Chapter 10
Holistic Approach for Condition Monitoring in Industrial Product-Service Systems

Eckart Uhlmann, Christian Gabriel, Abdelhakim Laghmouchi, Claudio Geisert and Niels Raue

Abstract The content of the book chapter is the development and application of a method for the cost optimized integration of a condition monitoring system for machine tools. An environment for the simulation of lifecycle costs considering the maintenance processes as well as the hardware will be described to identify an appropriate sensor concept. The approach of event based simulation allows an assessment of possible sensor concepts depending on the machine tool's performance. To realize a cost-effective condition monitoring solution using simple consumer electronics, such as Micro-Electro-Mechanical-Systems, and to provide high scalability a wireless sensor network has been developed and evaluated. It can be easily adapted to different specific applications because of decentralized data preprocessing on the sensor nodes, as well as services in the cloud. Within this network the sensors interact through a software agent system which is implemented in the machine tool and all of its subsystems. The Java Agent Development Framework will be used as a middleware. The modularization leads to a highly flexible system. Additionally, the agent system enables the interaction between the machine tool, the IPS^2 provider, and its service technicians. The book chapter includes the evaluation of the method in the field of grinding machine tools by means of a feed axis.

E. Uhlmann (✉) · A. Laghmouchi · C. Geisert
Fraunhofer Institute for Production Systems and Design Technology IPK, Pascalstrasse 8–9, 10587 Berlin, Germany
e-mail: eckart.uhlmann@iwf.tu-berlin.de

A. Laghmouchi
e-mail: abdelhakim.laghmouchi@ipk.fraunhofer.de

E. Uhlmann · C. Gabriel · N. Raue
Institute for Machine Tools and Factory Management, TU Berlin, Pascalstrasse 8–9, 10587, Berlin, Germany

L. Redding et al. (eds.), *Advances in Through-life Engineering Services*, Decision Engineering, DOI 10.1007/978-3-319-49938-3_10

10.1 Introduction

The perception of industrial services has changed during the past years. Nowadays, services are an integral part of the offer to fulfill the customers' demands. Therefore, services are no longer seen as an add-on to the core product. This development is due to two causes. Firstly, in the service sector increased margins can be achieved compared to the selling of products [5]. Secondly, the consistent orientation on customer needs leads to a stable customer provider relationship and thus advantages for all stakeholders involved.

Approaches for the transformation of companies from a product seller to a solution seller, where industrial services are an equal part, are subsumed under term servitization [10]. Product Service-Systems (PSS) are a particular form of servitization as the focus lies on the use of product shares and not merely on the selling of products. As a consequence, the customers benefit from a restructuring of the risks, responsibilities, and costs traditionally associated with ownership [1].

PSS for the industrial sector are called Industrial Product-Service Systems (IPS^2) and are characterized by the integrated and mutually determined planning, development, provision, and use of product and service shares including its immanent software components in Business-to-Business applications and represents a knowledge-intensive socio-technical system [5]. The individuality of IPS^2 can lead to innovative IPS^2 business models, since the IPS^2 provider guarantees specific machine tool availability or certain amounts of parts manufactured by the machine tool [5]. Therefore, the IPS^2 provider faces difficulties in predicting and controlling risks and uncertainties that usually would be the problem of the customer [8].

10.2 Objectives and Approach

IPS^2 allow the individual configuration of solutions consisting of product and service shares which is an advantage for customers towards the classical approach of buying goods. Therefore, a wide variety of external and internal influences on customer site have to be taken into account in the machine tool industry. In the case where the IPS^2 provider covers the guarantee of machine tool availability an appropriate and customer individualized concept for maintenance, repair, and overhaul (MRO) activities is crucial.

One objective of this book chapter is the development of a holistic approach to develop and simulate a customer provider relationship where a specific machine tool availability is guaranteed. Thus, the IPS^2 provider is able to quantify the effort considering the MRO costs over the entire lifecycle. In the development phase a business process model and notation (BPMN) based workflow management system is used to model and simulate the necessary activities for different MRO concepts.

These concepts differ regarding the existence of a condition monitoring for proactive maintenance or regarding the methods for spare parts supply. The cost efficiency of these concepts depends on different factors on customer site. Machine tools, which operate in a three-shift production, have different requirements concerning MRO concepts than machine tools in a one-shift production. In the same way the requirements differ in a cycle operation and in a job shop production with buffers.

As soon as the existence of a condition monitoring is necessary for the machine tool operation the IPS^2 provider is interested in a cost efficient solution for sensors and evaluation electronics. The second objective of this chapter is the development and application of a cost effective wireless sensor network for distributed condition monitoring. This wireless sensor network enables the IPS^2 provider to upgrade machine tools with low effort.

10.3 Method for Ensuring the Availability of IPS^2

10.3.1 Concept

A method is specified by activities, roles, specification documents, techniques, and a meta model [2, 11]. In the framework of this chapter the method for ensuring the availability of IPS^2 shall be described by means of activities and tools. The sequence of activities considers the IPS^2 lifecycle which consists of the phases planning, development, implementation, operation, and resolution [9]. Figure 10.1 shows the assignment of activities and tools to the IPS^2 lifecycle phases.

In an availability-oriented IPS^2 business model the provider prices the solution according to a specific technical availability. Therefore, the effort for the maintenance activities has to be calculated at an early stage of the IPS^2 lifecycle. In the framework of this chapter a workflow management system has been used to model and simulate the maintenance costs according to the components' properties and their operation conditions over the entire lifecycle. In the case where the integration of a condition monitoring solution represents the optimal approach the machine tool components have to be extended in the implementation phase by the integration of additional hardware by means of sensors and software for the data acquisition and analysis. During the IPS^2 operation the periodic acquisition of sensor data allows the classification of the results and therefore the assessment of wear margin for a specific component. In the following chapter the activities of the method will be described in particular.

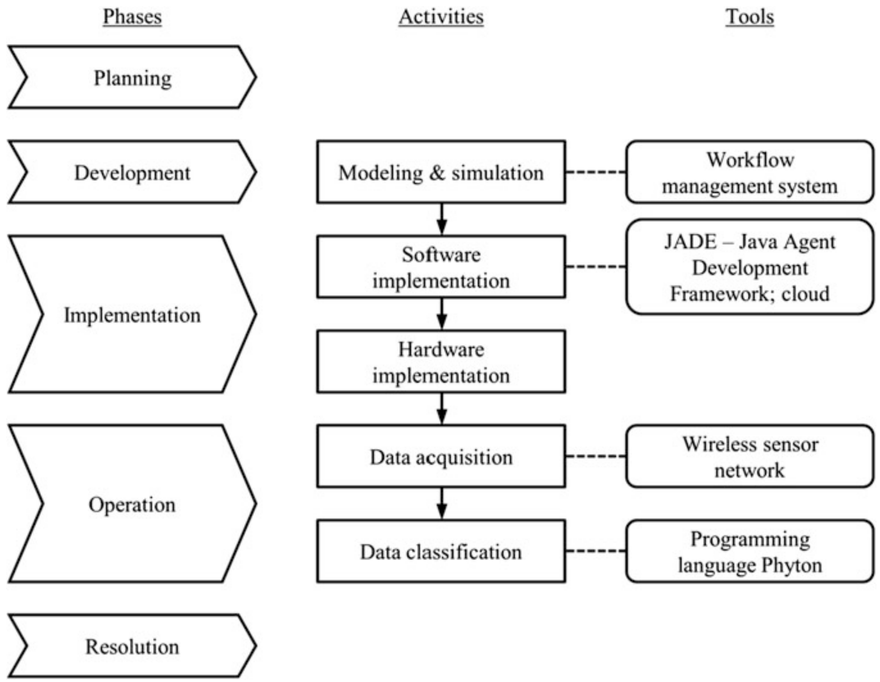


Fig. 10.1 Assignment of activities and tools to the IPS^2 lifecycle phases

10.3.2 Activities

10.3.2.1 Modeling and Simulation

Concrete activities shall be described by means of a ball screw spindle used in an NC-controlled centerless grinding machine feed axis. In the development phase the IPS^2 provider has to choose an appropriate concept to minimize the maintenance costs. This can be achieved by a reactive maintenance concept, the storage of spare parts on customer site, or an implemented condition monitoring system. According to the information of a machine tool manufacturer, the ball screw spindle exchange is necessary approximately every 5 years. As a consequence the ball screw spindle has to be exchanged twice regarding a machine tool's lifetime of 15 years. Figure 10.2 shows the generic business process for the exchange of components. This process model was created using the workflow management system IYOPRO developed by the company intellivate GmbH.

The process execution varies depending on the existence of a condition monitoring system or the storage of spare parts on customer site. If the condition monitoring is nonexistent, the service technician arrives at the customer site, analyzes the error, and checks the on-site availability of the necessary spare part. In cases where the appropriate spare part is not available a procurement activity is

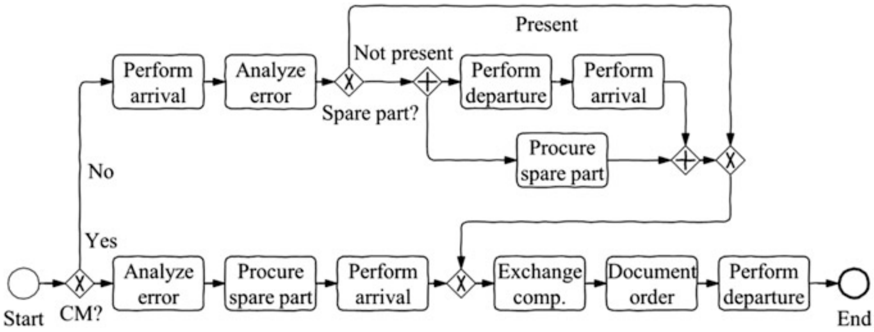


Fig. 10.2 Generic business process model for the exchange of components

Table 10.1 Durations of activities

Activity	Duration (h)			
	Reactive maintenance	Spare part	Wireless sensor network	Actual condition monitoring
Arrival	4	4	4	4
Spare part procurement	48	0	0	0
Analysis	1	1	0.25	0.25
Exchange	3	3	3	3
Documentation	0.5	0.5	0.17	0.17
Departure	4	4	4	4

necessary which may last several days and causes machine downtime. During that time the service technician departs and returns as soon as the spare part is available. After exchanging the component the service technician documents the order and departs from customer site. In the case where a condition monitoring is available the business process starts with the analysis of the error under consideration of appropriate sensor data and the procurement of spare parts. Subsequently, the service technician arrives at customer site and handles the order.

To identify the optimized maintenance concept the process model will be simulated applying the discrete event-based simulation (DES) method. IYOPRO allows the investigation of process models due to the implemented interface to the JAVA-based open-source bibliography DESMO-J [4]. Therefore, the process model has to be supplemented by durations of activities and costs of resources. For the durations of activities the four cases reactive maintenance, spare part, wireless sensor network, and actual condition monitoring were differentiated (Table 10.1).

Table 10.2 shows the costs of resources in the present process model. The costs for machine downtime differ with regard to the production type and the existence of buffers. The costs for a machine hour with buffers decrease in a three-shift production due to the optimized degree of utilization. Regarding a production type

Table 10.2 Costs of resources

Resources	Costs (€)			
	One-shift with buffer	One-shift without buffer	Three-shift with buffer	Three-shift without buffer
Machine downtime per hour	71.65	600	54.91	400
Service technician per hour	46.54	46.54	46.54	46.54
Travelling costs per trip (one-way)	100	100	100	100

without buffers the machine costs per hour are several times higher. The costs were estimated for the application scenario of an NC-controlled centerless grinding machine in a job shop production. Depending on the type of machine tools these costs can amount up to several thousand euros.

The process model was simulated for the four cases one-shift with buffer, one-shift without buffer, three-shift with buffer, and three-shift without buffer (Fig. 10.3). The reactive maintenance concept increases the maintenance costs in all four scenarios. This is due to the fact that the procurement of the necessary spare part may lead to a machine stop over several days.

Another interesting finding relates to the spare parts supply. In production scenarios with buffers between particular machine tools the storage of spare parts on customer site causes lower costs than the implementation of a condition monitoring according to the state of the art.

The implementation of the wireless sensor network for a proactive maintenance concept causes the lowest costs in all cases. This is primarily the result of the lower hardware and software costs compared to actual condition monitoring solutions. The development of the software and hardware will be described in the following chapters.

10.3.2.2 Software Implementation

The development of software tools ensures the interfaces between the components of the overall system (Fig. 10.4). The concept consists of the machine tool component, the wireless sensor network, which includes a MEMS vibration sensor, a minicomputer (Raspberry), the cloud (including services), and a smart device for remote access and visualization.

The data acquisition and processing is realized on the sensor node level. Furthermore, the classification of the features extracted from the diagnosis signal is transmitted to the cloud server. On the cloud server there are applications and services available for maintenance planning, remote condition monitoring, etc.

Via smart device the service technician or maintenance planner can receive the reports generated using a condition monitoring processing unit. This allows remote

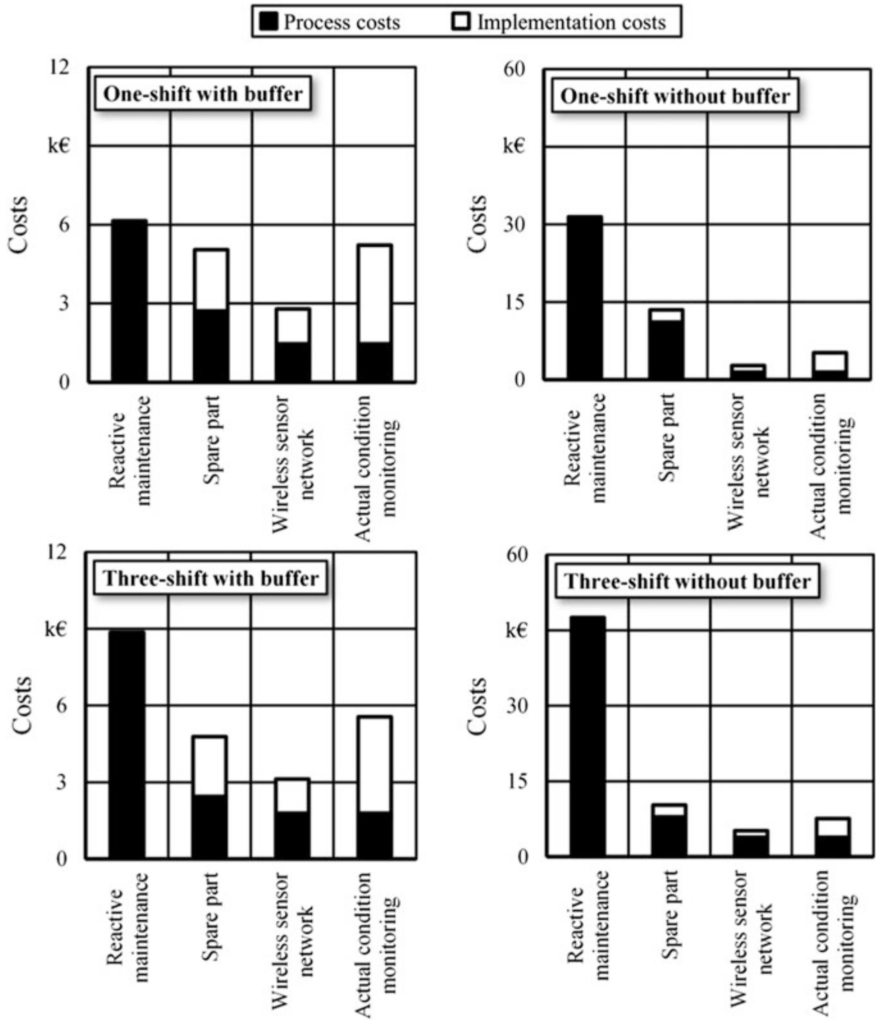


Fig. 10.3 Results of the simulation experiment

continuous monitoring of the production system and enables condition-based maintenance decision making of the system concerned.

The communication between these subsystems has been realized by the implementation of a software agent system. The Java Agent Development Framework (JADE) will be used as a middleware (Fig. 10.1). The internal communication between the software agents is performed through the asynchronous exchange of agent communication language (acl) messages. Two software agents interact by means of defined pattern. A communication always consists of an initiator and a responder. Initially, an agent demands another agent's readiness to execute a service

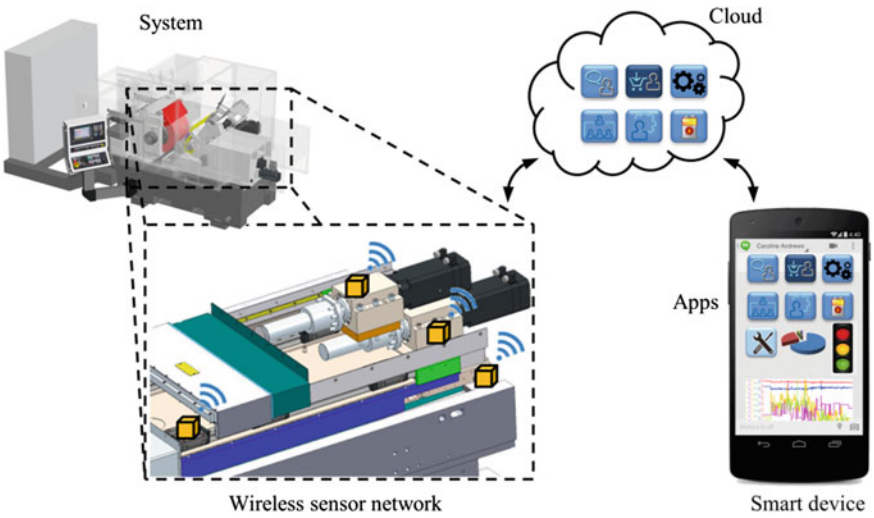


Fig. 10.4 Wireless sensor network on the production system

by a ‘call-for-proposals’ (cfp). As soon as the service is executable the second agent replies ‘propose’. Afterwards, the initiator agent executes the service by sending the message ‘accept-proposal’. After finishing the execution of the service the responder agent replies ‘inform-done’. This general procedure will be performed until the overall process execution is finished. Figure 10.5 exemplarily shows a part of the communication between several agents for the execution of the feed axis test run.

Initially, the machine tool agent secures the readiness of the cloud for the collection of data. If the cloud agent agrees the request, the axis test run can be ordered. The directory facilitator (DF) agent, a directory which announces which agents are available on the platform, informs about the necessary services to execute the test run. The agents and services will be blocked for the duration of the execution. Subsequently, the machine tool agent communicates consecutively with the agents to particularly perform the activities of the axis test run. After finishing the test run the result data are submitted to the cloud. The comparison of the results with previous test runs allows the assessment of wear margin of the feed axis.

10.3.2.3 Hardware Implementation

The modifications of the components will be described by means of a feed axis test rig. It incorporates a top slide, an inside slide, and a machine bed. Furthermore, two asynchronous motors drive the ball screw spindles, which move the slides along the unloading axis X4, and the infeed axis and recessing axis X1 in steps of 0.2 µm. Figure 10.6 shows a schematic representation of the axis test rig.

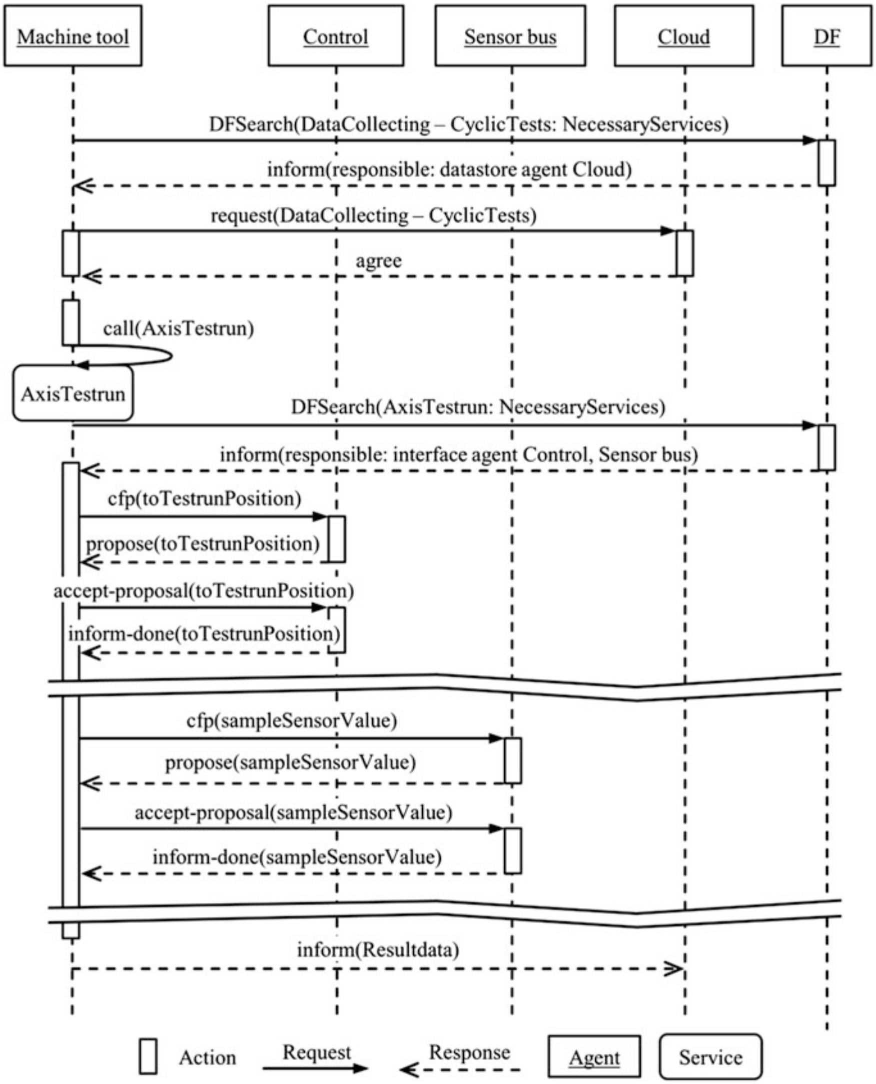


Fig. 10.5 Unified modeling language (UML) interaction diagram of the agent communication during the feed axis test run

The wireless sensor network is composed of four individual nodes which can act independently from each other. Each of these nodes is comprised of a MEMS temperature and vibration sensor. Furthermore, a Micro-Electro-Mechanical-Systems (MEMS) digital output motion sensor with 3-axes 'nano' accelerometer is used (Table 10.3).

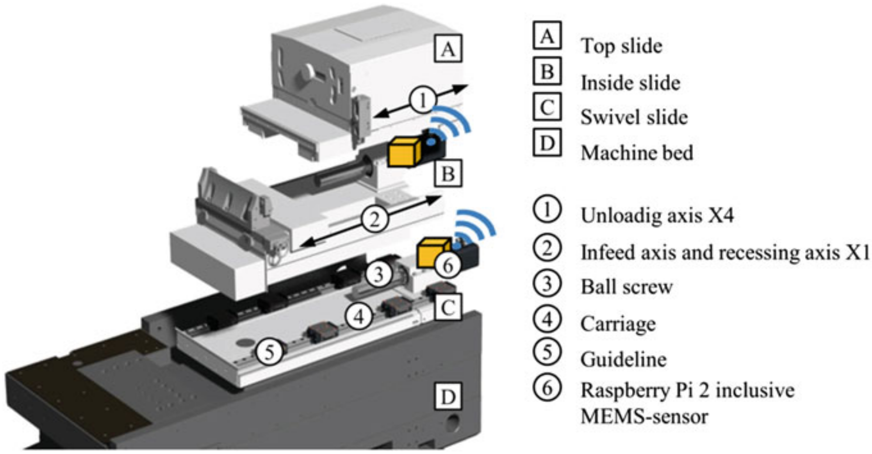


Fig. 10.6 Schematic representation of the axis test rig

Table 10.3 MEMS sensor LIS3DH specification

Feature	Value
Measuring accelerations with output data rates	1 Hz–5 kHz
Wide supply voltage	1.71–3.6 V
Ultra-low power mode consumption	<2 µA
Dynamically selectable full-scale	±2 g/±4 g/±8 g/±16 g
Data output	16 bit
Operating range	−40 to +85 °C
Shock survivability	10,000 g
Digital output interface	I2C/SPI

10.3.2.4 Data Acquisition

A design of experiments was developed for the acquisition of training data in order to evaluate the implemented algorithms. For this purpose, reproducible damages were created using the laser powder cladding method. A five-axes Trumpf TruLaser 7020 three dimensional laser cutting machine with various laser spot diameters was used to create a small and a heavy damage on the surface of the spindles (Fig. 10.7).

In addition, a fault was created on the needle roller/axial cylindrical roller bearing spindle. The acceleration of the spindles was measured on the axis test rig. They were conducted repeatedly under exactly the same conditions. The vibration was measured using a MEMS sensor with a sampling rate of 3,000 Hz. Figure 10.7 shows the collected vibration data at 1 and at 3 m/min for three different classes. It also shows the measured vibration data using MEMS sensor at the axis test rig at different operation speed.

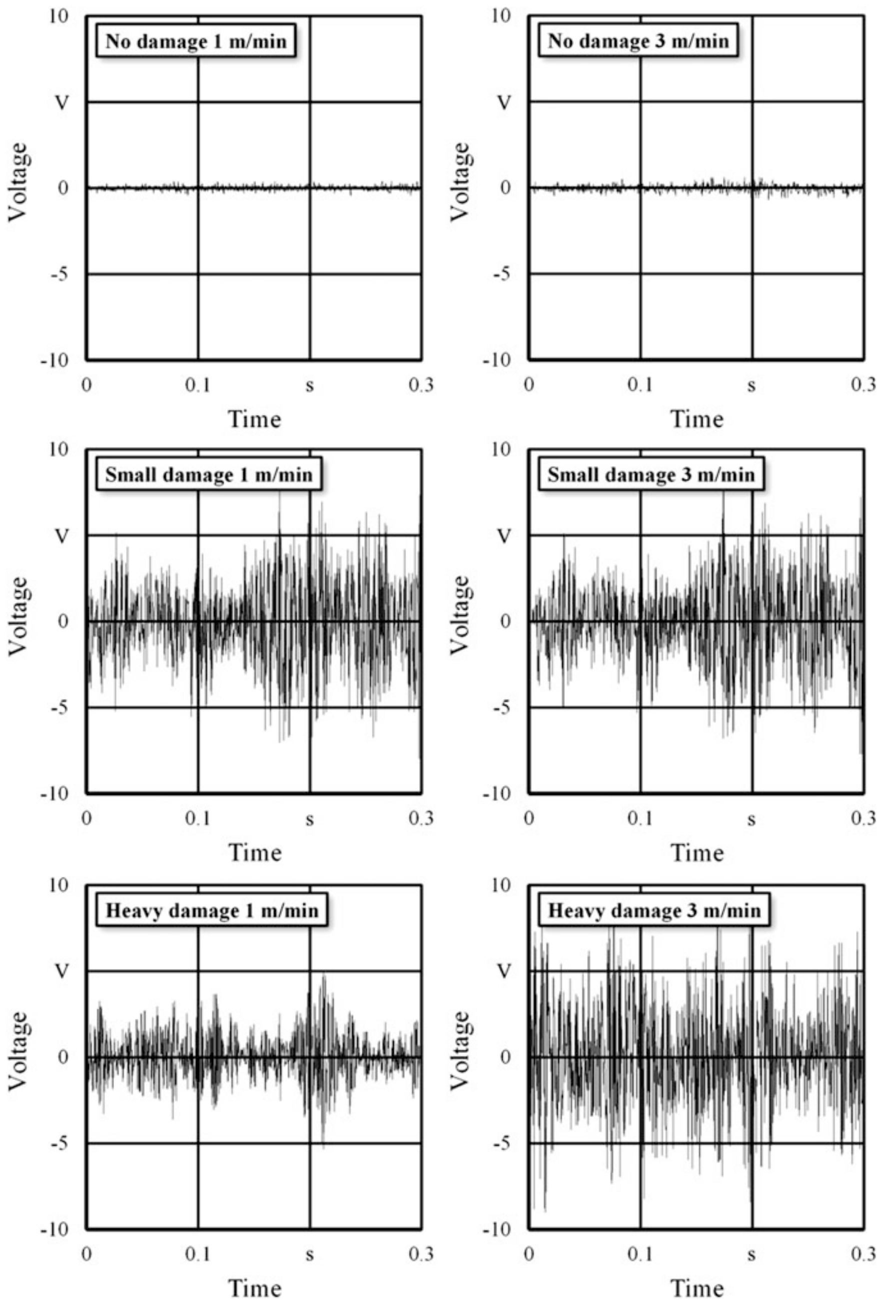


Fig. 10.7 Measured vibration data using MEMS sensor

10.3.2.5 Data Classification

The feature extraction using Python is realized on the Raspberry (Fig. 10.1). Python was selected to be used for this application because it is a fast programming language widely used in the field of data analysis using machine learning methods. To enable the feature calculation and classification on the Raspberry, extensions to the Python programming language such as Numpy, SciPy, Matplotlib, and Scikit-learn were used.

For the calculation of the selected features, such as statistical values (variance, mean, and kurtosis), the Scipy library is used. It includes the functions for the calculation of the statistical values.

For the classification of the features extracted, a support vector machine (SVM) algorithm is implemented [7]. SVM is a computational learning method based on the statistical learning theory. This function acts as an expert system which is based on the Vapnik-Chervonenkis (VC) Theory [6] and has recently emerged as a general mathematical framework for estimating dependencies from finite samples. VC theory combines fundamental concepts and self-consistent mathematical theory, well-defined formulation and principles related to learning. SVM is used successfully in many classification problems like text categorization, image classification, and bioinformatics.

For fault detection and diagnosis in industrial applications SVM is developed for recognizing patterns in the collected sensor data. These are classified to predefined fault condition of the considered component [3].

The most significant benefit of SVM is higher efficiency in high dimensional nonlinear classification problems while other statistical classifiers often fail in achieving such efficiency. The idea is to maximize the margin between hyper plane and the training examples. This can be done by finding the optimal hyper plane which has maximal margin.

After signal preprocessing, the statistical features are extracted. For the purpose of classification appropriate data has been selected. The implementation of the SVM algorithm on the Raspberry is done in Python. The feature extraction and SVM algorithm was previously implemented and tested in a MATLAB environment. Figure 10.8 represents selected features for the classification step. The well-defined areas in the diagram show that the features selected (mean, kurtosis, and variance) for the classification step of the data at 1 m/min are correctly chosen.

The classification accuracy using different test data at 1 and 3 m/min is illustrated in Fig. 10.9.

The graph shows that the features selected for the classification (Fig. 10.7) for no damage and heavy damage at 1 m/min is 100%. In contrast, the accuracy of SVM classification of the test data for small damages did not reach 100% for both speeds. To increase the accuracy for small damages other feature combinations has to be selected.

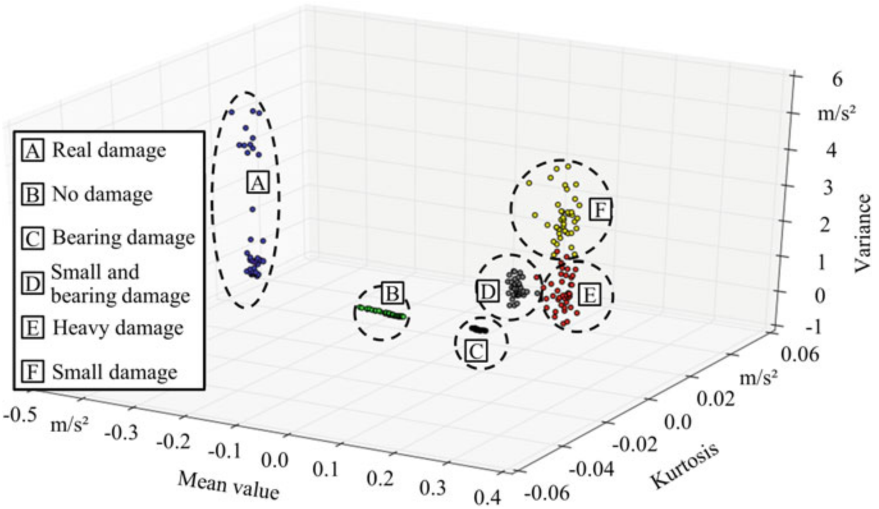
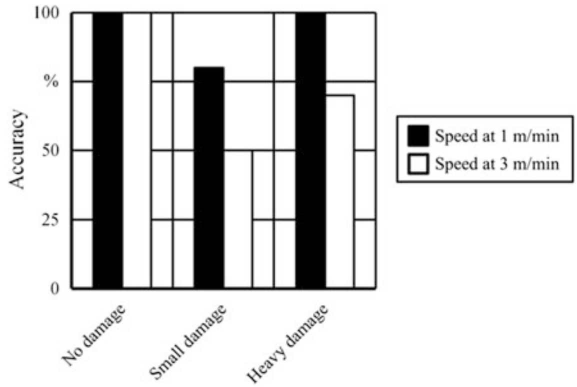


Fig. 10.8 Presentation of the suitable feature for the classification on the feature space

Fig. 10.9 SVM classification accuracy



10.4 Summary

This chapter presented a method for the customer individual modeling, simulation, and implementation of an appropriate MRO strategy for an availability-oriented IPS². Therefore, a generic business process for the exchange of worn components has been modeled, implemented, and simulated using the workflow management system IYOPRO developed by the company intellivate GmbH. The development of a simulation model allowed the comparison of different MRO strategies regarding the costs, i.e. occurred by a worn ball screw spindle of a feed axis. In this chapter the four cases reactive maintenance, spare part supply, and two condition monitoring approaches were compared and the most suitable concept for a grinding machine at the customer site was identified.

Furthermore, the development and implementation of a condition monitoring system, consisting of a wireless sensor network using Raspberry Pi 2 modules and MEMS vibrations sensors, has been described. The use of this network reduces the costs compared to industrial sensors and allows an easy adaptability to different specific applications due to decentralized data preprocessing, feature extraction, selection, and classification on the sensor node level.

The implementation of different classification and clustering algorithms on the sensor nodes will be addressed during further research activities. This will enable the condition monitoring steps (signal preprocessing, feature extraction, and classification) of the acquired data directly on the node level.

Acknowledgements We express our sincere thanks to the Deutsche Forschungsgemeinschaft (DFG) for funding this research within the collaborative research project SFB/TR 29 on Industrial Product-Service Systems—dynamic interdependency of products and services in the production area. Furthermore, we express our sincere thanks to the company intellivate GmbH for providing a license of IYOPRO Premium.

References

1. Baines TS, Lightfoot H, Steve E et al (2007) State-of-the-art in product service-systems. P I Mech Eng B-J Eng 221(10):1543–1552. doi:10.1243/09544054JEM858
2. Braun C, Wortmann F, Hafner M et al (2005) Method construction—a core approach to organizational engineering. In: Haddad H, Liebrock LM, Omicini A et al (eds) Proceedings of the 2005 ACM symposium on applied computing (SAC), Santa Fe, New Mexico, USA, March 13–17, 2005. ACM, New York, pp 1295–1299
3. Joachims T (2002) Learning to classify text using support vector machines. Springer, New York
4. Joschko P, Haan J, Janz T et al (2012) Business process simulation with IYOPRO and DESMO-J. In: Bruzzone AG, Buck W, Cayirci E et al (eds) Proceedings of the international workshop on applied modeling and simulation, Rende
5. Meier H, Roy R, Seliger G (2010) Industrial Product-Service Systems—IPS^2. CIRP Ann Manuf Technol 59(2):607–627. doi:10.1016/j.cirp.2010.05.004
6. Shawe-Taylor J, Cristianini N (2004) Kernel methods for pattern analysis. Cambridge University Press, Cambridge
7. Steinwart I, Christmann A (2008) Support vector machines. Springer, New York
8. Tukker A (2004) Eight types of product-service system: eight ways to sustainability? Experiences from SusProNet. Bus Strategy Environ 13(4):246–260. doi:10.1002/bse.414
9. Uhlmann E, Bochnig H (2012) PSS-CAD: an approach to the integrated product and service development. In: Shimomura Y, Kimita K (eds) The philosopher's stone for sustainability—Proceedings of the 4th CIRP international conference on industrial product-service systems, Tokyo, Japan, November 8th–9th, 2012. Springer, Heidelberg, pp 61–66
10. Vandermerwe S, Rada J (1988) Servitization of business: adding value by adding services. EMJ 6(4):314–324. doi:10.1016/0263-2373(88)90033-3
11. Winter R, Schelp J (2006) Reference modeling and method construction—a design science perspective. In: Haddad H (ed) Proceedings of the 2006 ACM symposium on applied computing (SAC), Dijon, France, April 23–27, 2006. ACM, New York, pp 1561–1562