Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 118 (2023) 175–180

www.elsevier.com/locate/procedia

16th CIRP Conference on Intelligent Computation in Manufacturing Engineering, CIRP ICME '22, Italy

Auto-identification of dynamic axis models in machine tools

Alexander Puchta[a,*], Valentin Riegel[a], David Barton[a], Jürgen Fleischer[a]

[a]*wbk Institute of Production Science, Kaiserstrasse 12, 76131 Karlsruhe, Germany*

* Corresponding author. Tel.: +49 1523 950 2627; *E-mail address:* alexander.puchta@kit.edu

Abstract

In metal-cutting manufacturing, ever smaller lot sizes lead to frequent changes in machining processes. For this, monitoring solutions help with setup and process optimization to achieve high quality and productivity at lower costs. For example, cutting forces may be monitored indirectly based on available data, like motor currents. However, this requires exact models of the individual dynamic behavior of machine axes. The determination of such models is time-consuming and cost-intensive. This paper presents an approach for the automatic identification of dynamic axis models, thus enabling an efficient deployment of force monitoring to a wide range of existing machines.

Keywords: Digital manufacturing system; Identifcation; Machine tool

1. Introduction

In machining production, there are many existing machines in heterogeneous machine parks that do not have any Industrie 4.0 functionalities [1]. They therefore do not have digital twins and models that describe the dynamic behavior of their feed axes. Due to ever smaller batch sizes and thus frequent changes in production, the demands on the machines are increasing. With exact models of the feed axes, strategies can be derived to increase the Overall Equipment Effectiveness (OEE). For example, process monitoring solutions during setup or process optimization can achieve high quality and productivity at lower cost. However, the low costs can only be achieved if no additional implementation effort is required through additional sensor technology. Since the cost of additional sensor technology would increase further with different installation spaces and environmental conditions of heterogeneous machine parks, the objective should be to monitor the cutting force indirectly based on existing data such as motor currents and position signals.

However, indirect monitoring based on the mentioned signals requires a precise machine model, which is time-consuming and costly to create. Thus, as in many Industrie 4.0 implementations, single solutions would occur [1]. Although these represent high-performance solutions, they cannot be directly transferred to other systems and are therefore not scalable. A new implementation effort would be necessary for the transfer. For this reason, automation would increase the potential for minimizing the implementation effort. In addition, there is a new parameterization of the models for the same structure of the systems. Therefore, a scalable solution is needed, which can be rolled out to different machine types over a wide area. This requires component models that can be linked to each other by defined input and output variables, enabling a modular structure of the overall machine model of a feed axis.

A scalable solution for autonomous creation of feed axis models would be an enabler for optimization of OEE. In addition, this would enable the integration of brownfield systems in the concept of software-defined manufacturing [2].

10.1016/j.procir.2023.06.031

2. State of the Art and Objectives

2.1. Automated identification of parameters in control systems

For the implementation of an automated modeling of feed axes, control signals are required. Since these are not directly available in heterogeneous machine parks in the brownfield in most cases, they must also be identified automatically. For this purpose, there is an approach in which the signals are characterized by machine learning and domain-specific rules. [3]

This approach can serve as an upstream step for automated model identification to further reduce implementation effort.

2.2. Dynamic behavior of feed axes

For feed axes, there are a large number of research approaches to describe the dynamics and derive applications for them. For example, the effects of the dynamic behavior of machine tools on process stability have been studied. The goal is to establish general guidelines that can be used to improve the process stability of a machine tool and achieve increased metal removal rates. [4]

In addition, a methodology was developed to determine local damping influences of linear axes and ball screws in machine tools. The damping models derived from the methodology allow a better estimation of the dynamic behavior of the machine. [5]

In order to improve the dynamic behavior, actuator technology can be used in addition to control concepts, which, however, are time-consuming to integrate or lack robustness under different operating conditions. Thus, an improved acceleration behavior and an increasing accuracy, as well as the dynamic stiffness can be increased. [6]

For simulation, elaborate finite element (FE) models are used to study the machine stability of vertical milling machines. However, these are limited to one configuration. [7]

Further research is concerned with the dynamic behavior of machine tools, with the aim of improving the predictive capability of dynamic simulations. For this purpose, a methodology has been developed to build predictive simulation models. The identification and quantification of the uncertainties as well as the evaluation of the influencing variables on the dynamic behavior of machine tools serve as a basis for decision-making in order to improve the predictive capability of future FE models. [8]

In addition, applications are also being developed that use artificial neural networks (ANN) to represent these dynamics, such as wear detection for tools used in face milling. However, suitable sensor technology and data sets are required for this. [9]

Thus, many approaches exist that hinder the dissemination of these due to their complexity and effort in implementation. In addition, they often do not offer the possibility of transferability and usually only consider partial components.

2.3. Dynamic parameter identification in machine tools

There are studies to identify the dynamic parameters of a CNC machine tool feed system. Here, a standardized structure of a CNC machine feed system is assumed with servo drive, AC servo motor and mechanical system. To ensure high speed and high accuracy and stability due to increasing rotation and feed rates, relevant system parameters are identified to model the dynamics of the axis. A simplified control model and linear identification model of the feed system is used here. Input and output signals come from sensors embedded in the CNC machine tool. The unbiased least square algorithm is used here to determine the friction values. However, the simulation values determined in this way have values between ten and seventy percent deviation. However, the parameter identification used can be applied. Thus, the system is suitable to evaluate variations of the overall mechanical stiffness in different working environments/life phases of a WZM. [10]

Accurate modeling and identification of feed axis dynamics is also enormously important in the design of a high performance CNC machine. Here the parameters are estimated using unbiased least squares scheme, Kalman filter. For this, the axes are jogged in closed loop at different speeds. However, the problem of transferability applies here again. [11]

For the identification of system parameters, e.g. vibrations of the machine table, genetic algorithms (GA) are successfully used. However, additional sensor technology is required for this. [12]

2.4. Modular digital twin for machine tools

There are also approaches to build digital twins of machine tools modularly based on structured experimental parameter identifications. The focus here is on the overall model of the machine. Thus, when the configuration of the machine parameters changes, the submodels can be modified without recalibrating the overall model. [13]

However, there is no methodology here to adapt these to different types of feed axis.

In summary, there are thus currently no scalable approaches that meet the requirements of modeling with low expenditure of time and money for different axis types, which serve as the basis for various applications such as force monitoring. Therefore, an approach is needed which determines the configuration of the feed axis in order to be able to transfer these models.

3. Auto-identification of dynamic axis models

In order to meet these requirements, a concept was developed which focuses on the automated construction and parameterization of precise and individual feed axis models for a wide variety of feed axes. Three phases were defined for this purpose. In the first step, the components of the axis must be identified in order to build a precisely adapted model. Then, the structure of the axis is determined and the identified

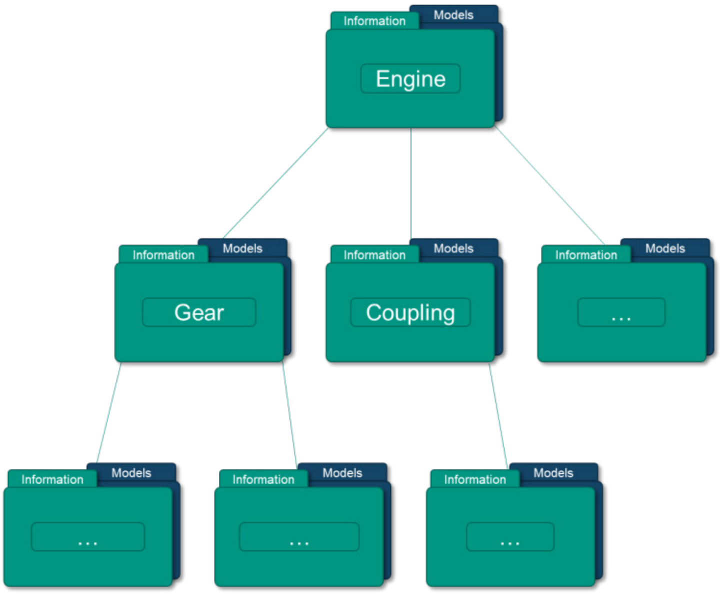

Fig. 1. Rule based decision tree.

components are built up in an overall model. The resulting parameter list is optimized using a genetic algorithm based on the initial reference run.

The basis of the methodology is a decision tree. This tree contains all possible components of feed axes. These components are grouped into categories such as engine and gearbox. The different component categories are divided into logical branches starting from the motor. For example, a ball screw is only placed in series after a belt gear. This is done by the condition that the belt has no possible connectors to other submodels in the later setup, because these are changed from rotatory to translatory by the ball screw. Therefore, all possible setups of feed axes are represented by one path each in the tree.

Due to this structure, different components can occur more often in the tree. However, this provides a high degree of traceability for the user. The nodes of the decision tree are enriched with domain knowledge. Thus, they contain all possible designs of the categories, such as belts and planetary gears for the category gearbox. In addition, knowledge about possible data sources of a component is also introduced. This can be used later for identification. The tree also shows the submodels of the components. Thus behind each expression of a node in the tree one or more models stand for the respective components. The system represents a superordinate structure for different component models. Component models can be implemented in their complexity adapted to the respective optimization application. So the models are regarded first as Black boxes and are defined by input/output variables. This opens the possibility of letting expert knowledge flow into the structure in the form of single component models. Fig.1 outlines the structure of the decision tree.

This approach enables the implementation of modular and self-configuring local monitoring functions, which is done here on the basis of a local force monitoring function.

3.1. Component identification

To build an axis-specific model, all components of the axis must be known. The concept provides two possibilities for this. In the first, the machine operator knows the structure of the feed axis. In this way, a skilled worker can select from existing components in his machine and thus determine the path in the decision tree. However, this involves an initial configuration effort on the part of the operator. To avoid this or to preserve missing knowledge of the machine, the second option offers an automated recognition of the components. This is based on a reference run. The structure of the reference run depends on the number and type of axes. These can be specified by the user or read out from the machine's control system.

When the reference run is performed, all possible data sources are recorded in the control. For identification, the decision tree is supplemented by rules. These rules are derived from an analysis of the features from the time series of the reference run. Thus, each node has one or more rules, which determine probabilities for the different paths of the tree by a fuzzy logic. In addition, probabilities are determined for the different subcategories of the nodes. The starting point is the motor node. To optimize the computing time, a threshold value can be defined for the probabilities. If this threshold is not exceeded, further probabilities in the subsequent nodes are not determined and only the promising configurations are pursued.

The rules are derived experimentally by time series data from tests with differently configured machine axes and axis test benches. In addition, the deposited domain knowledge is included. This allows existing data sources from the control system to confirm the existence of a component and thus increase the probability of the node. At the end of the step, all components are available to the system.

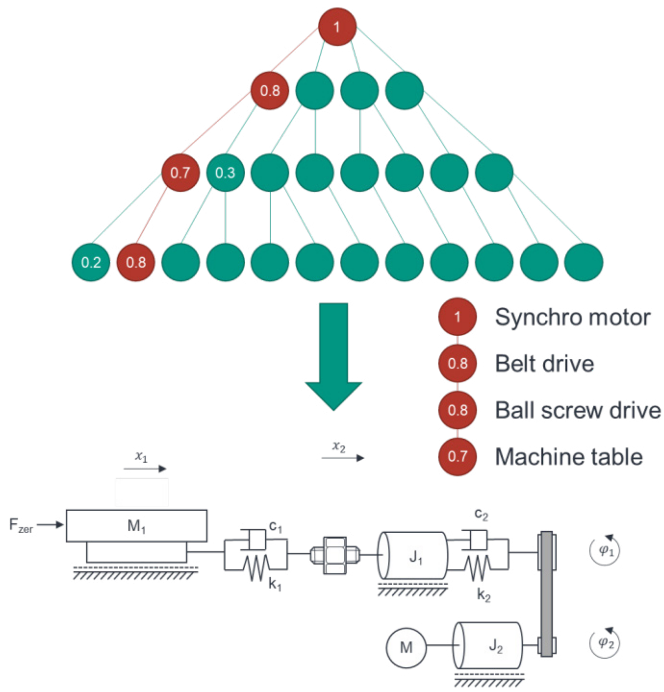

Fig. 2.. Creation of the overall model

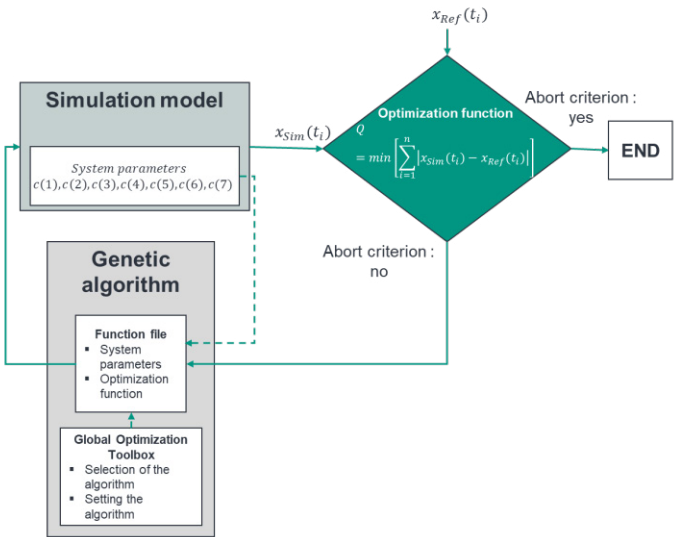

Fig. 3. Parameterization procedure.

3.2. Modeling

In the next step of the methodology, the individual overall model of the feed axis is created. The probabilities determined from the component identification are used for this. Starting from the node of the motor, the probabilities along each path are multiplied. This is done for all paths for which probabilities were determined. As a result, total probabilities are available for the different configurations of an axis. A ranking is created from the compositions available in this way.

The most probable configuration represents thereby the solution. Now the component models, which are contained in this solution, are taken from the decision tree. These are put together again starting from the engine at the defined interfaces and form the individual total model of the axis. The complexity of the models depends on the application. The overall model results in a parameter list for describing the model. Fig. 2 shows an example of the modeling process.

3.3. Parameterization

In the next step, the model parameters must now be determined. This is again data-based. In order to minimize the configuration effort of the overall system, the same reference run is again used for this as for component identification. For the determination of the parameters of the model a genetic algorithm is used for this. Global optimization algorithms are necessary because some submodels have a strong starting point dependency.

For optimization, the measured encoder signal of the glass scales of the real reference run and the position signal of the simulated reference run are used. The motor current of the real run is used as input to the model. Fig.3 shows the optimization process.

To limit the search space of the genetic algorithm, the stored model parameters are provided with domain knowledge. Thus, for the individual parameters in the components, value ranges are specified in which the parameter must lie, for example, a stiffness of a ball screw. If the quality criterion cannot be fulfilled with the available model, iterative jumps back to the model building step. There, the model that comes next in the ranking is created. With this model, the parameterization step is carried out again. As the complexity of the models increases, the complexity of the reference run must also increase in order to be able to determine the parameters of the models.

If the optimization criterion is again fulfilled, the parameterized model is available as a result. The model obtained in this way can be used to determine optimizations on the feed axis in order to increase the OEE of the plant.

3.4. Experimental Setup

To demonstrate the implementation of the third phase of the concept, a reference run was carried out with the X-axis of a 4-axis CNC machining center. A DMG DMC 60 H at the wbk Institute of Production Engineering from 1997 was used as the test machine for this purpose. The reference run was performed with the X-axis of the machine. To represent the simplest case of a reference run, the x-axis was moved by 400 mm. The specified feed rate was 5000 mm/min. The time series data were recorded at a sampling rate of 500 Hz with an edge device on the machine. The signals considered here are the current signal of the motor and the position signal at the glass scale of the stand. The system is shown in Fig. 4.

The configuration of the axis was determined using the decision tree and is composed as follows:

- Engine
- Belt drive

Fig. 4. DMC 60 H at wbk.

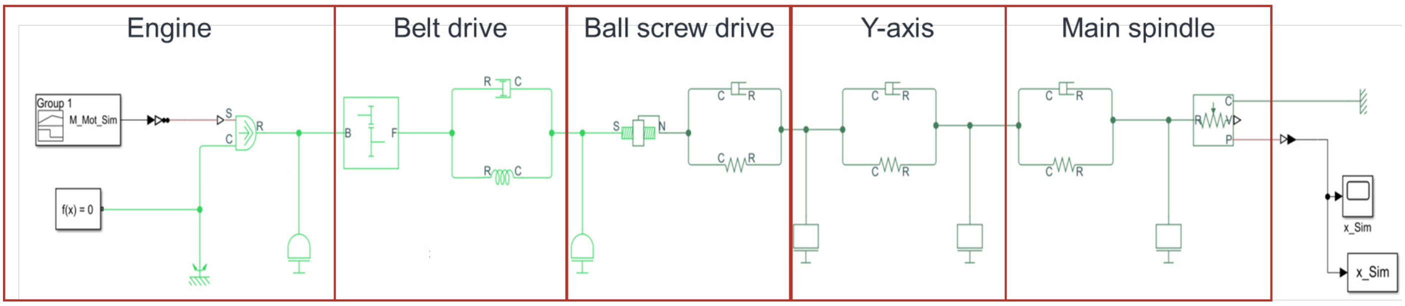

Fig. 5. Simulation model X-axis.

- Ball screw drive
- Vertical column
- Y-axis
- Main spindle

The submodels were built in Simulink and interconnected at their interfaces. The resulting Simulink model is shown in Fig. 5. The components can be described by parameters such as mass, stiffness, damping and moments of inertia.

4. Results and discussion

For testing the genetic algorithm, 7 quantities are determined in the system to ensure clarity. These are:

- Damper stand
- Stiffness stand
- Mass stand
- Damper main spindle
- Stiffness main spindle
- Mass y-axis
- Mass main spindle
- Mass main spindle

A genetic algorithm from MATLAB's Global Optimization Toolbox was used to determine the seven system parameters.

Table 1. Validation of the genetic algorithm

Components Parameter	Real value	Value - GA	Unit	Deviation in %
Damper stand	100000	100464.84	N/(m/s)	0.465
Stiffness stand	1500000000	1425000000	N/m	5
Mass stand	1000	950	kg	5
Damper main spindle	100000	102060.81	N/(m/s)	2.061
Siffness main spindle	100000000	95000000	N/m	5
Mass y-axis	500	475	kg	5
Mass main spindle	250	237.5	kg	5

The default settings of the Global Optimization Toolbox were used for the calculation. A 10-minute time limit was used as a termination criterion. The subsequent application of the algorithm in the simulation model for the reference run provides concrete results for the individual system parameters. Fig.5 shows the real and the simulated reference run after optimization. The objective function has a value of 43235.4 mm. This refers to a total of 2433 data points.

The basis for the validation of the algorithm are the real values of the seven system parameters from the data sheets and CAD models. The search space was limited to 5 percent of the real value. Table 1 below shows the real and determined values of the system parameters and the percentage deviations. The real values were determined from the data sheets and CAD models of the system. It is noticeable here that the values for the damping of the column and the spindle deviate the least from the real values. A large influence of these two parameters on the objective function is assumed, which is why the

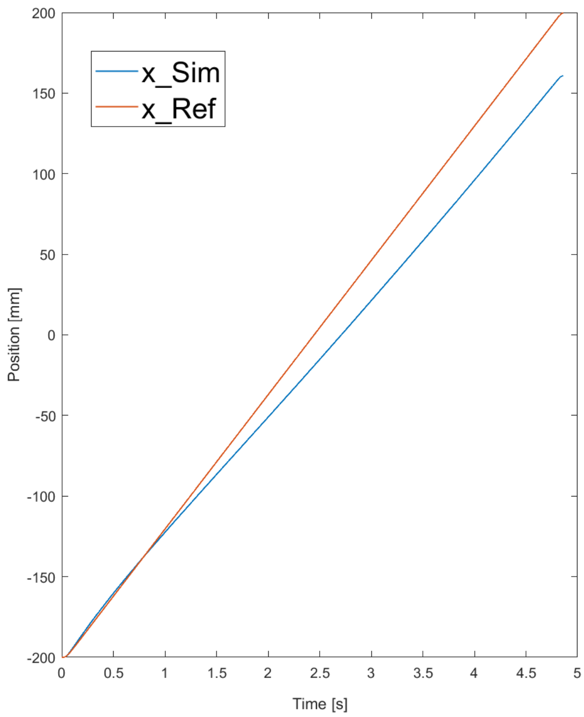

Fig. 6. Position of X-axis simulated (blue) and reference drive (orange).

algorithm mainly focused on the determination of the damping of the column and the spindle.

For the calculation of the masses and the stiffnesses, the algorithm was oriented to the lower slope of the value range defined in advance.

The results show that by a simple reference run and a simple spring-damper model of the axis, the damping of the system can be determined with small deviations by 2%, but the stiffnesses and masses of the system cannot be determined. Thus, a simple reference run and simplest models are not sufficient for a more detailed model parameterization.

5. Conclusion and Outlook

For the optimization of the OEE, models of the feed axis are required. These are often not available for existing machines and can only be determined at high cost and time. In addition, a different configuration of the feed axis makes a transferability of the models and the associated scalability of the application impossible. This paper presents an approach for the individual generation of dynamic feed axis models. It was shown that damping in the model can be determined by simple reference runs, but not masses and stiffnesses.

Future work will address, on the one hand, the derivation of the rules for the first phase of the methodology. For the second phase, it must be ensured that sufficiently complex models are used. In addition, the possibility of a more complex reference run must be investigated in order to be able to determine all parameters.

Acknowledgements

This paper is based on the research results of the project "Software-defined Manufacturing for the automotive and supplier industry – SDM4FZI". The project is funded by the German Federal Ministry of Economic Affairs and Climate Action. The authors of this publication thank the ministry for the funding.

References

[1] Barton D, Federhen J, Fleischer, J. (2021). Retrofittable vibration-based monitoring of milling processes using wavelet packet transform. Procedia CIRP; 2021, 96:353-358.

[2] Lechler A, Verl A. Software defined manufacturing extends cloud-based control. In International Manufacturing Science and Engineering Conference (Vol. 50749, p. V003T04A025). American Society of Mechanical Engineers; 2017.

[3] Gönnheimer P, Karle A, Mohr L, Fleischer J. Comprehensive machine data acquisition through intelligent parameter identification and assignment. Procedia CIRP; 2021. 104:720-725.

[4] Albertelli P, Cau N, Bianchi G, Monno M. The effects of dynamic interaction between machine tool subsystems on cutting process stability. The International Journal of Advanced Manufacturing Technology; 2012. 58(9):923-932.

[5] Brecher C, Fey M, Bäumler S. Damping models for machine tool components of linear axes. CIRP Annals; 2013. 62(1):399-402.

[6] Frey S. Verbesserung des dynamischen Verhaltens von Vorschubantrieben durch semiaktive Dämpfung; 2013.

[7] Hung J P, Lai Y L, Luo T L, Su H C. Analysis of the machining stability of a milling machine considering the effect of machine frame structure and spindle bearings: experimental and finite element approaches. The International Journal of Advanced Manufacturing Technology; 2013. 68(9):2393-2405.

[8] Schwarz S. Prognosefähigkeit dynamischer Simulationen von Werkzeugmaschinenstrukturen (Vol. 313). Herbert Utz Verlag; 2016.

[9] Saglam H, Unuvar A. Tool condition monitoring in milling based on cutting forces by a neural network. International Journal of Production Research; 2003. 41(7):1519-1532.

[10] Chen G, Zhu S, Zheng Q. Dynamic parameters' identification for the feeding system of computer numerical control machine tools stimulated by G-code. Advances in Mechanical Engineering, 7(8), 1687814015597830; 2015.

[11] Erkorkmaz K, Altintas Y. High speed CNC system design. Part II: modeling and identification of feed drives. International Journal of Machine Tools and Manufacture; 2001. 41(10):1487-1509.

[12] Wang X, Liu B, Mei X, Yun X, Li X, Li Q, Wang X. Genetic algorithm for dynamic parameters estimation of the machine tool worktable using the residual vibration signal. Journal of Vibration and Control, 1077546321993579; 2021.

[13] Dörrer F, Otto A, Kolouch M, Frieß U. Modularer digitaler Zwilling für Werkzeugmaschinen. Zeitschrift für wirtschaftlichen Fabrikbetrieb; 2021. 116(9):622-626