improve quality and reduce costs.

Although self-optimizing process planning is gaining increasing interest at the moment, core elements like digital process planning as well as data feedback from the process and quality control are already described in the VDI 4499 Part 2 guideline [20]. A literary overview of different approaches toward the implementation of self-optimizing process planning in SOMS for manufacturing processes is presented in [2]. While many processes with a geometrical defined cutting edge have already been investigated with respect to self-optimization, polishing operations have not been in the limelight of research.

Strategies for the quality control of the manufactured polished output are determined directly on the workpiece by the shape deviation [21] or roughness [22, 23]. In addition, sensors can be used to measure acoustic emission [24, 25] or force [26] during polishing in order to obtain indirect information on the quality. Through supervised learning, the training data set for the polishing process is formed from input parameters such as cutting speed v_c , feed rate v_f , cutting depth f_t or grain size of the tool and from measured output variables such as roughness parameters or material removal rates [27]. Supervised machine learning is mostly applied in manufacturing processes to predict the best possible correlation between process parameters and manufacturing outcome [28]. The key advantage of supervised methods like Support Vector Regression is the additional minimization of the generalization error instead of the calculated deviation between the observed and predicted data. The generalization describes the achievement of a generalized solution, if the regression model is provided with other data than the training data [29]. The influence of unforeseen disturbances or unknown process variables, which appear variably in the process and reduce the model accuracy, is compensated by the feedback of the new data sets and the adjusted correlations between inputs and outputs [30]. This approach is a key-enabler of self-optimizing production system [31]. Based on the data a simulation model of the manufacturing process can be developed. Simulation models represent the real process conditions. This makes it possible to draw on optimum parameters or

tool paths when generating the NC-code in order to create the desired target workpiece.

3 Approach of a self-optimizing process planning

In order to automate the production and post-processing as well as to fulfill the high surface requirements at the same time, a planning approach is developed which ensures the appropriate parameterization of the process. It considers the individual processing case to achieve the desired roughness locally.

The overall approach for the polishing process in the machine tool is shown in Fig. 1. After measuring the initial roughness and assigning it locally to the workpiece, the planning algorithm and roughness model are used to generate a suitable NC-code for the polishing process. This NC-code contains the tool path and the locally selected process parameters to achieve the desired target roughness.

In order to continuously improve the prediction of the target variable on the basis of the selected process parameters, the planning algorithm is extended by a self-optimizing roughness model. By measuring the resulting roughness after polishing, a comparison of the predicted and measured roughness results can be made. The measured values are stored in a database after each processing so that the database grows continuously. This data is fed back into the roughness modelling and thus serves to improve the prediction. The knowledge of all previous iterations is used to plan the next iteration. By using a machine learning method (Support Vector Machine, SVM), the model is continuously adapted and improved.

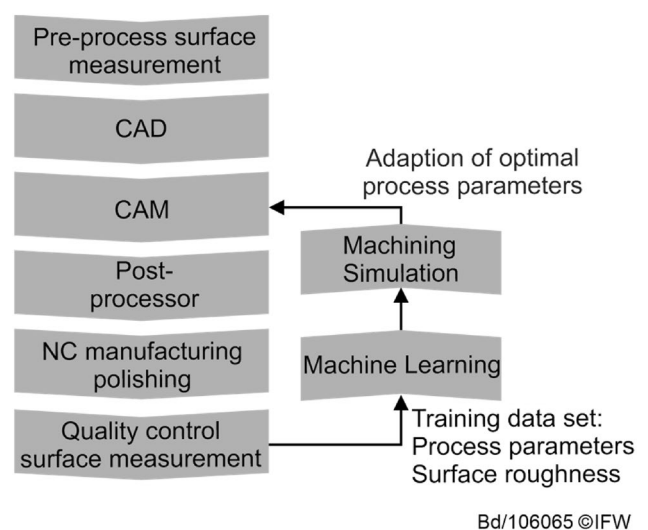


Fig. 1 Method for self-optimizing adaption of process parameters

3.1 Simulation of the polishing process

The implementation of the self-optimizing process planning is carried out by using algorithms for tool path generation and for deriving the locally suitable feed rate. This is based on an initially measured topography as well as the actual contact surface of the flexible polishing tool. The entire planning algorithm is implemented in the material removal simulation software IFW CutS which provides an open simulation platform for the technology-oriented simulation of manufacturing processes. IFW CutS offers basic functions of a dixel-based geometric process simulation on the basis of NC-based tool paths [32].

3.1.1 Contact width model of the polishing tool

To determine the contact area of the polishing tool at a defined inclination angle, only the contact width of the path of the polished area is considered. The contact width describes the width of the resulting polished path. This is due to the fact that for the roughness modelling the resulting roughness of the path is decisive and not the intermediate contact. Due to the flexibility of the polishing tool, the resulting width of the polished path cannot be taken directly from the determined cutting conditions of the simulation. To determine the contact width, the curvature radii are varied in experiments from $R = 50$ to 120 mm for both convex and concave workpieces. In the polishing tests, the depth of cut is varied in three steps ($f_t = 1; 1.5; 2$ mm). The influence of the inclination angle is not investigated in detail any further and is kept constant at $\alpha = 30^\circ$ which, based on preliminary tests, represents a good compromise between sufficient material removal and wear resistance. The feed and cutting speeds are also kept constant ($v_f = 300$ mm/min, $v_c = 15$ m/s), as these have no influence on the width of the polished path, which can also be explained by the geometric conditions.

The widths of the polished paths are found to be largely constant over the various radii of curvature, with occasional variations of 1–2 mm when polishing along the curvature. This results from the same contact conditions due to the constant angle between tool and workpiece. That is achieved by adjusting the inclination angle along the curvature during the polishing process. Across the curvature, however, the widths differ with changes in curvature. In this case, the different resulting widths of different curvatures from the experiment are mapped using a contact width model (Fig. 2). Thus all possible widths from $R = 50$ to 120 mm of different curvatures are enclosed by the ellipse.

The different circles represent the radii of curvature from $R = 50$ to 120 mm for the concave and convex workpieces respectively. Each contact width measured in the polishing

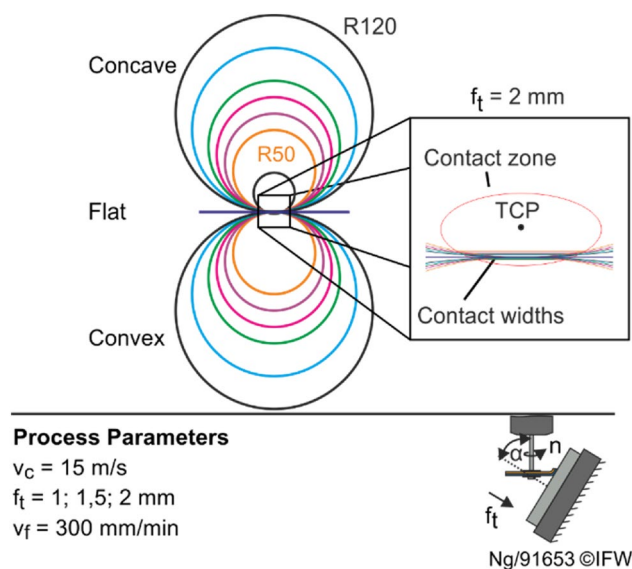


Fig. 2 Contact width model

experiments is plotted in this diagram as a horizontal line corresponding to its respective radius of curvature, so that both ends of the line touch the corresponding circle. An ellipse can be placed around these lines, resulting from different widths. The center of the ellipse is located at the tool center point (TCP). This is shown on the right side of Fig. 2 as an example of a depth of cut of $f_t = 2$ mm. Different forms of ellipses result from different depths of cut. The parameters for describing the ellipse are derived from this. For a depth of cut of $f_t = 2$ mm, for example, the ellipse parameters are $a = 12.2$ mm (radius in x-direction) and $b = 5.5$ mm (radius in y-direction). The contact described by these ellipse parameters forms the basis for describing the contact conditions in the process simulation.

The empirically determined contact width model is implemented in the material removal simulation. The workpiece is loaded into the simulation via a point cloud in comma-separated value (CSV)-format. Each point in the CSV-file contains information about position, orientation and local roughness value. This information is transferred to the dexels and serves as the basis for creating the dixel grid. So-called oriented dexels are used for the discretization. These are oriented perpendicular to the surface [33]. This property makes the representation of curved workpieces with oriented dexels particularly suitable.

In each time step, from the quantity of all dexels penetrated by the tool, the roughness is changed only by the dexels located within the contact zone. This zone is defined by the ellipse calculated from the contact width model. With this ellipse surface, the contact zone can be determined without having to explicitly determine the curvature of the workpiece across the tool, because the ellipse surface contains all

possible widths resulted of different curvatures (from $R=50$ to 120 mm).

3.1.2 Initial tool path

The initial tool path defines the movement of the tool and forms the basis for the adjustment of the process variables to achieve the desired roughness. It consists of points based on the addition of an offset value along the direction vector of the surface normal of each dixel. These points represent the positions of the tool center point (TCP). The offset value is calculated depending on the selected depth of cut and the inclination angle between tool and workpiece. The initial tool path is obtained by stringing together position values of the TCP from the tool. This results in polishing paths that run next to each other without overlap and cover the entire surface (full cut). In addition, the tool moves in the same direction (zig strategy). This is advantageous in terms of avoiding abrupt changes of direction.

In order to keep the contact width of the polishing tool as constant as possible, including for curved components, the approach is to keep the inclination angle between the tool and the workpiece constant. This is achieved by rotating the tool in the feed direction in line with the change in curvature of the workpiece. The information of the surface normals in the individual dexels is used for this. The process simulation and the resulting initial tool path are shown in Fig. 3.

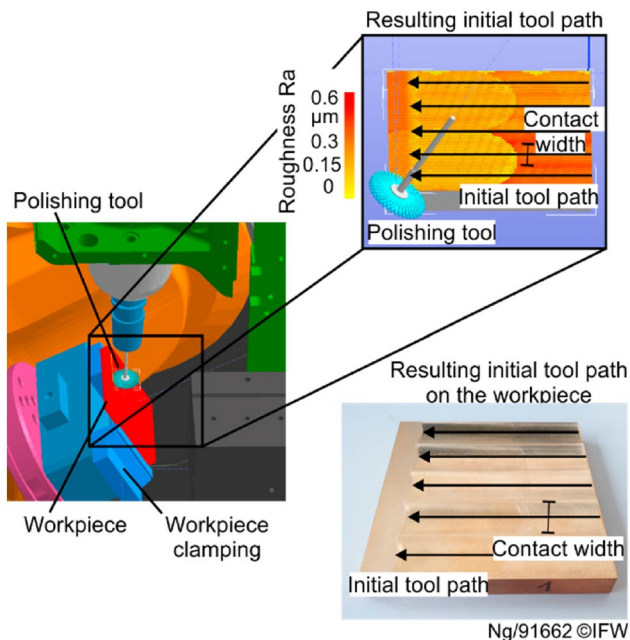


Fig. 3 Resulting initial tool path in the simulation and on the workpiece

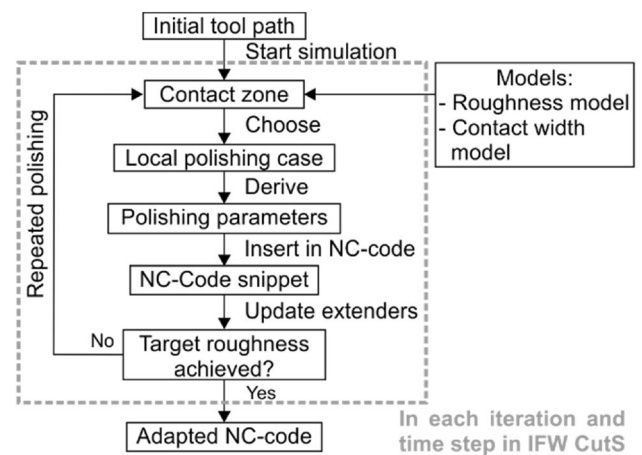


Fig. 4 Polishing planning algorithm

3.1.3 Representation and processing of the local workpiece information

In order to store the roughness from the measurement file locally as a surface attribute, an algorithm is developed which acquires roughness from the measurements quantitatively for a defined surface area and assigns the roughness values to individual points or dexels. This is achieved by implementing so-called dixel extenders. Each dixel is assigned its own extender. The roughness of individual workpiece areas is represented by coloring the dexels based on the roughness value stored in the extenders which is also shown in Fig. 3. In addition to the roughness, the extenders can also store any information for the respective dixel. Along with measured and simulated roughness values, so-called roughness lists (which contain the roughness values of different polishing iterations), locally selected process variables, and other variables for simulation and evaluation are also stored. The information about the locally selected feed rate is later included in the knowledge database from the extenders and used for the self-optimizing roughness modelling.

The extenders are used to simulate only the change in roughness. A shape deviation due to polishing is not considered, but can be taken into account by cutting the dexels. However, the polishing experiments show that there is hardly any change in the shape of the workpiece.

3.2 Adaptation of the polishing process to reflect individual processing requirements

The initial tool path is adapted in such a way that more intensive machining is carried out in areas of poor surface quality or where a specified target roughness has not yet been

reached. This is achieved by local reduction of the feed rate. Further roughness reduction is achieved by polishing multiple times at the same spot. The procedure for the adaptation of the NC-code is shown in Fig. 4. The tool moves according to the initial tool path in the material removal simulation. In each time step, the local situation is determined by reading and evaluating the extenders of the dexels penetrated by the tool and located within the contact zone determined by the contact width model. Based on the average initial roughness value of all penetrated dexels of the contact zone, the appropriate polishing case and the corresponding feed rate at this point are derived, allowing a maximum roughness reduction without falling below the selected target roughness. The roughness model used for this purpose is based on data from polishing experiments. Three different polishing cases are distinguished: no change in the feed rate necessary, change in the feed rate necessary, and no polishing.

The predicted roughness resulting from the selected feed rate is updated in the extenders of the affected dexels. The feed rate is updated in the initial NC-code at the respective position. As a result, by detecting the change of the polishing case, the NC-code segment is only added when the feed rate is changed. In places where the target roughness is already present, polishing is no longer performed and the polishing tool is moved upwards away from the workpiece.

In addition, a choice can be made between two different polishing strategies: with the undercutting strategy, polishing is performed in each case as soon as the measured roughness fails to meet the target roughness. The algorithm then selects the feed rate with which the maximal possible roughness reduction can be achieved without falling below the target roughness. If this is not possible, polishing is still performed at the feed rate which results in the smallest roughness reduction for this situation. Alternatively, with the approximate strategy, polishing will only be performed, if the roughness predicted from the selected feed rate is equal to or above the target roughness after polishing.

Areas are polished repeatedly, if the roughness has not yet reached the desired target value. This is done in the subsequent polishing iteration. The maximum number of polishing iterations is initially specified for repeated polishing. The planning of the process variables for a polishing iteration is based on the predicted roughness values of the previous iteration. For this purpose, the extenders contain so-called roughness lists, which store the roughness values for each polishing iteration. During interaction between tool and workpiece, the roughness values of the first polishing iteration are updated with the predicted roughness value. For the prediction of the following iteration, the values of the first iteration are used as initial roughness values. These are then stored in the second iteration of the roughness list. This is continued until the given number of maximum iterations is reached. The result

is an adapted NC-code for the individual machining case. The generated NC-code, based on CLDATA, is converted via the postprocessor and can thus be used on the machine.

4 Experimental investigations

For the experimental investigations, flat and simple curved workpieces are polished with varying parameters and an EVE type 1455 flexible polishing tool with the diameter $D = 26$ mm. The silicon carbide grains, which are bonded in polyurethane, are coarse enough to achieve the desired roughness range. Coarser grains, however, show increased wear and consequently higher roughness variations. In order to release the grains and ensure stable polishing conditions, all tools are used over a period of 5 min prior to the actual experiments. The polishing tools are changed after machining each workpiece, at the latest after 20 paths. The experiments are performed on a 5-axis machine tool (Sauer Ultrasonic 10) without cooling lubricant. The workpiece material is brass (Ms 58 according to DIN17660). The flat brass samples are ground beforehand in order to achieve consistent initial conditions ($R_a = 0.3\text{--}0.4$ μm). For this process, the parameter settings $v_c = 25$ m/s, $v_f = 40$ mm/min, $f_t = 0.01$ mm and axial feed rate $v_{fa} = 200$ mm/min are selected. The roughness measurements are carried out by tactile measurements on a Mahr Perthometer Concept Kontur with the settings shown in Table 1. Only the center of the polished path is measured and the measured values are assumed for the complete path.

The curved samples are produced by milling. The parameters ($v_f = 1400$ mm/min, $v_c = 240$ m/s, cutting depth $a_p = 0.66$ mm, cutting width $a_e = 2; 1.33; 1; 0.8$ mm) are chosen in such a way that a comparable initial roughness can be achieved. In order to exclude the influence of the initial roughness on the polished result, the difference between the polished surface roughness and the initial roughness ΔR_a of the material is selected as the target value. In the following, a negative ΔR_a indicates a reduction of the initial roughness.

Table 1 Measurement settings and technical specifications of the stylus profilometer

Parameter	Setting
Cut-off DIN EN ISO 4288	0.8 mm
Roughness parameter DIN EN ISO 4287	R_a in μm
Measuring direction	Polishing direction
Measuring position	Center of polished path
Vertical resolution	0.77 nm

4.1 Generating training data sets

In the experiments, the isolated influence of the feed rate on the roughness is examined in more detail because this has the greatest influence according to the screening experiments in [8]. The other parameters are kept constant. In the experiments, the feed rate is varied in ten steps from $v_f = 50$ mm/min to $v_f = 500$ mm/min. The inclination angle is kept constant at $\alpha = 30^\circ$ and the depth of cut at $f_t = 2$ mm. The cutting speed is also kept constant at $v_c = 15$ m/s. The data from the experiments is used to train the machine learning model. The results in Fig. 5 show that a higher roughness reduction can be achieved with a slow feed rate, which was to be expected. Thus, this parameter can be used for process planning to set desired target roughness values. In order to model the development of the surface after multiple machining steps, the samples are polished with varied repetition and feed rates. After each pass, the roughness is measured. During the experiments, polishing is repeated up to five times. The roughness reduction decreases with increasing repetition. In some cases, however, a roughness increase can also occur. The influence of different initial roughness values before polishing, generated by the different polishing repetitions, is taken into account for further modelling. The increased standard deviations may be due to uncertainty of roughness measurement or wear of the polishing tool. A systematic error can be excluded because the roughness difference is normally distributed. This is shown by the Shapiro–Wilk test, where the p-value is 0.25 and thus above the significance level of 0.05. Furthermore, five repeated measurements at the same surface location show a standard deviation of $0.003 \mu\text{m}$ regarding Ra. In this context, a reproducible measurement can be achieved by the stylus profilometer.

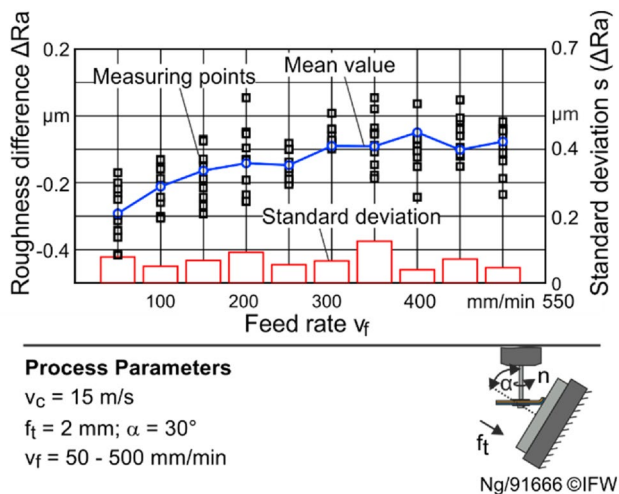


Fig. 5 Roughness changes with variation of feed rate

In order to be able to examine the influence of the curvatures of the components, two different radii of curvature ($R = 60$ mm and $R = 120$ mm) are selected for the convex and concave components respectively. For each unique geometry of the convex and concave parts, polishing tests are carried out by varying the feed rate, depth of cut, and number of repeated polishing operations. The feed rate is varied in five different step values ($v_f = 133.2; 200; 300; 400; 466.8$ mm/min). The depth of cut is also varied in five steps ($f_t = 0.67; 1; 1.5; 2; 2.33$ mm) and the number of polishing operations is varied in three steps ($N = 2; 3; 4$). All factor stage combinations are determined by using an orthogonal, rotatable, and centrally-composed experimental design. The polishing direction is transverse to the milling direction.

4.2 Extension of the planning algorithm with a self-optimizing roughness model

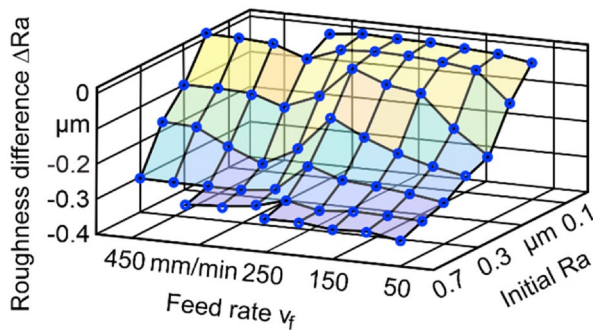
The planning algorithm is extended with a learning roughness model. In order to improve predictions of the roughness difference, the obtained polishing results are fed back to the roughness model. For this purpose, it is necessary to provide continuously updated data of the measurement after the process. For the planning and analysis of the iterative polishing process, measured values from a tactile roughness measuring instrument are used. As roughness cannot be measured over the entire surface, an interim solution is implemented by assigning the measured values over circular zones. The actual mean roughness value is measured around the center of each circular zone. Each circular zone represents one measured value so that all dexels within this circular zone have the same values. By selecting the radius and position of the circular zones, the entire surface of the workpiece is initialized with the measured values. For the plane workpiece, for example, eight circular zones are evenly distributed over the entire workpiece, as shown in the top right-hand corner of Fig. 3. The roughness measurements took place at the center of each circular zone.

The locally measured values before and after polishing, the locally selected process variables, as well as each predicted roughness are stored in the respective extenders. The resulting process information is exported from the extenders and stored in a knowledge database. The knowledge database then proceeds to grow with each polishing iteration and the generated roughness function is trained with larger data sets. Thus, a feedback of the polishing result into the roughness model is realized in order to allow a continuous, iterative update and adjustment of the prediction.

An SVM is used for modeling the acquired process data. In this study, MATLAB's "Regression Learner" is applied. The settings are listed in Table 2. Other available model types such as Linear Regression, Decision Tree, Ensemble Learning or Gaussian Process Regression were tested, but

Table 2 MATLAB settings of the SVM model

Parameter	Setting
App	Regression learner
Validation	Cross-validation, 50 folds
Model type	Cubic SVM
Kernel function	Cubic
Kernel scale	Automatic
Box constraint	Automatic
Epsilon	Automatic
Standardize data	True
R ²	83%



Process Parameters

$v_c = 15 \text{ m/s}$
 $f_t = 2 \text{ mm}; \alpha = 30^\circ$
 $v_f = 50 - 500 \text{ mm/min}$

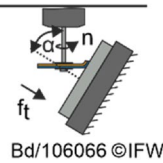


Fig. 6 3D-SVM regression roughness model

showed low prediction accuracy for this case. The model was trained with data sets from the initial experiments and, later, from the knowledge database. The cross-validation is applied because of its suitability for smaller data sets. Due to the fixed division of the measurement data set into 50 equally sized segments, one segment is used as validation data set and the remaining 49 as training data. After 50 runs, in which the segment assignment remains unchanged and the segment changes each time as validation data set, each segment is used once for validation [34]. This allows the data to

be split randomly, which leads to a more equal distribution of the data and avoids overfitting.

Figure 6 shows the derived roughness model. The roughness model achieves a coefficient of determination of 83%.

In order to transfer the ML-model from MATLAB to the simulation, an Excel file is generated as output from MATLAB, which contains the roughness function in discretized form. Different parameter combinations and the resulting changes in roughness are saved as a list for a range and step size to be set. For predicting the roughness change in process planning, the list is searched for the cases where the initial roughness of the list is closest to that of the simulation. The corresponding predicted value is used for the simulation and the appropriate feed rates are derived from this list. By means of an additional category in which the type of curvature is considered, the aspect of workpiece geometry is included in the database.

4.3 Validation of the self-optimizing planning algorithm

The algorithm’s ability to learn over the iterations and the self-optimizing planning of polishing processes were examined using three workpieces. The workpieces included planes (P1, P2 and P3), a single convex curved surface ($R = 60 \text{ mm}$, R60) and a double concave curved surface ($R = 90 \text{ mm}$ in both directions, R90). On the surface, different initial roughness values are produced in various areas by milling or grinding. This represents an individual machining case. The planning algorithm is tested to see how well a given target roughness can be achieved using the self-optimizing polishing process with increasing data sets. The applied initial roughness model was based on the experiments presented in Sect. 4.1. The surface is measured after each iteration, so that the next iteration is planned based on current roughness values. The actual roughness values are added to the knowledge base after each iteration and the roughness model is updated by re-training.

Table 3 gives an overview of the evaluation of all component geometries (P1, P2, P3, R60 and R90) with regard to the prediction quality of the roughness change ΔRa . The results are evaluated by means of the Root Mean Square Error (RMSE). The information from the extenders is used

Table 3 Prediction accuracy (RMSE) of self-optimizing polishing using various workpieces

Workpiece	Iteration					
	Initial	1st	2nd	3rd	4th	5th
P1	0.07	0.22	0.15	0.31	0.08	0.06
P2	0.07	0.10	0.05	0.04	Done	Done
P3	0.07	0.13	0.02	0.02	Done	Done
R60	0.05	0.28	0.00	0.04	Done	Done
R90	0.05	0.15	0.07	0.25	0.03	0.07

for the analysis by comparing the measured roughness values with the predicted roughness values. For P2 and P3, the waviness of the milling process is not taken into account in the roughness measurement and is filtered out accordingly. In addition, polishing is performed with the approximate polishing strategy. For P1, on the other hand, the waviness is taken into account, which resulted in more polishing iterations being necessary. This distinction is not necessary for the remaining components, as they do not show any waviness due to the grinding process. For P1 and the curved components, the undercutting polishing strategy is chosen. It can be generally seen that the deviation in the first iteration has increased for all parts compared to the starting data set in iteration 0. A possible explanation is the different range of the initial roughness between the experiments in Sect. 4.1 and the one generated here. Some of the workpieces exhibit initial roughness values that are also higher in some areas than those of the previous experiments. With increasing iterations, however, the RMSE value tends to decrease, which indicates a smaller deviation between the measured and predicted value. This shows the learning ability of the method with continuous adaption of the roughness model. The planning time is only significantly longer if the model is re-trained after each iteration. However, this can be done in larger intervals if the quality of prediction is sufficient.

Figure 7 depicts exemplary the development of the roughness over the course of three polishing iterations for the surfaces P2 and P3. It can be seen that four areas with different initial roughness values can be distinguished. The aim of the polishing process is in all cases to reduce the roughness to $Ra = 0.2 \mu\text{m}$. The results show that the surface quality improves with repeated machining. In this case, the target roughness is largely achieved after three iterations. In areas where the target is already achieved, repeated polishing no longer takes place (grey areas).

The validation shows that the approach works for different workpieces. In Fig. 7 it is demonstrated that P2 and P3 produce similar results after three iterations. Existing challenges are firstly the measurement method. Non-polished areas show a deviation of up to $0.06 \mu\text{m}$ after measuring again in the following iteration (Fig. 7). This is due to the fact that the previous measuring position cannot be hit exactly in the following measuring process. Moreover, the milling process produces a modified initial surface condition which differs from the training data set. Especially at higher initial roughness values around $0.7 \mu\text{m}$, the polished roughness of P2 and P3 shows higher deviations in comparison with lower initial roughness values in further iterations. Nevertheless, the model is capable of dealing with different initial situations and is able to derive appropriate process parameters to obtain the target roughness.

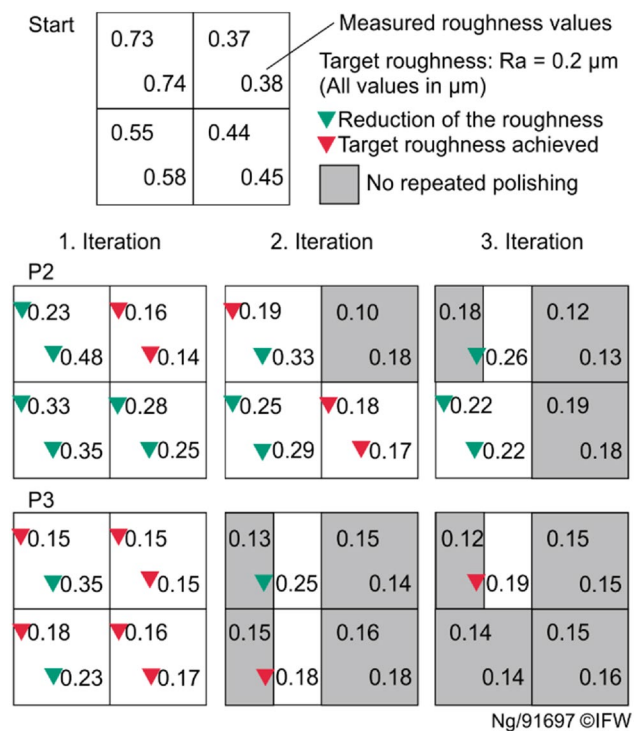


Fig. 7 Performance of the self-optimizing polishing

5 Conclusion and outlook

Process planning for finishing operation of complex workpiece geometries is usually done manually, which requires expert knowledge. In this study, an approach is presented for a self-optimizing planning of a polishing process, which aims to achieve a desired surface roughness independently of the initial roughness. For this purpose, a digital process chain and a learning roughness model based on SVM were developed. Although the polishing process itself is characterized by several influencing parameters, a prediction accuracy of 83% could be achieved. Based on the model an iterative procedure allows to adapt the polishing process in a self-optimizing manner. The approach was validated using plane and curved workpiece geometries.

In future work, it will be investigated to what extent the model accuracy can be improved by adding further input parameters such as tool wear and suitable surface measurement methods. Additionally, the transferability of the planning approach will be investigated by means of further set-ups such as new polishing tool and workpiece combinations. In this respect, the learning capability for unknown conditions will be validated.

Acknowledgements This research has been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy within the Cluster of Excellence PhoenixD (EXC 2122, Project ID 390833453).

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Brettel M, Fischer FG, Bendig D, Weber AR, Wolff B (2016) Enablers for self-optimizing production systems in the context of industrie 4.0. *Procedia* 41:93–98
- Möhring HC, Wiederkehr P, Erkorkmaz K, Kakinuma Y (2020) Self-optimizing machining systems. *CIRP Ann* 69(2):740–763
- Denkena B, Mörke T (2017) Cyber-physical and gentelligent systems in manufacturing and life cycle: genetics and intelligence—key to industry 4.0. Academic Press
- Sacharow A, Odendahl S, Peuker A, Biermann D (2013) Iterative, simulations-based shape modification by free-form deformation of the NC programs. *Adv Eng Softw* 56:63–71
- Stanula P, Ziegenbein A, Metternich J (2018) Machine learning algorithms in production: a guideline for efficient data source generation. *Procedia CIRP* 78:261–266
- Wang G, Zhou X, Liu J, Zhu P, Zhou H (2017) Polishing process planning based on fuzzy theory and case-based reasoning. *Int J Adv Manuf Technol* 90:907–915
- Klocke F, Dambon O, Behrens B (2011) Analysis of defect mechanisms in polishing of tool steels. *Prod Eng Res Dev* 5:475–483
- Denkena B, Dittrich M-A, Nguyen HN (2018) Technological CAD/CAM chain for automated polishing of geometrically complex workpieces. *Procedia CIRP* 78:313–317
- Bollen CM, Papaioanno W, Van Eldere J, Schepers E, Quirynen M, Van Steenberghe D (1996) The influence of abutment surface roughness on plaque accumulation and peri-implant mucositis. *Clin Oral Implant Res* 7(3):201–211
- Choopani Y, Khajehzadeh M, Razfar MR (2020) Using inverse replica fixture technique for improving nano-finishing of hip joint implant in abrasive flow finishing process. *Int J Adv Manuf Technol* 110:3035–3050
- Lu A, Gao Y, Jin T, Luo X, Zeng Q, Shang Z (2020) Effects of surface roughness and texture on the bacterial adhesion on the bearing of bio-ceramic joint implants: an in vitro study. *Ceram Int* 46(5):6550–6559
- Almeida R, Börret R, Rimkus W, Harrison DK, DeSilva AKM (2017) Material removal simulation for steel mould polishing. *Prod Manuf Res* 5(1):236–250
- Guo J, Liu K, Wang Z, Leong TG (2017) Magnetic field-assisted finishing of a mold insert with curved microstructures for injection molding of microfluidic chips. *Tribol Int* 114:306–314
- Biondani FG, Bissacco G, Mohanty S, Tang PT, Hansen Nørgaard H (2020) Multi-metal additive manufacturing process chain for optical quality mold generation. *J Mater Process Technol* 277:116451
- Zhao J, Huang J, Wang R, Peng HR, Hang W, Ji S (2020) Investigation of the optimal parameters for the surface finish of K9 optical glass using a soft abrasive rotatory flow polishing process. *J Manuf Process* 49:26–34
- Yamaguchi K, Ota M, Egashira K, Miwa H, Onchi Y, Tanada K (2016) Mirror-like surface finishing of PCD by fixed abrasive polishing. *Mater Sci Forum* 874:139–144
- Mosavat M, Rahimi A (2019) Numerical-experimental study on polishing of silicon wafer using magnetic abrasive finishing process. *Wear* 424–425:143–150
- Mohammad AEK, Hong J, Wang D (2017) Polishing of uneven surfaces using industrial robots based on neural network and genetic algorithm. *Int J Adv Manuf Technol* 93:1463–1471
- Segreto T, Teti R (2019) Machine learning for in-process end-point detection in robot-assisted polishing using multiple sensor monitoring. *Int J Adv Manuf Technol* 103:4173–4187
- Verein Deutscher Ingenieure (2011) Digital factory—digital factory operations part 2. Beuth Verlag GmbH, Berlin
- Michaeli W, Heßner S, Klaiber F, Forster J (2007) Geometrical accuracy and optical performance of injection moulded and injection-compression moulded plastic parts. *CIRP Ann* 56(1):545–548
- Kassubeck M, Malek T, Mühlhausen M, Kappel M, Castillo S, Dittrich M-A, Magnor M (2020) Optical quality control for adaptive polishing processes. In: 2020 IEEE southwest symposium on image analysis and interpretation, Albuquerque, USA, pp 90–94
- Segreto T, Karam S, Teti R (2017) Signal processing and pattern recognition for surface roughness assessment in multiple sensor monitoring of robot-assisted polishing. *Int J Adv Manuf Technol* 90:1023–1033
- Ahn JH, Lee MC, Jeong HD, Kim SR, Cho KK (2002) Intelligently automated polishing for high quality surface formation of sculptured die. *J Mater Process Technol* 130–131:339–344
- Segreto T, Karam S, Teti R, Ramsing J (2015) Feature extraction and pattern recognition in acoustic emission monitoring of robot assisted polishing. *Procedia CIRP* 28:22–27
- Nagata F, Hase T, Haga Z, Omoto M, Watanabe K (2007) CAD/CAM-based position/force controller for a mold polishing robot. *Mechatronics* 17(4–5):207–216
- Tong X, Wu X, Zhang F, Ma G (2019) Mechanism and parameter optimization in grinding and polishing of M300 steel by an elastic abrasive. *Materials* 12(3):340
- Wuest T, Weimer D, Irgens C, Thoben K-D (2016) Machine learning in manufacturing: advantages, challenges and applications. *Prod Manuf Res* 4(1):23–45
- Basak D, Pal S, Patranabis DC (2007) Support vector regression. *Neural Inf Process-Lett Rev* 11(10):203–224
- Thombansen U, Schüttler J, Auerbach T, Beckers M, Buchholz G, Eppelt U, Gloy Y-S, Fritz P, Kratz S, Lose J, Molitor T, Reißmann A, Schreiber A, Veselovac D, Willms K, Gries T, Michaeli W, Petring D, Poprawe R, Reisgen U, Schmitt R, Schulz W, Klocke F (2011) Model-based self-optimization for manufacturing systems. In: 2011 17th International Conference on Concurrent Enterprising, Aachen, Germany, pp 1–9
- Permin E, Bertelsmeier F, Blum M, Bützler J (2016) Self-optimizing production systems. *Procedia CIRP* 41:417–422
- Böß V, Denkena B, Breidenstein B, Dittrich M-A, Nguyen HN (2019) Improving technological machining simulation by tailored workpiece models and kinematics. *Procedia CIRP* 82:224–230
- Böß V, Denkena B, Henning S (2015) Investigation of the skive hobbing process by applying a dexel-based cutting simulation. *Procedia CIRP* 37:182–187
- Samsoniov V, Lipp J, Noodt P, Fenollar Solvay A, Meinsen T (2019) More machine learning for less: comparing data generation strategies in mechanical engineering and manufacturing. In: Conference: Symposium Series on Computational Intelligence (SSCI), Xiamen, China

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.