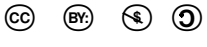
University of Trento

Dott. Antonio Di Dino

DESIGN AND DEVELOPMENT OF A DISTRIBUTED SYSTEM FOR MONITORING OF MACHINE TOOL BEHAVIOR

Prof. Francesco Biral
Prof. Paolo Bosetti

2014



UNIVERSITY OF TRENTO

Design and Development of a Distributed Systems for Monitoring of Machine Tool Behavior

Final Examination 07 / 04 / 2014

Board of Examiners
Prof. Oreste Salvatore Bursi (Università degli Studi di Trento)
Prof. Dionisio P. Bernal (Northeastern University, Boston)
Prof. Michel Destrade (National University of Ireland Galway)
Dott. Andrea Giovanni Calogero (Università Milano-Bicocca)
Dott. Paola Falugi (Imperial College London)

SUMMARY

The following thesis describes the development of a distributed system for the monitoring of numerical control machine tools. This work discusses all the design steps from the conceptualization of the hardware architecture to the development and implementation of monitoring algorithms. In the first chapter the state of art in machining automation will be presented, with a particular focus on monitoring and control system for machining operations. The second chapter describes the architecture of the proposed system both from the hardware and the software side. The third chapter goes into detail of the developed monitoring algorithms intended to the application on milling machines. The implementation and results of experimental tests will be discussed. Finally the fourth chapter proposes a new approach applied to the control of linear motion systems. The work presented in this thesis belongs to the applied research field that aims to enhance the automation level of machine tools by developing innovative techniques for the monitoring and control of machining process. The proposed monitoring system has been developed considering as key requirements the possibility to properly operate in several working conditions, the complete integration with the machine tool structure and the ease of use for unskilled personnel. The developed algorithms include the monitoring and mitigation of cutting vibrations, the detection and diagnosis of faults in spindle bearing and the emergency halt of the machine in case of collision. The resulting system provides a flexible and scalable framework easily adaptable to the specific machine tool and machining application. The monitoring tasks allow a fast setup and their execution is mostly automated, requiring a limited interaction with the machine tool end users. In conclusion the monitoring system improves the automation level of the machine tool providing a better control on the process execution. In addition it facilitates the assessment of the machine behavior allowing the objective evaluation of the operative conditions providing a useful support tool for the machine operator.

SOMMARIO

La seguente tesi descrive lo sviluppo di un sistema distribuito per il monitoraggio delle macchine utensili a controllo numerico impiegate nelle lavorazioni di fresatura. Questo lavoro tratta tutte le fasi di progettazione del sistema di monitoraggio a partire dalla concettualizzazione dell' architettura hardware fino allo sviluppo e implementazione degli algoritmi di monitoraggio. Nel primo capitolo verrà riportato lo stato dell'arte nel campo dell'automazione delle macchinine utensili focalizzandosi in particolare sui sitemi di monitoraggio e controllo delle lavorazioni. Nel secondo capitolo verrà descritta l'architettura del sistema proposto sia nella parte hardware che in quella software. Il terzo capitolo entra nel dettaglio degli algoritmi di monitoraggio sviluppati, fornendo la descrizione dell'implementazione e i risultati delle prove sperimentali. Il quarto capitolo infine riporta lo studio di una nuova metodologia applicata al problema del controllo di assi lineari. Il lavoro descritto in questa tesi si inserisce nel campo della ricerca applicata che mira a incrementare il livello di automazione delle macchine utensili, sviluppando metodi innovativi per il monitoraggio e il controllo dell'esecuzione del processo. Il sistema proposto è stato progettato considerando come requisiti chiave la possibilità di adattarsi alle diverse condizioni di lavorazione, la completa integrazione con la macchina utensile e la facilità di utilizzo da parte di utenti senza specifiche competenze. Gli algoritmi implementati riguardano il monitoraggio delle vibrazioni in fresatura, l'individuazione e la diagnosi di guasti nei cuscinetti mandrino e l'arresto della macchina in caso di collisione. Il sistema ottenuto fornisce una struttura flessibile e scalabile facilmente adattabile all'applicazione specifica. Le attività di monitoraggio sono facilmente conigurabili e vengono eseguite per la maggior parte in modo automatizzato richiedendo un interazione ridotta con l'utente della macchina. In conclusione il sistema di monitoraggio sviluppato rappresenta una soluzione facilmente adattabile a diverse tipologie di macchine utensili e di lavorazioni meccaniche. Le attività di monitoraggio implementate migliorano il livello di automazione della macchina utensile permettendo un maggiore controllo sull'esecuzione del processo. Inoltre forniscono una serie di valutazioni oggettive sullo stato della lavorazione e della macchina utensile rappresentando un utile strumento di supporto per l'operatore a bordo macchina.

ACKNOWLEDGEMENTS

The achievements of this thesis and my entire PhD career were made possible thanks to the help of several people among my professors, colleagues and family to which goes all my gratitude. First and foremost, I would like to thank my advisors Professor Francesco Biral and Professor Paolo Bosetti for encouraging and supporting my activities and for their continuous motivation to enhance my growth as a research scientist. In addition I would like to express my appreciation to PAMA and in particular to Matteo Grott and Alberto Nainer for their precious collaboration. A special thanks goes to Roberto Sartori for everything he taught me about programming. I would also like to show gratitude to Professor Nathan van de Wouw, Bram Hunnekens and Niels van Dijk that allowed me to collaborate with them at the University of Eindhoven and at Philips. It was a very important and formative experience and the last section of this thesis is owed to the time spent working with them. Further I would like to thank Professor Stefania Bruschi and Professor Luca Zaccarian for their help and kind willingness. A special thanks goes to my colleagues Maximiliano and Marco that shared with me the joy and sorrow of PhD career. Finally the last but not least thanks goes to my wife that accompanied, encouraged and stood me in this three years.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LISTINGS

ACRONYMS

ACC	Adaptive Control with Constraints
ACO	Adaptive Control with Optimization
ADC	Analog to Digital Converter
AE	Acoustic Emission
AI	Artificial Intelligence
ANN	Artificial Neural Networks
APC	Average Power Content
BN	Bayesian Networks
BPFI	Ball Pass Frequency of the Inner race
BPFO	Ball Pass Frequency of the Outer race
BSF	Ball Spin Frequency
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CD	Collision Detection
CM	Condition Monitoring
CMM	Coordinates Measurement Machines
CNC	Computer Numerical Control
DAC	Digital to Analog Converter
DFT	Discrete Fourier Transform
DOF	Degree of Freedom
DSP	Digital Signal Processing
DWT	Discrete Wavelet Transform
EA	Envelope Analysis
EBI	External Bus Interface
EDM	Electrical Discharge Machining
EEPROM	Electrically Erasable Programmable Read-Only Memory

ES	Extremum Seeking
FEM	Finite Element Modeling
FFT	Fast Fourier Transform
FL	Fuzzy Logic
FRF	Frequency Response Function
FSR	Full Scale Range
FTF	Fundamental Train Frequency
GAC	Geometric Adaptive Control
ICT	Information and Communication Technology
IIR	Infinite Impulse Response
MCU	MicroController Unit
MISO	Master Input Slave Output
MMI	Man Machine Interface
MRR	Material Removal Rate
MOSI	Master Output Slave Input
NC	Numerical Control
OMA	Operational Modal Analysis
PHM	Prognostic Health Management
PID	Proportional Integrative Derivative
PS	Power Spectrum
PSD	Power Spectral Density
RAM	Random Access Memory
RMS	Root Mean Square
ROM	Read-Only Memory
SCL	Serial Clock
SCLK	Serial Clock
SDA	Serial Data line
SDRAM	Synchronous Dynamic Random Access Memory
SK	Spectral Kurtosis

SLD	Stability Lobe Diagram
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SS	Slave Select
STFT	Short Time Fourier Transform
SVM	Support Vector Machines
TCM	Tool Condition Monitoring
TWI	Two Wire Interface
UART	Universal Asynchronous Receiver-Transmitter
UDP	User Datagram Protocol
VGC	Variable Gain Controller
VM	Vibration Monitoring

Part I

MACHINING AUTOMATION

1

INTRODUCTION

Automation in machining process plays a key role in achieving high production performances in terms of working time and parts quality. Despite the big research effort, automation is mostly applied to the execution of the machining operation, while the assessment of the process status is still committed to human expertise. In this chapter the state of the art of monitoring and control of the machining process is presented. The available commercial solutions and the research activities of the last years in this field will be described, providing a comprehensive viewpoint of the motivations behind the present work. The chapter ends with a discussion on the current achievements and future trends.

1.1 MODERN MACHINING

Machining operations are production processes employed to make mechanical parts with high geometrical and dimensional accuracy and surface finish quality. They are used both for production of goods starting from semi-manufactured casting workpieces and for finishing of components obtained by bulk deformation of casting processes. The high accuracy and flexibility made machining operations such as turning, milling and drilling an essential step in manufacturing production. Since the introduction of Numerical Control (NC) machine tools by Parsons Company and MIT in 1952 that subsequently evolved in the Computer Numerical Control (CNC) machine tools (1970_s), the advances in machining practice has always be linked to the advances in automation. Nowadays CNC operations are widely applied not only to metal cutting machine tools but also to non-conventional cutting machines (e. g. Electrical Discharge Machining (EDM) and laser cutting) and generally speaking to all that applications where is needed a motion control system such as industrial robots for assembly and welding, sheet metal cutting, automated measurement (Coordinates Measurement Machines (CMM)) and rapid prototyping. The workflow in modern machining practice consists of three phases:

PLANNING It includes all the offline tasks that lead to the generation of the program needed by the NC to perform the desired working operations.

EXECUTION It includes all the online tasks that allow to obtain the finished product.

EVALUATION Is the final phase which concerns all the control tasks of the working process yield.

In the planning phase Computer Aided Design (CAD) and Computer Aided Manufacturing (CAM) softwares are used to design (or redesign) and ana-

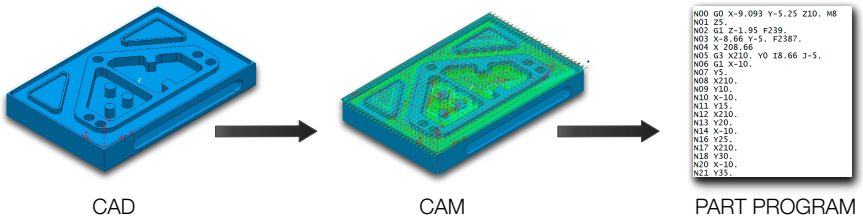


Figure 1: *Process planning operations*

lyze the component, to select the machine tool, the cutting tools, the cutting parameters, the workpiece fixture and finally to generate the tool paths basing on the collected information. The output of the CAM software is the part program containing the whole set of instructions that will be used by the NC to control the machine. The resulting part program is optimized in terms of minimization of the machining time, obtained by reducing the number of passes, rapid positioning and tool changes. Process and working planning is a slow and complex stage whose proper execution relies mainly on engineers experience. During the execution phase the machining process take place. The CNC machine interprets the part program and generates the references for position and velocity control of axes. The instructions for a programmed operation are sent to the servo motors that move the axes in order to obtain the desired path. The NC manages the axes interpolation moving the tool along linear, planar and three-dimensional trajectories and then allows the machining of parts with complex geometry. The axes positions are continuously tracked by sensors that send the measurements back to the NC closing the control loop. While on one hand the CNC machine performs the working operation driving the axes along the commanded paths, on the other hand the control unit can not neither perceive nor evaluate the process status. In other words the NC can reconstruct the actual machine configuration in terms of axes positions, but can not recognize and then compensate unexpected changes in the working conditions. Monitoring of the process execution and intervention in case of undesired situations are assigned to the machine operator that can command the machine tool through a control panel (Man Machine Interface (MMI)). During machining, the role of machine operator is to stand over the working operation in order to keep

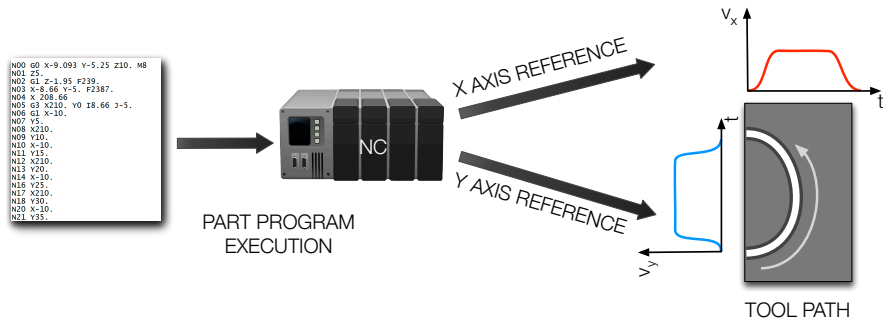


Figure 2: *Process execution operations*

the process under nominal conditions. This task is performed through the tuning of cutting parameters such as spindle speed and axes velocity (feed) and the replacement of the cutting tool according to its wear level. Moreover machine operator has to halt the machine if critical or dangerous conditions occur such as the collision with the workpiece. Then the quality of the resulting machined part depends also on the skill of the machine operator and on his ability on detecting and compensate anomalous working conditions. The evaluation phase involves the measurements of surface roughness and of both dimensional and geometrical accuracy in order to assess the quality of the finished parts and the compliance with the design requirements. This is a post-line task that is performed by skilled personnel both in the shop floor with portable instrumentation and laboratories where parts can be inspected also by means of dedicated NC machines designed for measurement purpose (CMM). The CMM can be programmed to automatically collect a set of data on single features (e. g. circularity of holes) or on the entire part under examination. The resulting measurements are compared with the CAD model in order to check the process yield. If there are pronounced differences with the design requirements the process must be partially (re-machining) or entirely repeated. This is true especially during the setup of the machining process for a new part where several iterations among the three working phases can be needed. From this brief description two main aspects show up: on one hand the planning phase is not a trivial task since the choice of cutting parameters and the optimization of tool paths have a direct effect on costs and working time. On the other hand the contribution of the machine operator during the working phase plays an important role on the quality of the finished workpiece and consequently also on the production costs. This means that the expertise and the skills of engineers and workers are still key factors to obtain a proper execution of the process. In the last thirty years and even more nowadays, the whole manufacturing industry is forced to seek reasonable trade-offs among the increasing request on product quality and production flexibility, on one side, and the constant need of costs saving and productivity improvement, on the other side. According with this trend, the interest of machine tools producers and customers moved towards the reduction of non productive time of machines preventing unexpected production halt, the improvement of parts quality and globally the optimization of the available resources. This resulted in an high push, for both machining industry and research, toward the design of more versatile and flexible machine tools and the improvement of the entire process chain [1, 2]. Especially for the last point this is achievable with the expansion of the planning phase, by considering not only the part geometry and the material characteristics, but also the working process including the modeling of the cutting mechanics. This will result in more accurate process simulations with the capability to generate part programs with tool paths and cutting parameters optimized in each point of the workpiece. Moreover the use of more sophisticated CAM software that includes also the geometric models of the machine tool and of the clamping systems allows to avoid the risk of collision during the working execution. In addition, the enhance-

ment of the machine tool should be achieved making it capable to assess the working status by adding a monitoring layer that can collect and provide information on the actual working conditions, tool wear and breakage, machine health and critical situations (e. g. collision). Finally the integration of the evaluation phase also inside the inline task would allow the machine to quickly compensate working errors, reduce the re-machining time and avoid rejected parts. Improvements of the planning phase aim to the generation of part programs that ensure controllable working processes, understood as stable cutting operations (e. g. without regenerative chatter) with predictable product accuracy and tool wear. The overall purpose is the reduction of setup times and the achievement of cost-effective working since the first batch. Equipping the machine tools with monitoring systems is intended to improve the reliability of the process by the fulfillment of three main tasks:

MONITORING The system measures quantities related to the process status or to the machine conditions and elaborate one or more indicators that are shown in the MMI as aid to the machine operator.

DIAGNOSTIC The system evaluates the health of machine parts (bearings, gears etc.) and in case of failure try to find the source. Together with the monitoring task facilitates the maintenance scheduling and the fault detection.

ADAPTIVE CONTROL The system works in closed loop on the machine. It can automatically tune process parameters according to a given control law. In the simplest case the control stops the machine whether dangerous conditions occurs. In more sophisticated cases the adaptive control can track a desired working condition (cutting force, power, Material Removal Rate (MRR) and so on), optimize the process or maintain the required surface quality.

A monitoring and control system can improve the productive capabilities of the machine with a twofold aim. Firstly providing the needed support to the machine operator with objective assessments of the process and assisting him in the corrective actions regardless his skills. Secondly the increased automation level of the machine tool allows to perform unattended operations contributing to the development of unmanned machining systems. The main areas of development of these systems concern the improvement

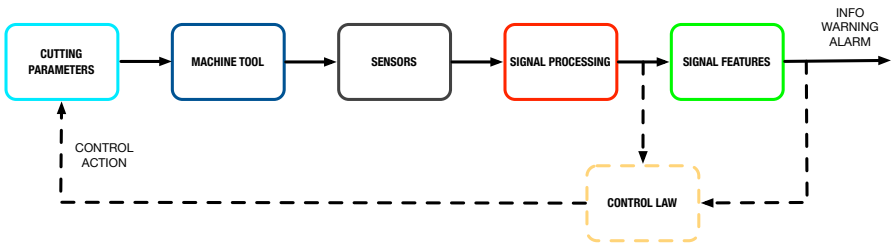


Figure 3: *Process execution operations*

of cutting conditions, vibration mitigation, chatter suppression, tool condition monitoring, workpiece collision, fault detection and diagnosis. The benefits achieved with the use of monitoring systems consist of the overall improvement of machined parts quality, the reduction of machine idle and unexpected halt, the avoidance of rejects and a safer usage of the machine. A generic architecture for a monitoring system is shown in figure 3. The feedforward branch (solid lines) describes the basic elements needed for the monitoring action that consists of measurement of process variables and the processing of acquired signals. The resulting data are then used to provide monitoring and diagnostic information to the machine operator in the form of numerical values, graphs, warnings or alarms. The feedback loop (dashed lines) completes the architecture with the control action that tunes the cutting parameter in order to achieve the desired performance.

1.2 INDUSTRIAL APPLICATIONS

Despite the pressing needs of the machining industry and the big research effort in this field, the advances in the automation level of machine tools and on the whole process are limited, especially with regard to the monitoring and control of cutting operations. In the following sections will be presented the main commercial products used to improve the machining process.

1.2.1 *Planning*

Modern CAM softwares (e. g. [3, 4, 5]) have additional features including an accurate geometric model of the machine. This allows to simulate its kinematics and verify the collision risk between the tool part (tool, tool holder and spindle) and the workpiece part (workpiece, clamp, table). Moreover there are also solutions only intended for simulation [6] that offer the possibility to check the generated part program, avoid collision and optimize the feed rate according to the amount of material that has to be removed. Finally it is worth to mention the softwares developed by the Manufacturing Automation Laboratories [7] that, unlike the majority of commercial solutions, perform the complete simulation of the machining process providing the cutting forces, power and torque, the machine vibrations and the chatter occurrence highlighting the areas of the part program where the process does not fulfill the required working conditions.

1.2.2 *Execution*

The design and development of a commercial monitoring system for machining operation is a challenging task. Beside the choice of the algorithms to detect the typical undesired situation as tool wear, chatter and collisions it is also needed to achieve the compatibility with the available NC. Moreover the monitoring system should be robust enough to perform its tasks regardless the machine architecture, the workpiece and the tool geometry.

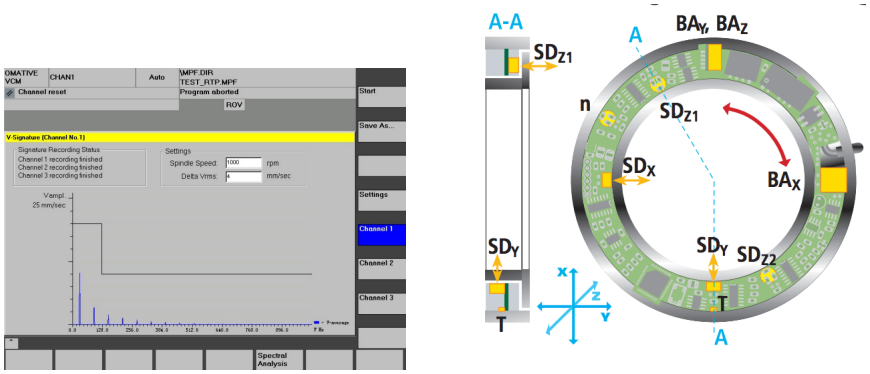


Figure 4: *OMATIVE VCM vibration signature recording (a) (courtesy of OMATIVE Systems) and Prometec 3SA ring spindle sensor (b)(source and copyright: PROMETEC GmbH)*

Because of this issues, nowadays few commercial applications can be found. The most common solutions are monitoring systems with adaptive feed regulation based on measures of spindle load. These systems allow the setup of reference levels for the cutting power that will be tracked by tuning the axis velocity during the working operation. Moreover they can be programmed to detect a worn or broken tool and to halt the machine if overload conditions occurs. Usually the basic architecture consists on the load sensor located on the spindle and a dedicated firmware implemented on the NC that performs the data acquisition and the control tasks. However in some case the NC is not prearranged for monitoring tasks and then additional control unit and sensors are needed. Heidenhain [9] provides a monitoring system with adaptive feed regulation and chatter mitigation functions acting on the axis velocity. The chatter control however is limited to the frequency band of the drive controller. Omative, Caron Engineering, Artis and Nordmann [10, 8, 11, 12] provide monitoring systems for adaptive feed control and tool condition monitoring. These systems tune the feed according to the desired power and rise warnings or halt the machine in case of overload. Omantive [13] developed also a vibration monitoring system capable to halt the machine if anomalous working conditions or collisions are

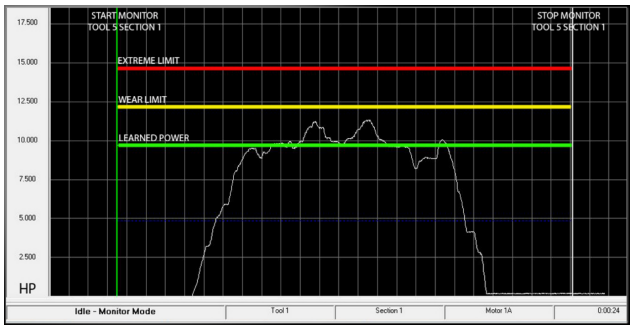


Figure 5: *TMAC [8] adaptive feed control (Courtesy of Caron Engineering, Inc.)*

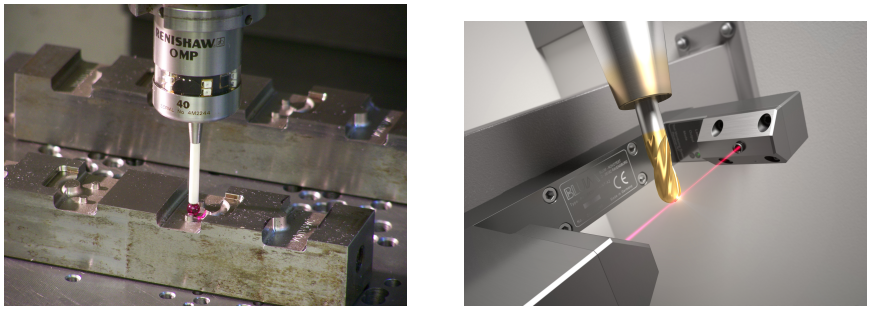
detected. In addition it can perform condition monitoring tasks based on synthesized spectrum analysis (Fig. 4a) and logging tasks for post failure analysis. However, unlike the previous systems an accelerometer and an additional signal processing unit are needed. A different solution to perform monitoring tasks is offered by Prometec [14] that designed a modular system that acquire data both from drive sensors, by communicating with the NC, and from additional sensors as accelerometers, load cells and acoustic emission transducers. This system has an embedded Digital Signal Processing (DSP) unit for signal processing and allows to perform tool wear, tool breakage and collision detection, spindle condition monitoring and unbalance recognition. Moreover they developed an intelligent sensor for spindle diagnostic that embeds inductive sensor to measure shaft displacements, accelerometers and temperature to assess bearing health and spindle overload and finally a tachometer for spindle speed measurement (Fig. 4b). The most common monitoring strategies rely on the comparison between signal features and fixed limits (Fig. 5), however different techniques can be used especially for tool monitoring as dynamic threshold, part signature and pattern recognition related to tool breakage signals [15].

1.2.3 *Evaluation*

Evaluation of the machining process has been partially moved from postline to in-process task thanks to the development of touching probes (e. g. [16]) and CNC measurement cycles that allow to setup the working operation by measuring the position and orientation of the workpiece (work reference system definition) and the tool geometry (tool presetting). Touch probes are used also to perform in-cycle gauging operations for the measurement and compensation of thermal effect, tool wear and part distortion. Moreover dedicated touch probes or contactless sensors are used to assess the integrity of the tool edges. In this way it is possible to control and maintain the desired tolerances. The use of these devices can be programmed in the part program in order to integrate the evaluation and compensation tasks in the current process but can be also executed after the process for the inspection of the machined part.

1.3 RESEARCH STATE OF ART

Academic research in machining has been very prolific over the last fifty years in order to improve the machining process (in terms of quality and productivity) and to increase the automation level of the machine tools. Several engineering fields have been interested in the development of strategies leading to the improvement of machining operation and fully automated machines. In general two main research branches can be distinguished. The first one is much more oriented toward the innovation of machine tool design considering the introduction of new mechatronics devices intended for the improvement of the actuation and control capabilities of the machine.



(a) (b)

Figure 6: *Touch probe (a) (source and copyright: Renishaw plc) and contactless tool setter (b) (source and copyright Blum LMT, Inc.)*

Additional actuators, such as electromagnetic, piezo and hydraulic actuators, are used to increase bandwidth of the traditional feed drives (superimposed axes), to actively modify the stiffness and damping of the machine structure, to control the tool deflection and to keep the process stability. The main goals of these techniques are the increase of positioning accuracy and speed of the feed axes, the compensation of thermal effects on the machine structure and drives, the vibrations damping and the avoidance of chatter during the cutting operations. An extensive description of these techniques can be found in [17]. The second approach aims to make the machine tool capable of optimize the working operation, according to the desired performance, and evaluate the process status by equipping it with monitoring and control systems. The development of an "intelligent machine" can be outlined in the following steps:

PROCESS MODELING This step is needed both to increase the knowledge of the process and to develop the decision making strategy for control purpose.

SENSING Onboard machine sensors together with additional sensors are used to measure the process variables of interest. The most commonly used sensors are accelerometers, dynamometers, Acoustic Emission (AE) sensors and inductive current sensors.

SIGNAL PROCESSING Signal coming from sensors are elaborated by means of amplification, filtering and other common signal processing techniques.

FEATURES EXTRACTION Indicators sensitive to the observed phenomena are derived from signals both in time and in frequency domain.

CONTROL Relying on the selected features the control action is executed in order to keep the process under nominal conditions. Several strategies can be used ranging from limit setting to more sophisticated Artificial Intelligence (AI) techniques.

Because of the complex interaction among the machine, the tool and the workpiece, modeling is needed to get a better insight of the working process. According to [2] the main approaches for cutting modeling are:

- Analytical modeling based on the cutting force computation considering tool geometry, shear angle, chip flow and friction conditions.
- Slip-line modeling based on the use of material models to predict the mechanical response and temperature distribution in the shear zone and in the tool-chip contact zone.
- Finite Element Modeling (FEM) that analyzes a mashed model of the workpiece and tool to determine material strain and stress and the chip flow.
- Mechanistic modeling based on the assumption that the cutting forces result from the product among the uncut chip area, the specific cutting energy of the material and the cutting parameters.

Analytical and mechanistic methodologies are mostly used for simulation and optimization purposes [18]. Several works have been conducted for the estimation of cutting forces, surface roughness and the prediction of cutting stability as well documented in [19]. These models accurately describe the phenomena related to the cutting process and allow to estimate the performance of the machining operation in terms of surface quality, required cutting power and tool life. The results coming from the modeled process are used to identify the critical parts of the machining operation and to optimize it in order to properly select the cutting parameters. In addition, also AI-based modeling techniques are used. The most common techniques are based on Artificial Neural Networks (ANN), Bayesian Networks (BN) and Fuzzy Logic (FL) inference systems in order to predict tool wear, surface quality and to perform machining state diagnosis (e. g. tool breakage) [20]. However also relatively new approaches as Support Vector Machines (SVM) are catching on especially for surface roughness prediction [21]. Despite the AI-based models lack of physical meaning they are often integrated with monitoring systems, since they provide a reliable and simple tool to estimate process variables from several indirect measurements and signal features. However it is worth to point out that these models need a proper training to work effectively. Then it is required a dedicated experimental campaign that must cover a wide range of operative conditions in order to guarantee both good generalization capability and accuracy of the AI-based system. The remaining steps, listed above, form the building blocks of a generic monitoring system. While modeling of cutting process allows to predict its performance, monitoring the machining operation allows to assess the actual operative conditions and to take the required corrective actions. The development of reliable and effective monitoring and control systems represents the key aspect for the increase of the automation level of machine tools and the achievement of unmanned machining operations.

1.3.1 *Monitoring of Machining Operations*

The monitoring task is generally developed through three main phases: the measurement of process variables, the selection of signals features representative of the observed phenomenon and the control of the process. A wide research effort has been spent for the development of each of these fields. The literature state of art of Liang et al. [22] and of Teti et al. [23] provides a comprehensive overview of the monitoring of machining process including the main advances in sensors and sensing techniques, signal processing and control strategies.

Sensing Techniques

Traditionally the sensing techniques are distinguished in direct and indirect measurements. With direct methods the desired information is obtained by the actual value of the measured variable. Indirect methods involve the measurement of one or more variables related to the analyzed phenomenon. The information of interest is then estimated by means of empirical relations or models. For example the tool wear can be assessed directly by measuring the extension of the corollary with a vision system, or indirectly by using a relationship between spindle power and wear level. Directs methods are generally very accurate but difficult to implement in an existing machine and can also impose some limitations in the machining operation (e. g. cutting without lubricant). This make direct methods more suitable for laboratory applications than for workshop usage. Indirects methods are less accurate than the direct ones but on the other hand they are more easy to implement and then more common in practical applications. The process variables that are usually monitored are cutting forces, power and torque, vibrations, temperature and the acoustic emission of the cutting operation. Since the work of Tlusty et al. of 1983 [24], the most used sensors for monitoring and control applications have been dynamometers, accelerometers, current sensors and AE sensors. However also different techniques are used including vision systems, temperature sensors, strain gauges and displacement sensors.

Cutting Forces

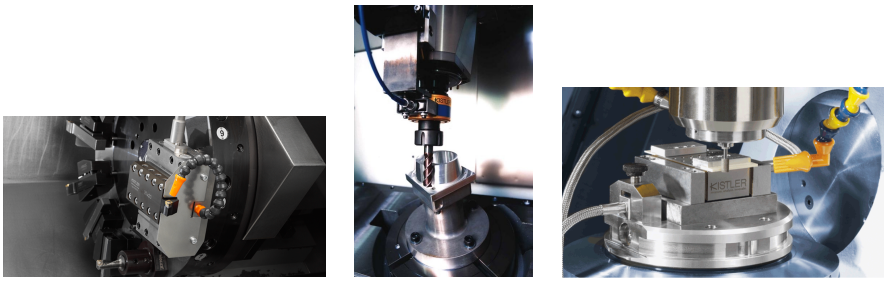
The measurement of machining forces provide the best information on the cutting mechanics that is used to evaluate tool wear, cutting stability and the quality of the machined part. Cutting forces and torque can be measured directly by means of multi-component piezoelectric load cells [25, 26]. There exist commercially available tool holders (Fig. 7a, Fig. 7b) and working tables (Fig. 7c) with embedded force transducers that are widely applied in laboratory tests. However their dimensions reduce the available working area and the lack of overload protection impose some limitations in the machine usage. These restrictions together with their high cost make this solution rarely used in industrial applications. An other approach is the use

of strain gauges-based transducers that measure the deformation of a flexible structure caused by the cutting forces [27]. This solution though suffer of reduced stiffness and limited bandwidth that make them unsuitable for most of the standard applications. For this reason Ma et al. [28] proposed a new method based on a piezoelectric strain rosette mounted on the tool holder that provided good results both for force measurements and chatter detection, compared to more traditional sensors systems. Indirect machining force assessment includes the measure of spindle deflection, by using contactless displacement sensors [29], and power or current consumption measurements from spindle [30] and feed drives [31]. The critical issue in the measure of spindle displacement is given by the calibration of the spindle system, consisting of spindle, tool holder and tool, whose stiffness must be determined accurately in order to estimate the cutting forces. Moreover this method is also affected by displacements caused by motor heat that must be compensated in the measurement. Spindle and feed drive have embedded current sensors for motion control that are used to compensate force and torque disturbances. Therefore cutting forces can monitored by measuring the current variations. Since no extra sensors are needed this solution is easy to implement and very cheap but there are some important limitations that must be considered. Feed drives have to compensate the dynamics of the moving axes then their bandwidth is typically less than 100 Hz. This reduces the quality of sensory information and can make this solution ineffective for some monitoring and control tasks. Moreover, for the most common axis architectures, friction and deflection of ball screw and side guideways have a strong influence on feed drive action. These effects vary with feed velocities and axis position and then an accurate calibration is needed in order to compensate their contribution in current and power variations. The effect of friction is less significative in spindle and cutting torque can be easily obtained from current measurements but this is true only if a motorized spindle is used. In some machines spindle and motor are connected through a gearbox, a linking shaft and sometime a driving belt that have to be accurately modeled in order to calculate the right value of torque. In any case, from the cutting torque is only possible to obtain the forces tangential to the tool making this method less reliable for some applications (e. g. tool wear monitoring). A complete overview of force-based monitoring and control techniques can be found in [32].

Acoustic Emission and Vibrations

AE signals are stress waves generated by the sudden release of energy within a material that propagates through the material itself. Possible causes of stress waves can be high deformations and cracks growth. In machining AE is due to the cutting and chip formation mechanics and the variations of AE signals can be related to tool wear, chip breakage, cutting vibrations. For these reasons AE sensing has been widely used in literature especially for tool condition monitoring (wear and breakage detection) but also for chatter detection and surface integrity of the machined part [33]. The high

bandwidth of AE sensors ranging from $10\,kHz$ to $1\,MHz$ can detect most of the phenomena related with machining operations and make them insensitive to machine vibration and environmental noise but require signal processing and an accurate calibration according with the cutting conditions. In addition AE signals are sensitive to process parameter and sensor location and then the choice of the proper sensor position and calibration can become a non trivial task. Due to these issues AE-based techniques are considered much more reliable for monitoring when are used in multi-sensors approach and despite AE sensors are also inexpensive and not invasive for the working space they are hardly employed in workshop applications.
Vibrations in machining are an important indicator of the working conditions and then a good variable for monitoring applications. Cutting vibrations can be distinguished in forced and self excited vibrations. In the first case vibrations depend on process conditions such as interrupted cutting or tool engagement but also on non homogeneous material properties and tool wear. Forced vibrations are stable and their magnitude can be reduced acting on feed velocity since they are strictly related with cutting forces. Self excited vibrations, also known as regenerative chatter, are an unstable process condition due to waviness of the machined surface that causes a varying chip thickness. Under certain conditions, related with rotational speed and depth of cut, the chip thickness may grow very quickly increasing the cutting forces and the vibrations level. Chatter is a particularly dangerous working condition that produces poor surface quality, reduces tool life and can result in rejected parts. The most used sensors for vibration measurements are piezoelectric accelerometers since they have relatively high dynamic range, high full scale range (suitable for collision detection applications), and are insensitive to constant accelerations. The state of research about the phenomenon of chatter and the techniques to detect and avoid it has been reviewed by Quintana et al. in 2011 [34].



(a) (b) (c)

Figure 7: *Dynamometer applications for turning 7a and milling 7b,7c (source and copyright of Kistler Holding)*

Machine Vision

The use of vision systems is a relatively new approach in monitoring of machining operations mainly focused on tool condition monitoring [35]. The wear level can be assessed both in a direct manner, by analyzing the image of the tool edge, and indirectly through the examination of the machined surface. Vision-based techniques represent an inexpensive way to perform contactless direct measurement of tool wear and surface finish and thanks to the wide amount of available image processing algorithms it is possible to develop real time monitoring application. The reliability of this method though is affected by the adopted illumination system and working conditions (chip, lubricant and dirt). Lighting type and level is one of the most critical aspect for the acquisition and processing of images since low or non-uniform illumination does not allow to draw the needed information from the image. Diffuse lighting systems like LED lights and fiber optic guided light seems to be the best solution to improve the image quality for inline applications. Vision systems require in addition a proper image processing strategy that allows to obtain robust and significative indicators of the observed phenomenon. Numerous studies in literature proved that vision-based sensing techniques can be useful for monitoring tasks but further research activity is needed to increase their robustness and reliability for real time applications in the severe conditions of machining environment.

Signal Processing and Features Extraction

Signal processing includes all those strategies that are performed before extracting the desired signal features from the acquired data. This task is needed to obtain a clean signal from most of the disturbance sources existing in industrial applications. The basic approach consists of signal amplification, low pass filtering and A/D conversion that can be performed by dedicated device and nowadays by sensor itself (digital sensors). Further processing can be the digital filtering of signal in order to extract information in the frequency band of interest and the signal segmentation.
Features extraction is performed both in time and frequency domain and most common signal features are:

TIME DOMAIN Amplitude, mean, Root Mean Square (RMS), variance or standard deviation, skewness, kurtosis, peak and peak to peak amplitude, crest factor and synchronous averaging.

FREQUENCY DOMAIN Fast Fourier Transform (FFT) and Power Spectral Density (PSD) transformation, dominant peaks amplitude, energy in frequency bands, skewness and kurtosis.[1]

Other signal features are derived from those listed above by using for example ratio of two extracted features. The analysis in the frequency domain by means of Fourier transformation is affected by the averaging of the signal frequency content in the period of observation. For this reason a

1 see [23, 20] for references

time-frequency analysis is often performed applying the Short Time Fourier Transform (STFT) that computes the FFT inside a time sliding window of n samples. In this way it is possible to capture the variations of frequency components over time. This approach though suffers of an important limitation: an high time resolution obtained with the choice of a small window will result in a poor frequency resolution. Wavelet analysis overcomes this drawback providing a time-frequency signal transformation by using windows of different length for the low frequency and high frequency content. Discrete Wavelet Transform (DWT) decomposes the signal in approximation (the low frequency part) and detail (the high frequency part) by convolution between the signal and the impulse response of a low-pass and an high-pass filter. This operation is iterated in a multi-level decomposition where the subsequent signal approximation is further processed in order to extract lower resolution components. A second approach is given by Wavelet Packet Transformation where both approximations and details are decomposed in order to increase the number of frequency bands. In both cases signal components obtained by wavelet analysis are generally used as separated signals from which extract the desired features.

Control Action

Control of machining process is traditionally classified as:

ADAPTIVE CONTROL WITH CONSTRAINTS (ACC) Systems tune process parameters in real time to keep the monitored variable (e. g. power or cutting forces) to a constant value.

ADAPTIVE CONTROL WITH OPTIMIZATION (ACO) Systems select process parameters to fulfill a specific objective as for example maximize tool life or minimize working time.

GEOMETRIC ADAPTIVE CONTROL (GAC) Systems are much more oriented to maximize the finishing operations yield by controlling tool deflections and wear.

Adaptive control systems have been reviewed in 2004 by Liang et al. [22] and since then among the recent works it can be cited the study of Quin et al. [36] that proposed a discrete sliding mode control algorithm to perform robust control of cutting forces in milling. Simulated case studies proved the effectiveness of the proposed method. Rubio et al. [37] developed a model following adaptive controller based on fractional order hold discretization method. In this approach the discretization generates several reference model running in parallel and the control law is provided by the best performing model chosen at each sample time by a switching algorithm. Zhang et al. [38] proposed a control scheme in which a ANN is used to tune the parameter of a PID controller according to the cutting force. Even in this two cases the reported results are referred to simulation of the milling process. Other preliminary studies on control strategies were [39, 40, 41]. Yao et al. [42] have recently developed a PID controller with static parameters

assisted by a AI-based controller (FL or ANN controller). In this case the PID controller is used to maintain the process stability while the AI-based controller is intended to improve the system response to process nonlinearities. Experimental tests demonstrated that both controllers successfully tuned the feed in order to keep the cutting force around the reference value. In addition also the advantages of online training of the ANN controller were provided. Zuperl et al. [43, 44] used a ANN-based adaptive controller in conjunction with an off-line process optimizer. First the recommended process parameters are obtained by the off-line optimization of the working process and during the cutting operation the adaptive controller is used to prevent the cutting forces to overcome a given level while keeping the maintaining the desired cutting performance. The experimental results performed on a irregular workpiece geometry provided an improvement of 27 % of machining efficiency compared to a conventional milling operation. Saikumar and Shunmugam [45] developed a feed rate adaption controller suitable for both roughing and finishing operations. In addition they provided also a test workpiece geometry to be used for testing purposes of different control strategies. Despite the good results obtained, this method has been applied only to $AISI4340$ steel and the controller strategy has been developed following a specific experimental campaign. Yang et al. [46] proposed an adaptive surface quality control based on two neuro-fuzzy systems. The first one predicts the surface roughness relying on cutting parameters and cutting forces measured with a plate dynamometer. According to the estimated value the second one corrects the feed rate in order to control the quality of the machined part. Tang et al. [47] proposed a control methodology for regulation of forces, position errors, and contour error in machining base on a hierarchical architecture with two levels: the machining forces and contour error one and the servomechanism one. The goals of constant cutting forces and zero contour error are propagated to the bottom level where are joined with the tracking goal of the motion controller. An optimal control law is then formulated to match the desired goals. Simulated turning operations provided promising results for the application of this approach. Rubio et al. [48] proposed an expert rule based system for cutting parameters selection in the milling processes. A set of rules is used to build the Pareto optimal front that provides the optimal cutting conditions. Bosetti et al. [49] developed an additional controller to increase the automation level of the machine tools that implements a supervision and optimization loop. The controller collects process state information from the CNC and from dedicated sensors performing the control action on feed and spindle speed overrides. By means of a simplified process model and of a optimal control algorithm the proper cutting parameters are selected in real time on the basis of a weighted multi-objective target function. Moreover a vibration prediction module is included so that the occurrence of chatter can be estimated in the pre-processing phase and the spindle speed can be varied according to the cutting stability. Off-line process optimization for a $21\,s$ simulated tool path has been carried out in $350\,ms$ demonstrating the capabilities of the proposed approach for in-line optimization. Finally Denkena

et al. [50] developed a cutting force control in a prototype CNC machine equipped with a contactless magnetic guided spindle slides. This actuator is completely independent from the machine axes an is actuated to compensate the cutting forces by varying the depth of cut. The proposed approach is intended for finishing operations in which the surface roughness and the geometrical accuracy are much more important than the machining time.

1.3.2 *Monitoring Objectives*

Tool Condition Monitoring

Tool life is one of the most important issues in machining practice. Tool wear modifies the geometry of the cutters negatively affecting the geometric accuracy and the surface finishing of the workpiece. Moreover tool wear and tool breakage are responsible of up to the 20 % of process halt [22] resulting in unpredictable downtimes that reduce the machine productivity and consequently rise the overall working costs. The latest comprehensive review of Tool Condition Monitoring (TCM) techniques [51] highlight the developments observed in research for turning, drilling, end milling and face milling. This study shows that usually monitored variables are:

- Cutting forces and torques measured directly with dynamometers or estimated through spindle current.
- Vibrations measured with accelerometers.
- Acoustic emission.

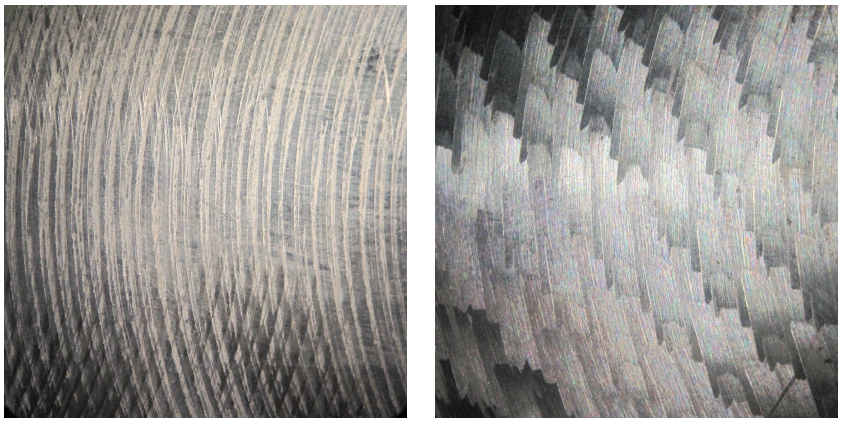
Tool wear assessment is performed by using both time and frequency domain methods, statistic indices and AI-based approaches mainly oriented towards ANN and FL models. Wavelets are also used to detect wear growth and tool breakage but, as authors noticed, their potential has not been fully exploited. Moreover other mixed time and frequency domain methods (e. g. STFT) have been seldom used, despite their effectiveness for the analysis of transient signals has been demonstrated. This work also shows that there not exists monitoring techniques explicitly developed (or effective) for two or more working operations such as milling and drilling. This is quite odd since modern CNC machines can execute several machining operations and the monitoring system should be capable to properly detect wear in each condition. A recent review of flank wear prediction methods for turning [52] confirms the trends discussed. In particular force, vibrations and AE measurements are recommended for in-line TCM applications while frequency domain and time-frequency domain signal processing are considered much more advantageous than other techniques. In addition the authors emphasize the need of developing "universal systems" since most of the monitoring application reviewed are application specific and their effectiveness depends also on the material and on the working parameters. Vision-based techniques proved to be a promising approach for TCM [35]. Vision systems can be used for direct measurement of the worn area allowing

with a single measure the detection of different wear mechanism (e. g. flank wear and chipping) and the tool breakage. In addition tool wear can be inferred also from surface roughness measurements. The application of these techniques in industrial environment for real time monitoring require further research especially aimed to the development of robust systems and for a proper integration with the machine tool. However the direct measurement of land wear extension could be easily implemented for in-process evaluation of the tool wear in actual machining environment (see for example [53]). Concerning TCM in milling Soichi and Shimizu [54] developed an adaptive control for tool life regulation based on force measurement and on the assumption that cutting forces slowly increase with the growth of the tool wear. Force measurement are performed at previously defined tool path checkpoints at intervals of several meters, selected in those region where the process is in steady state conditions. The force trend is then used to predict the residual tool life and feed is adjusted in order to track the target tool life. Cutting force was measured with four displacement sensors mounted on a spindle collar but it can be also used current measurement of feed and spindle drives that makes this method easy to implement in common CNC machines. Experimental results demonstrated the effectiveness of this approach even if is not suitable to detect abrupt tool damages as chipping or breakage. Girardin et al. [55] chose to monitor the variation of spindle rotational frequency to assess tool wear by using the rotational encoder of the spindle. Authors noticed that differences in rotational frequency slowdown are good indicators for differential wear of the cutting edges (wear of each edge with respect to the others). This indicators proved to be also more sensitive to tool wear and breakage than cutting forces since speed variation correlates to the mechanical work performed by each tooth. However this approach was tested only in straight milling and with a single edge cutting at a time that is a simple working condition then further research work should be done to verify its reliability for other operative cases. Pejryd et al. [56] followed a similar approach using spindle and axis encoder signals to monitor tool wear. Several signal features were extracted (e. g. range, RMS, variance) and compared with those obtained by a new tool through the sum of squared distance statistic. The repeatability of the proposed method was tested and was found to be strong for short-term measurement while got weaker for long-term measurement probably because of changes in machine tool condition. The authors highlighted the potentiality of using encoder signals for monitoring purposes but stated also that further investigations are needed. Ritou et al. [57] proposed a monitoring method of tool wear by estimating the radial eccentricity of the teeth related to the cutting force with a mechanical model of the milling process in the angular domain. Cutting forces were measured with two eddy current displacement sensors mounted on the spindle nose and angular domain signal processing techniques as synchronous averaging were performed. Tool wear is evaluated by means of control charts used to monitor the cutting forces for each tooth. Wang et al. [58] used an incremental learning network (distributed Gaussian ARTMAP) to classify tool status. Cutting forces measured with a

table dynamometer were used to extract several signal features (RMS ratio, variance, kurtosis, skewness...) in order to train the network. Experimental results demonstrated that the proposed approach succeeded to detect the right tool wear level outperforming an other online learning classifier. Doukas et al. [59] made a preliminary study to find the relationship among tool wear, vibration level and spindle torque evaluated through spindle current measurements. As a result the authors stated that RMS value of spindle current could be used to asses tool wear. Prickett et al. [60] developed a microcontroller-based TCM system mainly intended for tool breakage detection. The system process the spindle power signal in frequency domain to extract the energy of the cutting harmonics and also in time domain to measure the energy variation related to tool rotation. Finally a set of rules based on threshold levels of the monitored indicators allows to rise a warning both in case of excessive wear or tool failure. Ai et al. [61] related tool wear to the cepstrum coefficients of the acoustic spectrum of the milling operation. Cho et al. [62] used a multi-sensor approach to develop a TCM system. Several signal features were extracted from force, vibration, AE, and spindle current measurements and subsequently selected using different approaches. Finally wear level was assessed by using three different AI-based classifiers. The aim of this work was to investigate the most accurate set of sensors - features - classifier for TCM purposes. From the experiments it was found that the highest accuracy can be achieved by using force, vibration, and AE sensors together with correlation-based feature selection method and majority voting machine ensemble. Moreover SVM classifier outperform the ANN-based classifiers. Tobon-Mejia et al [63] proposed a data-driven prognostic approach that uses raw data coming from a monitoring system to predict the evolution of cutting tool wear. The prediction model was built by using a Dynamic Bayesian Network that was able to assess the remaining useful life of the tool in the experimental tests.

Vibrations and Chatter

Metal cutting process usually involves the presence of vibrations. In milling forced vibrations are caused by the cutting edges entering and exiting from the workpiece but generally speaking the interactions among the machine, the tool and the workpiece are a source of vibrations. Moreover, mechanical imbalances, cutters eccentricity (due for example to tool wear), gearboxes and so on are responsible of forced vibrations. Although vibrations are always present during the machining process an high level is an unwanted condition that can reduce the working performance, decreasing the tool life and worsening the surface finish. The most detrimental situation is represented by the onset of regenerative chatter consisting of self excited vibrations that continuously grow, leading the process to instability. Chatter worsens the workpiece surface (Fig. 8b), reducing the geometric accuracy, the cutting edges, causing tool chipping or even breakage, and also can damage the machine tool. For these reasons chatter has been considered with high interest in academic and industrial research. Altintas and Weck



(a) (b)

Figure 8: *Texture of a milled surface under normal cutting condition (8a) where feed lines are visible and with chatter (8b) where the surface is marked by the tool vibrations.*

[64] reviewed the main chatter modeling approaches for turning, milling, drilling and grinding while Quintana and Ciurana [34] in 2011 reviewed the state of research distinguishing among off-line and in-line methods to prevent or suppress chatter. Finally Siddhpura and Paurobally recently reviewed the chatter research in turning operation. Chatter onset is due to a self-excitation mechanism resulting from the waviness of the chip thickness. Vibrations caused by the cutting process make the cutting edge to oscillate producing a wavy surface that is encountered at the next revolution in turning or by the next tooth in milling and drilling operations. This generates a new wavy surface that can make the chip thickness to exponentially grow with the consequent increase of cutting forces. Stability of cutting conditions is related to dynamic properties of the machine, tool and workpiece and to the cutting parameters. The most common way to evaluate the process stability is to use the Stability Lobe Diagram (SLD) that allows to draw the border between stable and unstable cut as a function of spindle speed and depth of cut (Fig. 9). SLD are generally obtained, starting from the Frequency Response Function (FRF) of the tool - tool holder - machine set, according to the following steps [65]:

1. Select a chatter frequency from a natural mode of the FRF.
2. Calculate the phase shift between the current vibration mark and the previous one (inner and outer modulation).
3. Calculate the critical depth of cut that defines the stability border.
4. Calculate the corresponding spindle speed for each stability lobe.
5. Repeat the operation for each natural frequency.

The FRF of structure is computed from the tapping test of the tool tip usually performed with an impact hammer and an accelerometer with standstill

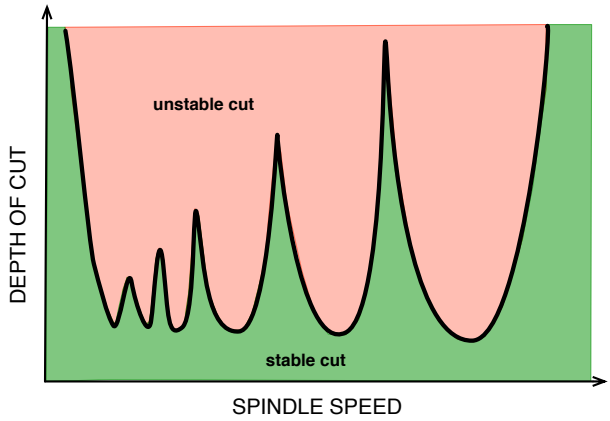


Figure 9: *Stability lobe diagram*

spindle. Knowing the stability region allow to select the cutting parameters that maximize the MRR and then the productivity of the machine in stable conditions. The SLD is specific for the machined material, the working conditions (e. g. the width of cut in milling) and workpiece - tool pair. Then it should be computed every time one of this elements varies resulting costly and difficult to perform by machine end-users since it requires time, high knowledge level on system dynamics and machining process and skilled personnel to perform the experimental modal analysis. For these reasons the use of SLD-based techniques is not very widespread in workshops except for particular applications where the cost of components makes mandatory the absence of rejects (e. g. machining of aeronautic parts). Nevertheless researchers are still strongly interested to develop strategies for the identification of SLD and the suppression of chatter at its onset. The main issue related to the hammer test is that the FRF of the structure, obtained in static conditions, does not take into account of the speed-dependent effects of the spindle dynamics that can modify the stability region shifting the lobes. This require a more accurate modeling of the system or an experimental validation of the computed SLD. Cao et al. [66] modeled with a finite element method the dynamics of an high speed spindle and investigated the speed effects on spindle shaft and bearings. Authors found that the gyroscopic moment of the spindle shaft does not affect the direct FRF of the tool tip while amplify the cross FRFs. Moreover the centrifugal forces acting on the shaft and the bearings lower the stiffness of the spindle as speed increases. The SLD computed taking into account the speed effects revealed that stability lobes shift significantly towards low speed. A different approach was proposed by Abele and Fiedler used the Numerical Subspace State Space System Identification technique (N4SID algorithm) to identify the state space model of the spindle during milling. The identified model provided a more accurate SLD compared to that one obtained in static conditions. Other authors used Operational Modal Analysis (OMA) techniques to extract modal parameters of the system during machining. Zaghbani and Songmene [67]

chose two different OMA techniques to obtain the SLD. Li et al. [68] proposed a method to generate random excitation during milling to be used with two OMA identification methods. In both cases the SLD obtained from machining data resulted much more accurate in stability region prediction. Because of the uncertainties in stability prediction described above, the in-line chatter detection and suppression is a key task in monitoring systems. In this case SLD identification is not needed and chatter occurrence is detected by analyzing signals features obtained from vibration, force, AE, etc. measurements. Liu et al. [69] used the energy and kurtosis of feed drive current signals as indicators for chatter in turning. A SVM was trained using current and acceleration signal as input-output pairs resulting in a 95% of accuracy on chatter detection. Jia et al. [70] developed a chatter detection strategy based on standard deviation and one-step autocorrelation function. Lamraoui et al. [71] defined two chatter indicators based on instantaneous spindle speed measurement processed in angular domain. The first indicator takes into account the periodic contribution of spindle speed and it is high when cut is in stable conditions. The second one is related to the aperiodic (random) part of the signal and is higher when chatter occurs. These indicators derived from cyclostationarity analysis of the spindle speed signal developed in [72]. This method was effective on detecting chatter and resulted also easy to implement in existing milling machines. Abele et al. [73] proposed a signal processing technique for a chatter monitoring system implemented on a PLC. The algorithm proved to be capable to detect chatter condition even with the limited sampling frequency of the PLC, resulting a promising method for the development of an inexpensive chatter monitoring system. Tangjitsitcharoen and Pongsathornwiwat [74] developed a chatter monitoring algorithm based on dynamometric measurement. Ratios of average variance of forces and experimentally selected threshold were used to evaluate the occurrence of chatter. Tangjitsitcharoen et al. [75] improved the system by using the same algorithm to scan the wavelet decomposition of the force signals. In this way they increased the robustness of the algorithm for different cutting conditions. Kim et al. [76] developed an embedded controller for chatter detection that computes the RMS value of the band pass filtered vibration signal. If the value overcome a specific threshold the controller compensate the cutting parameters by using a look up table and communicates them to the NC. The performance of the proposed system was obtained my measuring the surface roughness with and without compensation. The average improvement given by the controller was about the 25 %. Albertelli et al. [77] investigated the effects of spindle speed variation in turning on chatter mitigation and on process yield. Authors noticed that spindle speed variations presents high vibration mitigation properties and does not affect the surface finishing. Van Dijk et al. [78] developed two control strategies to avoid chatter, the first one set the tooth pass frequency equal to the dominant chatter frequency while the second one mitigates the chatter using spindle speed adaption. In both cases chatter free milling operation were achieved in experimental tests. Kakinuma et al. [79] proposed a disturbance observer to assess the spindle torque disturbances introduced by the cutting

process. Using a set of digital filters allowed to extract information on chatter occurrence from the observed disturbance. The results of this work were used by Yoneoka et al. [80] to develop a control system that adapted the cutting parameters if chatter onset was detected on disturbance torque signal. Morita et al. [81] monitored the vibrations in machining and measured the actual chatter frequency at its onset. This is then used to correct the natural frequency of the system and to find a stable spindle speed closer to the current one. The new value is finally sent back to the NC to suppress the chatter vibrations.

Machined Part Quality

Surface roughness and geometric accuracy are the variables used to evaluate the process yield. Wrong process parameters selection or anomalous working conditions due to high vibrations, chatter or excessive tool wear, affect the surface finishing and the geometry of the machined part increasing the risk of rejects or requiring re-machining operations. In-line monitoring of part quality is an important task that allows to adapt process parameters in order to avoid machining defects. Denkena et al. [82] developed a model-based monitoring system that assesses surface topography according to force measurements. The dynamic behavior of the tool - tool holder - machine set was modeled in order to calculate the tool tip displacements. The proposed method proved to be accurate under moderate cutting conditions while did not provide reliable information in case of more severe working operation. However even if information on surface roughness were not precise it was still possible to assess the occurrence of chatter. Costes and Moreau [83] measured the tool movement during cutting by using two laser displacement sensors. Height variations due to tool deflection were taken into account in order to improve the estimation of the surface topography. AI-based monitoring system are quite common for surface roughness assessment. Quintana et al. [84] used signals available int the NC kernel (drives current, axes positions...) and R_a measurements to train a neural network. This approach provided an high accuracy of average surface roughness prediction, comparable with the output of a ANN trained with accelerometer signals. This work was further developed by Brecher et al. [85] that implemented the monitoring system on a machine tool. The machine without the need of external sensors was capable to evaluate the R_a value during the machining operation providing optimized cutting parameters to the operator. García et al. [86] investigated the use of force, vibration and AE measurements to assess R_a value in turning. From experimental results, cutting forces proved to be more sensitive for R_a estimation while the remaining signals did not seem to influence that parameter. Finally a polynomial regression model and a ANN were used to R_a prediction providing comparable outcome. A similar approach was used by Asiltü et al. [87] that trained an ANN using force and roughness measurement for surface finishing prediction. Vrabel et al. [88] used an ANN-approach to evaluate surface roughness in drilling. One neural network was used to es-

timate tool wear by using the cutting force, the feed and cutting velocity as input. The predicted flank wear is then used as an an additional input for a second neural network to assess the surface finishing. Abdul-Ameer et al. [89] developed a vision-based sensing device mounted in a CNC machine intended for surface roughness measurements.

Machine Condition Monitoring

Condition-based and predictive maintenance are strategies that allow to plan maintenance actions in advance relying on the actual condition of the machinery. This reduces the risk of machine fault and then of unexpected downtimes and, on the other side avoids premature maintenance actions. In both cases, operating costs are reduced and machine productivity is increased. Prognostic Health Management (PHM) involves the use of monitoring system to detect machinery parts failure in advance in order to plan prompt maintenance activity, only focused those parts that are actually damaged. Lee et al. ([90],[91]) provided an overview on PHM and discussed the development of maintenance approaches and the future trends in modern manufacturing systems. Jiao and Meiling [92] applied a fault diagnosis strategy to a NC machine tool. The authors divided the machine tool in functional blocks (e. g. spindle, feed drives, CNC and so on) and developed an object-oriented approach to define a set of rules to assess machine part fault. The system uses both rule based reasoning and case based reasoning mechanism to find the actual situation of the machine. Finally a ANN uses the processed information to provide diagnosis on detected fault. Bediaga et al. [93] used two accelerometers mounted in a spindle head and AI-based techniques to develop an automatic fault detection and diagnostic system of spindle bearings. The extracted signal features were related to a method used in bearing condition monitoring (Basis Pursuit Decomposition) and were used to build a FL and a ANN classifiers. The FL-based system resulted more suitable for the application and was implemented in the machine. Bisu et al. [94] proposed a method for vibration analysis based on synchronous envelope analysis in order to characterize both the machine tool state and the milling tool wear. Authors adapted envelope analysis, that is generally used for condition monitoring of bearings and gearboxes, to the machining field and implemented a monitoring system with two accelerometers and a speed sensor. Neugebauer et al. [95] described a design strategy for a condition monitoring system of machine tool focused on spindle unit. Spindle bearings were monitored acquiring accelerometric, AE and temperature signals. Temperature trend, envelope analysis of vibration and an AE indicator were used to assess the spindle health. Experiments performed on a test bench allowed to prove the effectiveness of the proposed method. The AE based indicator resulted the more sensitive to bearing damage allowing to detect a fault at the 50% of the bearing life, followed by the envelope analysis (20% of the estimated life). Temperature measurement was effective only at the end life of the bearing showing a temperature growth in the last 800 *hours* before the predicted substitution. Moreover authors described the

monitoring action needed to assess the status of the tool clamping unit and the rotary union. Liao et al. [96] proposed a reconfigurable prognostic platform designed to be easily integrated in several machine tools. Time domain, frequency domain and mixed time-frequency domain (e. g. wavelet) analysis algorithms were used to extract signal features while several AI-based classifier were implemented as health assessment and diagnostic tools. In order to validate this system an automatic tool change application and the spindle bearing condition monitoring were used as case studies with successful results.

1.4 DISCUSSION ON FUTURE TRENDS

The developments and the achievements in research field to improve the machining operation aim to the increase of the machine tools automation level that involves the monitoring, control and optimization of the working process. Despite the many successful experimental results, there is a limited transfer towards industrial applications or commercial products. The reasons that often arise as the main causes of the lack of development to practical implementation can be grouped in:

COST Many of the adopted approaches are based on expensive sensors systems (e. g. piezoelectric dynamometers) that offer the best sensing capabilities but their use result convenient only in few industrial fields.

COMPLEXITY The advanced techniques proposed to develop an effective monitoring and control system are perceived as tricky to use and in addition, because of the wide diversity of machining operations, it is often needed a fine tuning of the system to work correctly. This require at least a basic knowledge of the implemented methods and then a properly trained personnel.

However, advances in machining operations will rely mainly on the development of smart machine tools or machining systems. The design and implementation of reliable and flexible monitoring and control systems is then a key point to achieve this goal. These systems should be completely autonomous in the evaluation of the working conditions with self-tuning capabilities and the possibility of learn from unexpected or new machining situations. Moreover they should provide simple and clear process information that can assist the machine operator regardless his experience. Nowadays, thanks to the needs on other fields such as automotive robotics and Information and Communication Technology (ICT), the offer of small-sized, cheap sensors is continuously increasing together with the availability of powerful microcontroller units. This facilitates the development of programmable sensors system that can be fully integrated in the machine tool architecture. Moreover trends in latest research works show that there is a constant effort in developing monitoring systems based on signals already available on the machine tool as the drive and spindle current or coming from the actual sensory equipment of the machine. This, combined with innovative signal processing approaches (e. g. angular domain signal processing), AI-based

decision making models and advanced control techniques, seems to be a promising way towards the requirements of modern machining technology. It addition to research activities, the development and deployment of process monitoring and control systems requires also an effort by NC producers that should design a standardized interface in order to provide an easy access to axis and spindle control loop signals and allow the communication between the monitoring system and the CNC resulting in a better integration with the machine tool.

The present thesis fits in this context and summarizes the work done in the development of a distributed system, based on microcontrollers, designed for monitoring tasks of milling process. In the following chapters will be discussed the motivations and the aim of this work, then the hardware - software architecture and the implemented algorithms will be described. Finally will be proposed an advanced control strategy and its application on linear motion system calibration and chatter suppression will be discussed.

Part II

DISTRIBUTED MONITORING SYSTEM

2

MONITORING SYSTEM ARCHITECTURE

This chapter describes the architecture of the distributed monitoring system. Firstly the motivations and the aim of the developed system will be discussed. Then the hardware architecture will be described focusing on the adopted microcontrollers and sensors. Finally the software architecture both of the main system and of the single tasks will be presented in order to give an insight of the implemented monitoring and control strategies.

2.1 MOTIVATION

The orientation towards increased productivity, cost reduction and high flexibility of manufacturing plants requires an enhanced automation level of modern machine tools. An improved process control is required in order to increase the effective productive time of the machine, by reducing unexpected downtimes, and to guarantee an high quality of the machined parts, limiting re-machining operations and avoiding rejects. Consistently with this view, the prevention of faults and the proper planning of machine maintenance is a needed condition for an efficient production line. In this context, the automated supervision of machining process is a strong need in manufacturing. Equipping the machine tools with monitoring systems allows to support the machine operator in the process control, and guarantees the machine productivity and the part quality even under critical working conditions. In addition an effective monitoring and control system is a key element for minimally attended or unmanned machining. This work arises from these needs with the aim to develop a monitoring system for machine tools, capable to assess the working status, identify the source of anomalous situations (e. g. excessive vibrations) and take corrective actions to restore nominal working conditions, avoiding damages to both the machine tool and the workpiece. The proposed system is mainly intended to supervise the machining process and assist the machining operator through the tuning of process parameters and by providing warning and alarm messages on the MMI. The design of the system involved the choice of sensors and the definition of the architecture, needed to collect information on the process, the development of monitoring algorithms and the interface with the NC of the machine. This work has been partially developed within a project funded by PAMA s.p.a., a machine tool manufacturer that aims to equip its machines with a monitoring system in order to improve the machining performances and the safety during their usage. According to these needs the monitoring purpose has been mainly focused on the reduction of cutting vibration, on Condition Monitoring (CM) of spindle bearings for predictive maintenance applications and on emergency halt of the machine in case of collision.

Figure 10: *Moving column milling machine with extended boring bar (courtesy of PAMA S.p.A.)*

2.2 REQUIREMENTS

The monitoring system is intended to equip large CNC machine tools with moving column or gantry architecture, retractable ram and boring bar. This machines are used for conventional machining including face and side milling, drilling and boring both in roughing and finishing operations. In the first case large depth of cut and tool engagement are used in order to achieve high MRR. The cutting forces, and consequently the vibrations involved, are high affecting the tool life and the machined part quality. In the second case the MRR is less relevant but there exist strict limitations on surface finish requiring a careful choice of cutting parameters. Moreover on these machine can be mounted several accessory heads that are mainly used to modify the spindle axis orientation (Fig. 11) in order to continue the working operation without rotating the workpiece. Therefore during the process the machine can considerably change its configuration and vary the working conditions. For this reason the monitoring system must be capable to adapt to different situations avoiding loss of performance and should be fully integrated with the machine tool architecture. The system is required to perform three tasks: the monitoring and mitigation of cutting vibrations, the detection of machine collision, and the identification and diagnosis of damaged spindle

Figure 11: *Accessory indexable head with milling axis rotated of* 90° *(courtesy of PAMA S.p.A.)*

bearings. Vibration level is affected by several factors dependent on cutting parameters, process stability, tool wear and on the workpiece geometry (interrupt cutting conditions) that makes it an important indicator of process status. Containing vibrations helps to avoid severe working conditions that result detrimental for tool life, surface finishing and in some cases also for the machine tool. Collision detection is needed to promptly halt the machine in order to avoid or contain serious damages. CM tasks such as spindle bearing health assessment allows a more accurate planning of the machine maintenance (condition-based or predictive maintenance) avoiding downtimes due to faults or vice versa unnecessary costly intervention. Finally the monitoring system must be easy to use for the machine operator that can access to high level information on the process while more detailed data can be made available to engineers for more detailed analysis. According to this description the system requirements can be summed up as follows:

TASKS Cutting Vibration Monitoring (VM) and mitigation of critical vibratory effects. Collision detection and immediate halt of the machine. CM of spindle bearing with detection of the damaged elements.

ROBUSTNESS Process monitoring must work properly in different working operations and with different configuration of the machine.

FLEXIBILITY The system must be embedded on the machine tool regardless its architecture and the adopted NC. Moreover it can be possible to extend its functions if new monitoring tasks are needed.

EASE OF USE General process information must be provided to the machine operator in order to help him in the evaluation of the ongoing working operation. More detailed information for example on events occurred during process and bearing status can be accessed for further offline analysis.

COST AND MAINTAINABILITY The system should be inexpensive and easy to maintain, update or extend.

2.3 ARCHITECTURE

The design requirements described in the previous section led to the definition of a microcontroller-based architecture used to create a set of measurement nodes, distributed on the machine, that make up the actual monitoring system. Information on both the machine and the working status are collected and sent to the NC through an other microcontroller that act as a bridge between the machine and the monitoring device (Fig. 12). Nodes are classified as *monitor* and *collector*. *Monitor* nodes interface with sensors, acquire data and process them in order to extract the desired signal features and develop the necessary corrective actions. *Monitors* are located on the machine or accessory heads close to the measurement point. *Collector* is a single node located on the machine that manages the information flow between monitors and NC. The *collector*-NC communication is performed

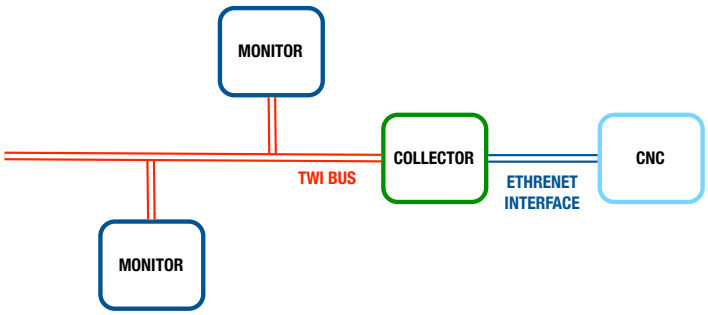


Figure 12: *Monitoring system basic architecture*

through an ethernet interface with an UDP protocol. *Monitors-Collector* communicate through a TWI bus that allows the connection of several devices in master-slave mode by using two bidirectional serial lines, one for the clock reference (SCL) and one for the data signals (SDA) as shown in Fig. 13. The communication bus has been organized with a multimaster configuration in which *monitor* and *collector* nodes are both masters and slaves. In normal monitoring operations *monitors* are defined as master and the *collector* as slave. In this way each *monitor* node can communicate the information related to its specific task when ready and the *collector* waits for the incoming data. In cases where *collector* has to communicate with a monitor (e. g. sending a setup parameter), it becomes a master and the *monitor* behaves as a slave. The bus natively support multimaster configuration handling communication priorities with an arbitration process. Each slave is identified by an univocal address and for *monitor* nodes this is communicated to the *collector* when they are connected to the bus. This allows the plug of a new node even with the system running (hot plug) and it is particularly suitable for the connection of accessory heads during the working operation. The choice of sensors was driven by the embeddability and economic requirements and for these reasons accelerometric data are collected both for the vibration mitigation and the CM tasks. Sensors used for this application are triaxial MEMS digital accelerometers with a configurable full scale range up to $16\,g$ and a sampling frequency up to $5\,kHz$. MEMS sensors are small sized ($< 10\,mm^2$), inexpensive ($\sim 1\,€$) and do not need signal conditioning so that no additional units are required and are much more suitable for the integration on the machine structure.

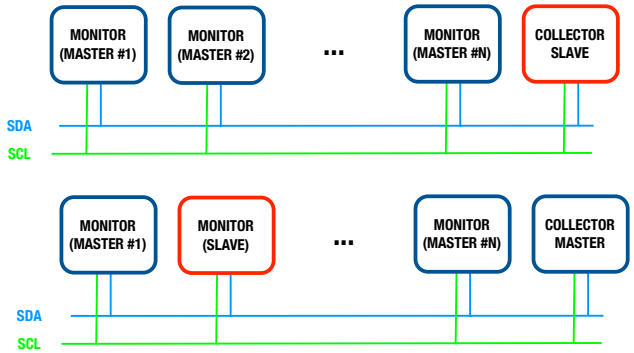


Figure 13: *Multimaster configuration of the* TWI *bus*

2.3.1 *Hardware*

Monitor nodes have been developed on 8 *bit Atmel* AVR microcontrollers (Tab. 1) since they support a large number of peripheral interfaces and have a good resolution of the ADC, making them suitable for signals acquisition and data communication. Collector node has been implemented on a more powerful ARM microcontroller (Tab. 2) with ethernet interface and a embedded Linux operating system installed. Despite it has been used mainly for bridging tasks between the monitoring system and the machine NC, that could seem simplistic for this kind of hardware, the choice was related to future development of the system. This might include additional functionalities as an user interface and a remote access through a web server. The adopted accelerometers are LIS3DH MEMS transducers from STMicroelectronics (Tab. 3) since their bandwidth and Full Scale Range (FSR) was the most suitable for this application.

Hardware specification

MONITOR NODE	
CPU	
FAMILY	ATxmega128A1
ARCHITECTURE	8 *bit* AVR
FREQUENCY	32 *MHz*
MEMORY	
FLASH	128 *KB*
SRAM	8 *KB*
EEPROM	2 *KB*
EBI	max 16 MB
DEVICES	
I/O PINS	78
UART PORTS	8
SPI PORTS	12
TWI PORTS	4
ADC CHANNELS	16@12 *bit*
DAC CHANNELS	4@12 *bit*

Table 1: *Monitor specifications*

COLLECTOR NODE	
CPU	
FAMILY	AT91SAM9G20
ARCHITECTURE	32 *bit* ARM9
FREQUENCY	400 *MHz*
MEMORY	
ROM	64 *KB*
SRAM	32 *KB*
EBI	
DEVICES	
I/O PINS	96
UART PORTS	7
SPI PORTS	2
TWI PORTS	1
USB PORTS	3
ETHERNET	1
ADC CHANNELS	4 @ 10 *bit*

Table 2: *Collector specifications*

SENSORS	ST-LIS3DH ACCELEROMETER
SPECIFICATIONS	
RANGE	$\pm 2 - 4 - 8 - 16$ *g*
BANDWITH	2.5 *kHz*
RESOLUTION	$8 - 16$ *bit*
INTERFACE	
TWI	$100 - 400$ *kHz*
SPI	max 10 *MHz*

Table 3: *Accelerometer specifications*

Hardware Configuration

The monitoring system hardware architecture is shown in figure 14. On the machine side there is the *collector* node communicating through the ethernet connection with the NC (Siemens PCU50) and listening on the TWI buffer for incoming data. Close to the spindle is located the first *monitor* node (MASTER #0) acquiring signals from one or more accelerometers located on the spindle bearings. Finally the accessory head is equipped with the second *monitor* node (MASTER #1) that acquire the accelerometric data from sensors

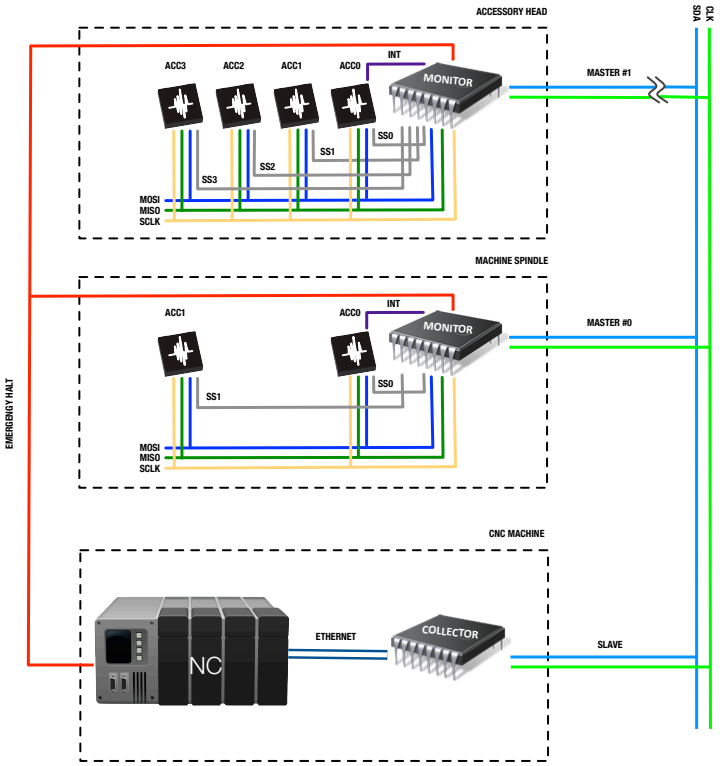


Figure 14: *Hardware architecture*

on the head bearings. During VM task each *monitor* is designed to acquire up to 4 accelerometers signals sampled at $5\,kHz$. Sensors are connected with *monitors* through an high speed Serial Peripheral Interface (SPI) with a $10\,Mbps$ data rate. The bus is intended for master-slave communications and consists of the clock line (SCLK) and two data transfer lines (MISO and MOSI). In addition SPI bus need also a SS line for each device connected in order to communicate with only a slave at a time. The *monitor* node is defined as master, it provides the clock reference and commands the SS lines. Accelerometers are the slave devices that respond to master requests. However the acquisition is regulated by the first accelerometer (ACC0) exploiting a function of the LIS3DH transducer that triggers an interrupt each time that a new data is available. Then a fifth digital line is needed by one sensor only for the interrupt driven acquisition. Finally every *monitor* node is connected with the CNC through a single wire digital line used to halt the machine, avoiding the slower communication on the TWI bus and allowing a prompt response of the system in case of emergency. *Monitors* equipping accessory heads have further monitoring requirements and then they require additional hardware (Fig. 15). A logging system has been implemented in order to provide information to technical support on events occurred to the accessory head. This function is requested to be active even when the accessory head is detached from the machine. Therefore a backup battery is needed for the power supply *monitor* that is kept in a low power consumption mode. The event logger records on an external storage memory the head connection and disconnection from the machine and the exceeding of vibrations

limits. In addition if events such as a collision occurs when the accessory is not operative the *monitor* wakes from its power down state and acquire data from the accelerometers storing them in the internal memory.

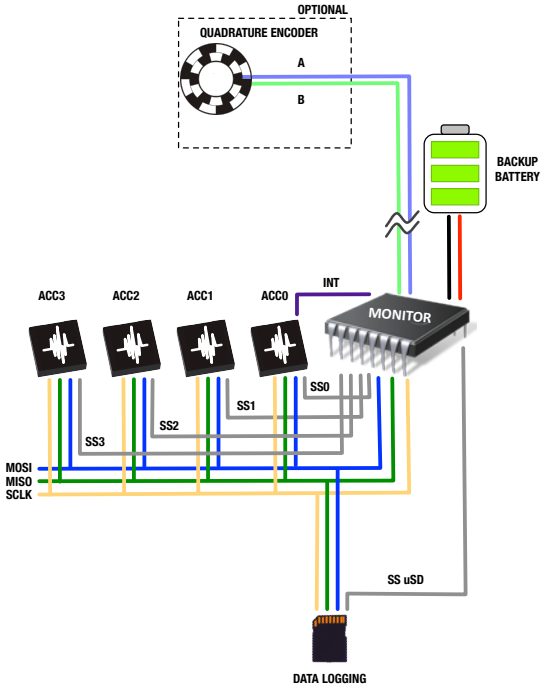


Figure 15: *Extended hardware architecture for accessory heads*

2.3.2 *Software*

Main Architecture

The management and the execution of the monitoring tasks depends on several factors as the working conditions, the machine status and the operator inputs. It is necessary that some tasks are configured, enabled or disabled by the machine tool user according to the specific case, while it is mandatory that other tasks (such as the collision monitoring) are automatically activated and can not be accessed from the outside. For these reasons the main software framework has been organized as a state machine that consists of the following states (Fig. 16):

IDLE Idle state closes all the monitoring activities and the communication with *monitors*. On the other hand The communication services between the *collector* node and the machine are kept enabled in order to receive external commands for state transition.

INIT Init is a transitory state where sensors and the monitoring tasks are configured. It is executed when the monitoring system is started or after a reset. When the parameters are set up the system automatically moves to the inactive state.

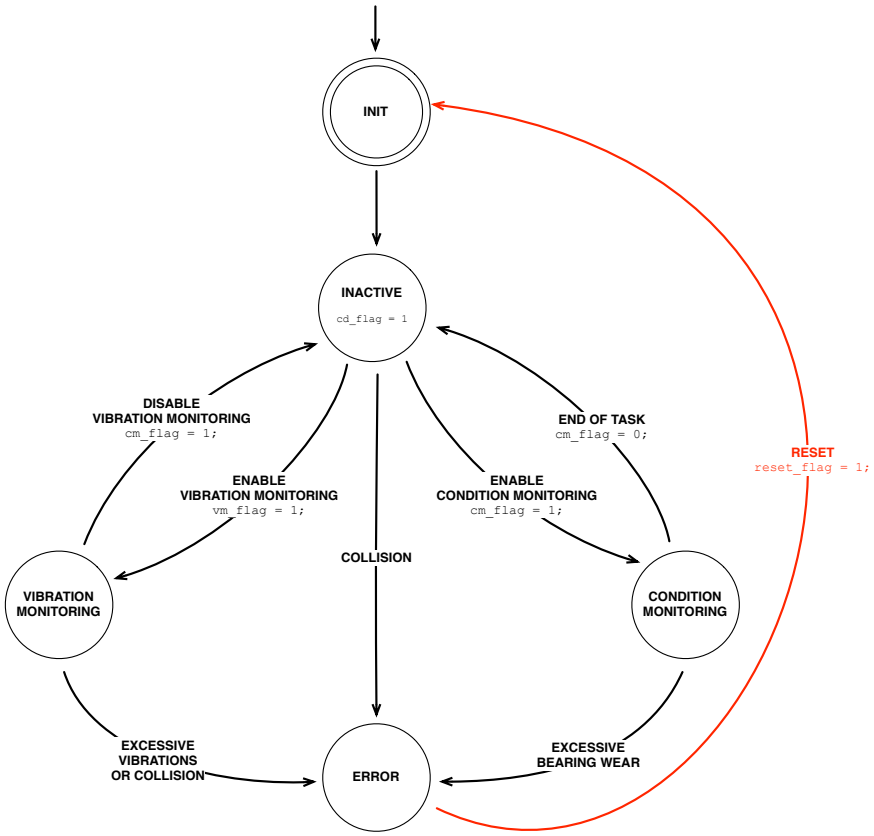


Figure 16: *State machine architecture*

INACTIVE Inactive state means that the system is ready and waiting for the activation of a monitoring task by the machine operator. However all the tasks considered relevant for the machine tool safety are enabled in this state (e. g. Collision Detection (CD)).

VIBRATION MONITORING VM activity can be activated during the machining operation to reduce the drawbacks of high vibratory effects.

CONDITION MONITORING CM is periodically run by the machine operator to assess the health of spindle bearing.

ERROR When a collision or an other potentially dangerous condition occurs the system halts the machine, triggers an alarm and moves in the error state waiting for an external command.

When the machine tool is started the system manager activates the state machine that initializes the *monitor* nodes for the execution of the programmed functions. Sensors parameters are stored in structures dedicated to each task

```
1  /*VIBRATION MONITORING INITIALIZATION*/
   void VM_init()
   {
           /*FILL THE LIS3DH PARAMETERS FOR EACH CONNECTED SENSOR*/
           for (uint8_t i = 0;i<N_ACC;i++)
6          {
                   /*ENABLE X Y Z AXES*/
                   vm_device_params[i].axes = xyz_enable_bm;
                   /*ENABLE 5 kHz ACQUISITION*/
                   vm_device_params[i].frequency = low_5000Hz_bm;
11                 /*SET FS TO 16g*/
                   vm_device_params[i].fsr = _16g_bm;
                   /*HIGH RESOLUTION OUTPUT*/
                   vm_device_params[i].high_res = 1;
                   /*SET DATA READY INTERRUPT PIN*/
16                 vm_device_params[i].intpin = int1_drdy_bm;
                   /*ENABLE BLOCK DATA UPDATE*/
                   vm_device_params[i].bdu = 1;
           }

21         /*SET THE THRESHOLDS*/
           vm_lowLim = VM_LIMIT_LOW;
           vm_highLim = VM_LIMIT_HIGH;
           /*RESET THE VM FLAG*/
           vm_flag = 0;
26 }
```

Listing 1: *Initialization procedure example code*

and the configuration parameters of the monitoring algorithms are loaded from the *monitor* EEPROM or from the NC. As an example, listing 1 describes the initialization procedure of the VM mode. At line 5 the connected accelerometers are configured to work at $5\,kHz$ and with a FSR of $16\,g$ then starting from line 22 the default vibration limits are loaded and the VM flag is reset (`vm_flag = 0`). After the initialization procedure the system enables the CD function and waits for an input from the machine operator that has to select the monitoring task. If a collision occurs in this phase the system changes the logic level of the emergency communication line and the NC halts the machine generating an alarm signal. The system, being in the error state, waits for its recovery. At this point no other actions are allowed except than the system reset. The remaining tasks are commanded by the machine operator basing on the required monitoring actions. VM provides some interaction with the machine tool user that can re-program its intervention, by using an implemented learning function, and can disabled it when is no more needed. The VM state can be also automatically exited for safety reasons, if a collision occur or too high vibration level is encountered, moving in the error state. CM activation is performed on demand and

can not be stopped until the analysis is finished. During CM the CD task is disabled since the accelerometer configuration is different. However this is not a safety issue for the machine because it can not be moved during the analysis. The CM task does not require any interaction by the user and it is autonomously executed till the end. Normally the CM task return to the inactive state when finishes both with a negative or a positive damage detection. In the last case only a warning is sent to the user but, if a severe damage or a too much extended wear is assessed, the system block the machine and switches to the error state. Obviously the system can be reset and the machine can be operated again but a maintenance intervention is immediately required. The monitoring functions have been organized with a common structure that is executed when the desired state is entered. This structure is organized as follows:

INIT Initialization procedure run one time at the startup of the *monitor* node or after a reset as previously described.

SETUP Sensor and acquisition are configured according to the parameters stored during the *init* phase. It is run every time the monitoring function is called.

PROCESS SAMPLE This phase contains all the activities needed for the acquisition of the signals. Generally it involves the sampling of data and their storage in the microcontroller memory for further analysis.

GET RESULT The desired signal features are obtained in this phase. It can be performed both offline at the end of acquisition period (e. g. CM) or inline during the acquisition (e. g. VM).

SHUTDOWN The monitoring task is closed and the exiting action is performed according to the obtained results.

Vibration Monitoring

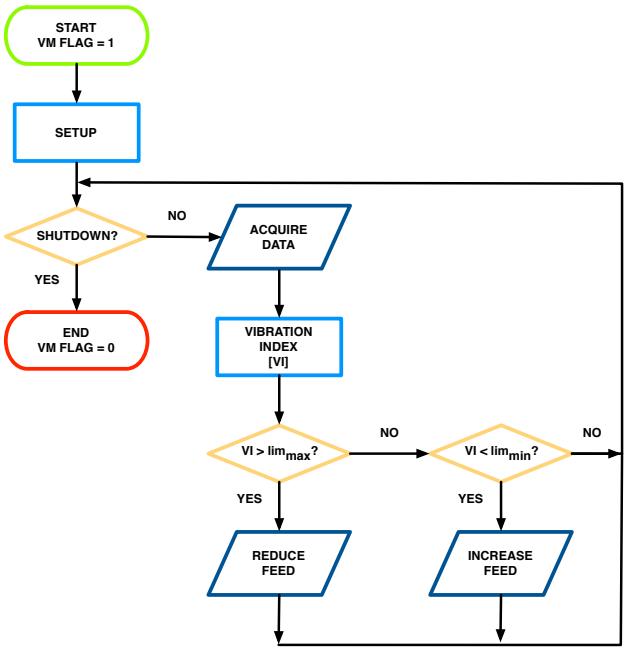


Figure 17: VM *task flowchart*

When the VM task is enabled (Fig. 17) the setup procedure configures the accelerometers with the parameters previously defined in the initialization phase, checking if all the sensors are properly connected to the *monitor*. Then a short acquisition to measure the data offset is started and the offset value is stored for each axis. Finally the flags and counters used by the monitoring algorithm are set. At this point the signals acquisition and the data processing are performed until the machine operator stops the monitoring task. The computed signal feature [1] is compared with two vibration thresholds. If the vibration level exceeds the upper limit the *monitor* commands a reduction of the feed axis, while, if the level is below the lower limit, the *monitor* requires to increase the feed. When the vibration level falls within the two threshold, the working operation is considered safe and no corrective actions are needed. In addition a learning functionality has been implemented in order to set the vibration limits according to the actual working condition. If the machine operator notices that the controller action is not adequate, he can modify the intervention thresholds of the system, adapting them to the machining conditions that considers as optimal. In this case the operator has to tune the process parameters in order to reach the desired working conditions (as usually does), then he enables the VM system in learning mode and the *monitor* starts to record the vibration index. After a certain time interval (e. g. $1\,s$) the averaged value of the recorded data is used to set the vibration limits. The monitoring task then starts without further actions of the machine operator.

1 See section 3.1.5 for a description of the VM algorithm

Spindle Bearings Condition Monitoring

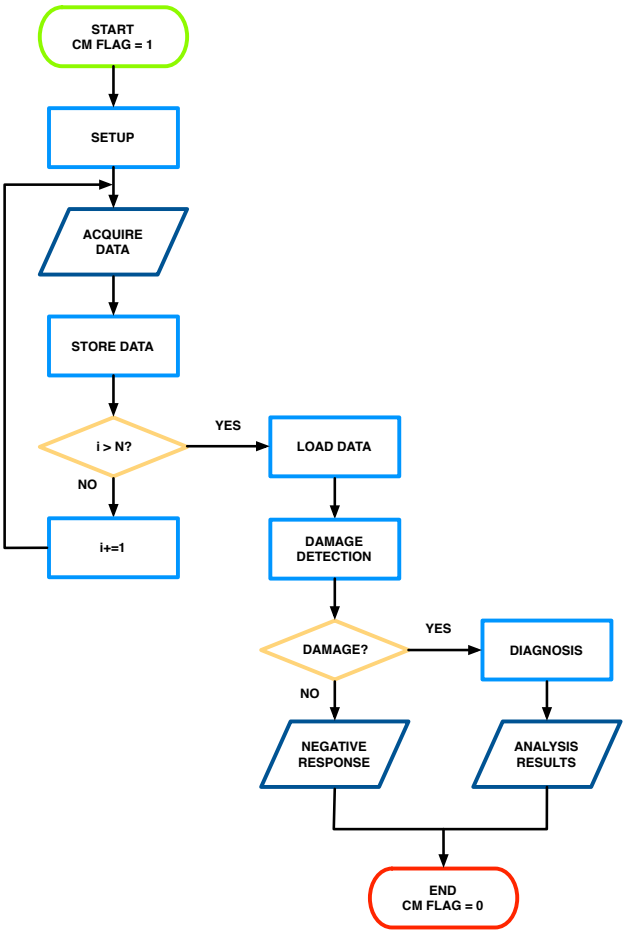


Figure 18: CM *task flowchart*

The function of CM of the spindle bearings is divided in three phases: signal acquisition, damage detection and damage diagnosis. As for the VM task, at the beginning the setup procedure is performed in order to configure the accelerometers. The FSR is set to $2\,g$ in order to have a better resolution since for this task the signal to noise ratio is low. After the configuration the actual sampling frequency is computed by running the accelerometers and by timing the acquisition of a fixed number of data. This is needed because the CM signal features are extracted in the frequency domain and an accurate value of the sampling frequency is needed. When the setup is completed the system acquires the vibrations from each sensor for $3\,s$, storing the data in the external RAM. The acquired signals are then analyzed with a mixed time-frequency domain signal processing method [2] that detects the existence of fault on the examined bearing. If the analysis provides a positive response, the diagnosis of the bearing is executed in order to define the location and severity of damage. The results of the analysis are sent to the

2 See section 3.2.3 for explanation

NC that publishes them in the log file of the machine and on the front panel to warn the machine operator.

Collision Detection

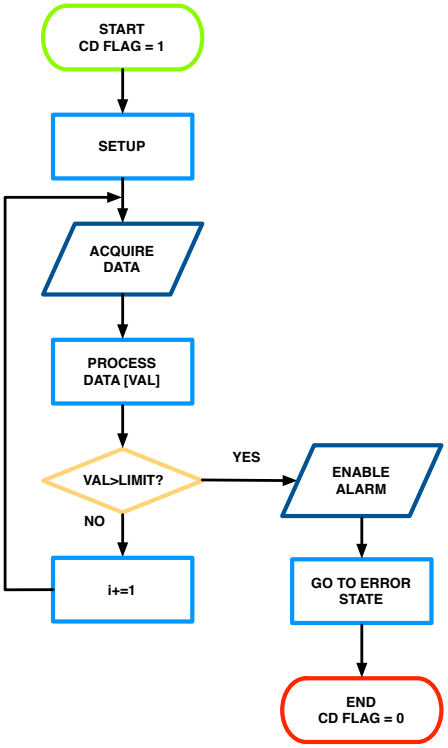


Figure 19: CD *task flowchart*

Collision detection task is enabled when the monitoring system is activated. The CD function does not communicate with the *collector* node but uses a digital line to halt the machine as soon as a impact is detected. This ensures the fastest reaction of the monitoring system in particularly dangerous conditions. When the monitoring system enters in the inactive state, the CD function is enabled. The setup phase configures the accelerometers and then initiate the acquisition. The acceleration signals are processed and the returned value is compared with a threshold at the maximum sample rate. When this limit is exceeded the *monitor* immediately change the level of the digital line halting the machine. The CD routine continues by sending a message to the MMI through the *collector*. Then the system switches to the error state waiting for the reset procedure commanded by the machine operator.

3

MONITORING ALGORITHMS

This section describes the algorithms implemented in the distributed monitoring system. For each task will be provided the motivation and the purposes of the monitoring activity. The algorithms will be analyzed in their main elements. Finally the result of the experimental test will be provided in order to demonstrate the validity of the proposed approach.

3.1 VIBRATION MONITORING

Cutting vibrations are the most common disturbance that affect the process yield. Vibrations are normally present in machining operations caused by discontinuous working conditions (especially in milling) and by the sensitivity to the cutting forces of the set composed by the machine tool, tool holder, cutting tool and workpiece. Machining vibrations are generally classified as:

FREE Free vibrations happens whenever the system, displaced from its equilibrium position, is made to freely vibrate.

FORCED Forced vibrations are caused by periodic or variable external forces generated by the engagement of the cutting edges with the workpiece. Wrong cutting parameters, workpiece geometry and tool wear affect the magnitude of forced vibrations. Moreover also faults on the machine can introduce oscillating or impulsive forces whose effects have an impact on the cutting operation.

SELF EXCITED Because of vibrations the cutting tool leaves a wavy surface on the workpiece at each pass causing a varying chip thickness between two consecutive passes. In certain condition the process can become unstable with an exponentially growing chip thickness and an increasing level of vibrations.

Free vibrations can occur for example when the deflected tool is exiting the workpiece. The cutting forces causing the deflection go quickly to zero and then the tool retrieves its original shape. This kind of vibrations usually have low magnitude and run out after a short transient. Forced and chatter vibrations, on the other hand, can be particularly detrimental for the machining process. A severe vibratory effect produces a poor surface finish, increases the tool wear rate and it is harmful for the machine tool elements. Moreover in the worst case the workpiece can be irreparably damaged. Cutting vibrations are the result of a lack of stiffness in the machining system (e. g. the use of a slender tool) and can be limited with a proper selection of cutting parameters and working setup as tool choice and workpiece clamping. However variable working conditions during the process can lead to vibration increase and also to the occurrence of chatter. The mitigation of

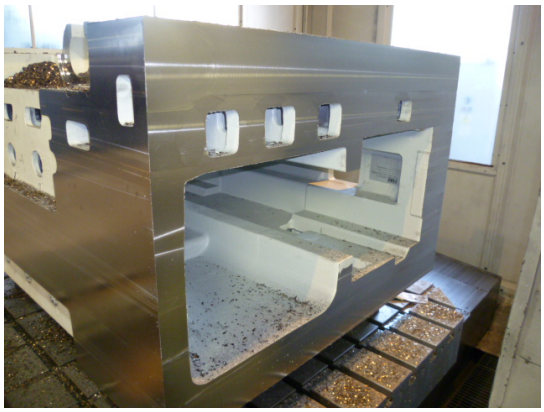


Figure 20: *Workpiece used in the acquisition of vibration signals (courtesy of PAMA S.p.A.)*

undesired vibratory effects is assigned to the machine operator that using the overrides controls on the front panel can tune the axes feed and the spindle speed in order to restore the expected process conditions. The effectiveness of the operator intervention is related to its own experience and skill and therefore his action may be not adequate. The monitoring of cutting vibrations and the detection of growing or excessive vibration level is then an important task to be performed in order to control the process execution and support the machine operator. The development of the vibration monitoring activity has been organized in four phases starting from accelerometric measurements collected during machining operation. From the acceleration signals it can be derived two important parameters: the magnitude of the vibratory effect, and the resulting resonant and harmonic frequencies. Both of these features have been taken into account in order to detect a reliable indicator of the vibrational state of the machine with the aim of performing an inline identification of critical machining conditions. The proposed features have been compared by using the measurements of several roughing operations and, at the end, the most suitable has been chosen for the VM task. The experimental validation of the proposed system was performed on a moving column boring-milling center. Vibrations magnitude was sensed with a triaxial piezoelectric accelerometer placed under the spindle headstock. This position was chosen as close as possible to the milling tool in order to collect data strictly related to the working vibrations. The accelerometer was a DYTRAN 3213M6 characterized by a full scale range of $\pm 50\, g$ and a bandwidth of $1.5 - 5000\, Hz$. With the sake of a better insight of the cutting process, both speed and power consumption of the spindle were also collected. Tests were performed during face milling operations on workpieces with holes, pockets and variable width of cut (Fig. 20), in order to verify the proposed monitoring system in the most demanding scenario in terms of cutter stress. Then a supervisory system based on the selected vibration index has been designed and experimentally tested in order to verify the proposed approach in a field test. Finally the designed

monitoring algorithm has been implemented on the microcontroller architecture of the *monitor* nodes.

3.1.1 *Time Domain Signal Feature*

Root Mean Square

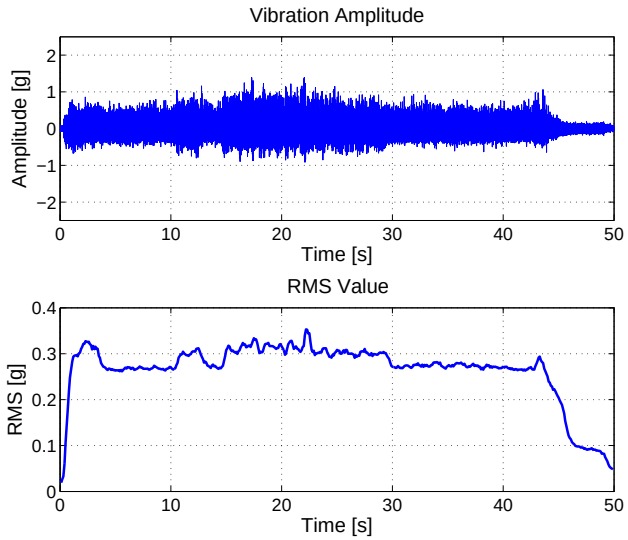


Figure 21: *Vibration signal and its RMS-based index*

Time domain analysis of signals for monitoring purposes is generally intended to measure the amplitude, variability and trend of the measured data. The most common signal features include, magnitude, mean, variance, RMS and power of signal. In some cases as the assessment of impulsive behavior of the signal also specific statistic features as skewness and kurtosis can be computed. In this work the RMS (1) value has been selected as vibration index since it provides more information than the amplitude or mean value, being related to the power content of the signal.

$$s_{rms} = \sqrt{\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} s(\tau)^2 \, d\tau} \tag{1}$$

The implementation in discrete time is straightforward and for the triaxial accelerometer data the vibration index has been computed as the resultant of RMS values of each axis according to equation 2:

$$a_{rms} = \sqrt{a_{rms,x}^2 + a_{rms,y}^2 + a_{rms,z}^2} \tag{2}$$

where $a_{rms,j}$ is the RMS value of the vibrations on j axis computed inside a moving window of N samples.

$$a_{rms,j} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} a_{i,j}^2} \tag{3}$$

Figure 21 shows the measured acceleration and the RMS based vibration index for a face milling operation where the tool engagement has a small variation, causing a slight increase of the cutting vibrations that is noticed by the RMS index.

3.1.2 *Frequency Domain Signal Features*

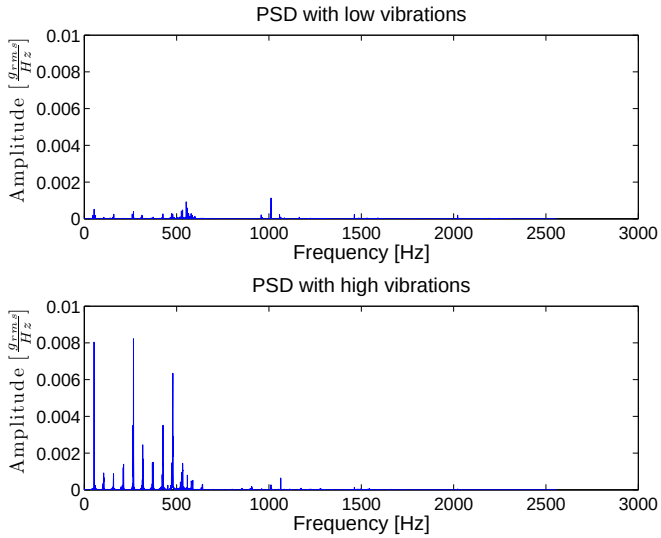


Figure 22: *Spectra differences with low and high vibration respectively. Cutting harmonics are amplified because of the increased vibration level.*

Vibrations measured during the cutting process are directly related to the machining conditions. In fact the variation of cutting parameters, due to the workpiece geometry (depth or width of cut), or to the machine operator action (feed and spindle speed), causes the variation of cutting forces and then of the resulting vibration level. In the spectral signature of milling operations, the vibration signal consists usually of the cutting harmonics, defined by the tooth passing frequency, and the structural resonances of the machine, tool and tool holder. If the process is stable the cutting forces affect mainly the cutting harmonics. However if the cutting conditions lead to process instability also the structural frequencies are involved in the vibration increase. In particular chatter occurrence is usually detected when one of the structural resonances is excited by the regenerative effect. Figure 22 shows how the cutting harmonics are modified by an high vibration level due to an interrupt cutting condition. The analysis of vibration signals in the frequency domain is then particularly useful since the contribution of each component of the forcing input can be easily isolated and studied. The proposed signal features, extracted from frequency domain analysis of measured signals, are intended to quantify the vibration level in terms of the signal power. In other words, the proposed vibration indexes aim to measure how much signal power is used to excite the harmonics related

to the cutting dynamics. For this reason both signal features have been extracted from the PSD of the vibration signals estimated with the periodogram method (4).

$$\hat{P}(f) = \frac{2}{N\, f_s} \left| \sum_{n=0}^{N-1} s[n]\; e^{-i\frac{2\pi f n}{f_s}} \right|^2 \tag{4}$$

The term $\sum_{n=0}^{N-1} s[n]\; e^{-i\frac{2\pi f n}{f_s}}$ is the Discrete Fourier Transform (DFT) of the sampled signal s(t) and f_s is the sampling frequency. Considering the single side spectrum the above equation can be written as:

$$\hat{P}(k\, df) = \frac{2}{fs\; N} |S[k]|^2 \quad \text{with } k = 0, 1, ... \frac{N}{2} - 1 \tag{5}$$

Equations 4 and 5 have been normalized by $\frac{2}{f_s\; N}$ so that the area under the PSD is equal to the total power of the signal.

Average Power Content

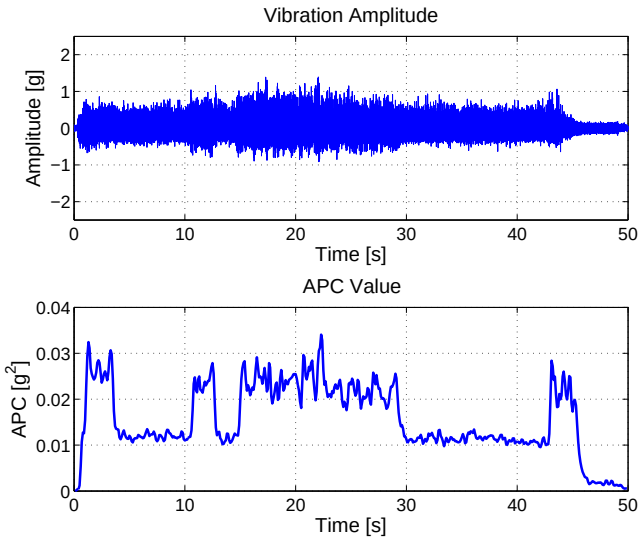


Figure 23: *Vibration signal and its APC-based index*

Given the PSD, the power content of the signal in each frequency bin is given by $\hat{P}(f)$. The Average Power Content (APC) within a frequency band is obtained by:

$$\bar{P} = \int_{f_1}^{f_2} \hat{P}(f) df \tag{6}$$

Equation (6) can be written in discrete time as:

$$\bar{P} = df \sum_{k=k_1}^{k_2} \hat{P}(k\, df) \tag{7}$$

The APC has been obtained according to equation (8) starting from the resultant of the PSDs computed for the accelerations of each axis. In this way the effect of vibrations is globally considered regardless the cutting direction.

$$APC = \int_{f_1}^{f_2} \sqrt{\hat{P}_x(f)^2 + \hat{P}_y(f)^2 + \hat{P}_z(f)^2} df \tag{8}$$

It is worth to notice that the resultant PSD can not be obtained from the resultant of the vibration measurements, since the signal squaring introduces distortions due to the signal rectification. Limiting the APC in the low frequency region, for example choosing the frequency range below ten times the tooth passing frequency, allows to extract the power related to the cutting harmonics. In this context the APC value refers to the power dissipation due to the tool vibrations. Therefore anomalous cutting condition causing an excessive vibration level can be detected. Figure 23 shows the measured acceleration and the corresponding APC value in the same working conditions described in section 3.1.1. The APC is computed within a moving window whose size and overlap (or update rate) can be freely selected. For signal of figure 23 the APC has been computed in a $0.5\,s$ window considering the power content of the $0 - 500\,Hz$ frequency range.

Peaks Count

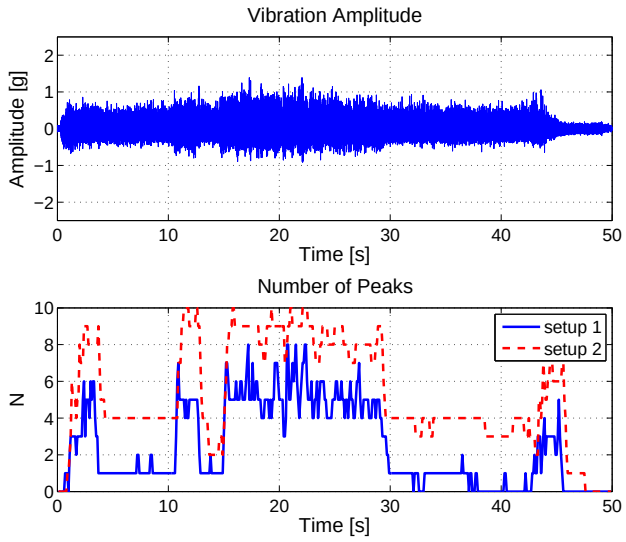


Figure 24: *Vibration signal and the corresponding number of peaks*

The vibrations level during the process can be indirectly assessed by counting the number of peaks that raise at a certain time instant. Cutting forces are not purely periodic because of the tool oscillations that introduce random varying components. Therefore the spectrum of measured vibrations is characterized by several peaks in the sidebands of the cutting harmonics. If vibrations are low these components are generally masked by the

background noise but, in case of anomalous or too severe cutting conditions, their height increases and they appear in the spectrum. Counting the number of peaks in the PSD provides an evaluation of the magnitude of the vibratory effect independently from the tooth passing frequency. Similarly to the previous vibration index it is also possible to focus the analysis on a desired band by counting the peaks in a limited frequency range. An important issue of this method is the need to consider in the count only the peaks affected by the increasing vibrations. For this reason it is required to set a threshold in order to avoid the addition of the noise peaks in the index computation that would make hardly noticeable the variations in the vibration level. For this reason a preliminary analysis is needed in order select the proper threshold. By measuring the number and height of peaks in case of good working conditions it is possible to tune the vibration index so that it returns zero in case of acceptable vibrations. The plot of figure 24 shows an example application of this method. The peaks count proves to be very sensitive also to slight changes in the vibration level. However because of the threshold there are numerous spikes that make it noisy (blue solid line). It is possible to reduce this effect by modifying the width of the moving window in which the computation is performed (red dashed line). This improve the frequency resolution of the PSD increasing the number of peaks in the band of interest. However a wider window implies also a delayed detection of the vibration level and then a tradeoff between the responsiveness of the algorithm and a stable behavior must be found.

3.1.3 *Remarks*

The choice of the most suitable signal feature among the three proposed has been performed analyzing the vibration measurement acquired during the roughing operation of a prismatic cast iron workpiece. Data have been collected during the entire part machining in order to consider several working conditions including the first passes on the raw surface of the cast workpiece and the effect of tool wear. This in addition with the geometric features of the part, as pockets and section variation, provided a wide variety of vibratory conditions that have been used for the selection of the most suitable vibration index. As an example, it can be considered the results shown in figure 25. The first plot shows the accelerometric measure of the Y axis of a face milling operation where at $0\,s$ the tool begins the engagement with the workpiece. The vibration level raises and settles after a short time. Actually it can be noticed a longer transient where the vibrations decreases taking around $20\,s$ to reach a stationary condition. This is due to the increase of process damping related to the tool wear that in this pass was at an advanced stage. Finally the most significative contribution in the vibration level is given between $50\,s$ and $62\,s$ where a variation of the cutting width occurs due to the tool passing over a pocket. The change of cutting conditions causes a consistent increase of vibrations. The behavior of the proposed signal features is shown in the remaining plots. Each method is capable to detect the variations on the vibration level with the RMS and the

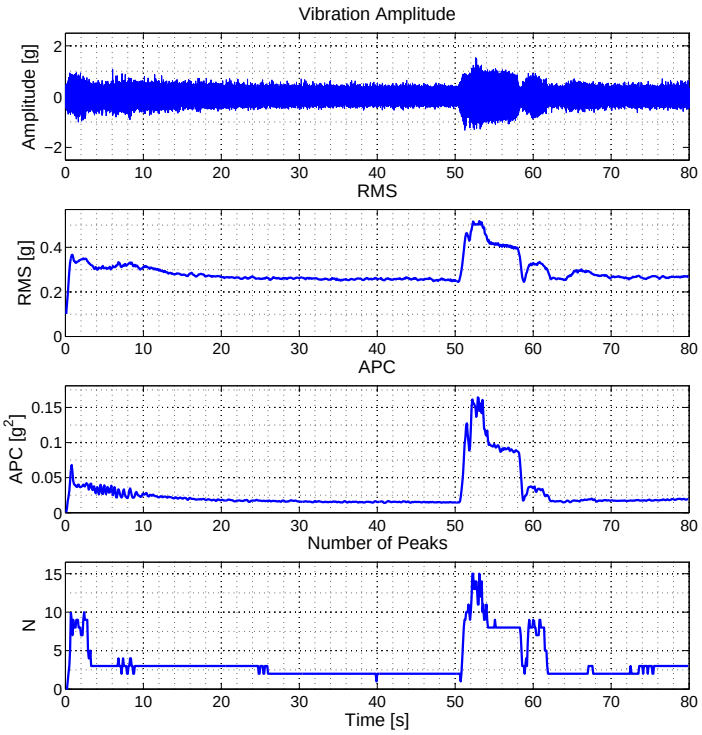


Figure 25: *Comparison among the three proposed signal features*

APC indexes that can follow more accurately the trend of vibrations. Instead the APC value and the peaks count appear more sensitive to changes of the vibrations magnitude. However, it can be noticed that the peak count returns almost the same value both for the vibrations related to the tool engagement and to the passage over the pocket. This can lead to a wrong estimate of the vibratory effect and then to a malfunction of the monitoring system. As previously discussed, this issue can be overcome by adjusting the window size. The accuracy and the sensitivity of the vibration index can be improved with a larger window but this reduces the responsiveness of the monitoring task as in the case showed in figure 26 where the extraction of the signal feature by a window of $2\,s$ implies a latency of $\approx 1\,s$ in the detection. The peaks count method is then the most disadvantageous for the implementation in the monitoring algorithm since it needs an accurate tuning both for the choice of the minimum peaks height threshold and for the proper size of the computation window. Moreover a large window involves an high number of samples and then increases the computational burden of the FFT-based signal processing. For this reason the use of this method has been considered improper for the implementation on the microcontroller of the *monitor* nodes. According with the results, the vibration index more suitable for the VM task is the APC both for the sensitivity to the vibration level and for the detection accuracy. However the computational drawbacks related to the use of the FFT algorithm on a large amount of data is still

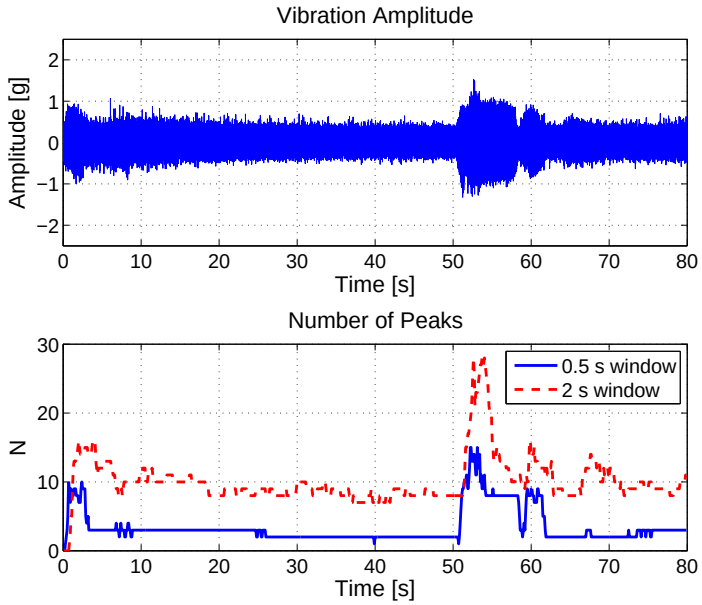


Figure 26: *Window size effect on the peaks count index. The use of a larger window improves the accuracy of the evaluation of vibration level.*

present. In fact each *monitor* has to manage, in the worst case, the acquisition of three axes from 4 accelerometers at $5\,kHz$ (cf. §2.3.1). Considering the use of a $0.5\,s$ time window, as in the exampled described above, the total amount of samples is equal to 30000 and the FFT computation must be repeated for each axis and sensor. This means an high processing time and the need of almost $200\,kB$ of memory only for data storage. *Monitor* units have only $8\,kB$ of internal RAM available and then the data has to be moved in an external memory. As a consequence the computation is further slowed by the time needed to access the external memory. However this limitation has been overcome considering the meaning of the APC value. The power of a signal $s(t)$ can be expressed according to the *Parseval* theorem as:

$$P = \frac{1}{T}\int_{-\infty}^{\infty} s(t)^2\,dt = \int_{-\infty}^{\infty} |S(f)|^2\,df \tag{9}$$

where $|S(f)|^2$ is the PSD $s(t)$. Moving to discrete time notation and considering the $0 \leq f \leq \frac{f_s}{2}$ range (single sided spectrum), the equation (9) can be rewritten using the periodogram equation (5):

$$\frac{1}{N}\sum_{j=0}^{N-1} |s_j|^2 = \sum_{k=0}^{\frac{N}{2}-1} \hat{P}(k\,df)df \tag{10}$$

The LHS term of above equation is equal to the squared value of the RMS (3), while the RHS term is the definition of the APC value computed on the entire frequency band (7). Therefore according to the *Parseval* theorem, using the squared RMS is equivalent to the computation of the APC. Moreover the possibility to obtain the APC of a limited frequency band can be obtained as

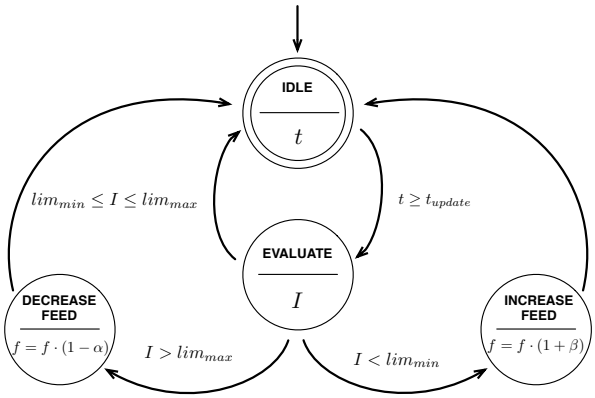


Figure 27: *State machine scheme used in the validation tests.*

well in time domain by filtering the signal before the RMS computation. This equivalence provides a big advantage since it allows a more efficient implementation of the VM algorithm in compliance with the design requirements as will be described in section 3.1.5.

3.1.4 *Experimental Validation*

The selected vibration index has been implemented on a machining supervisory system used to experimentally verify the effectiveness of the proposed method. The system was intended to modify the feed velocity according to the measured vibrations in a similar way to the machine operator action. Two thresholds are set to define a range in which the vibration level is considered acceptable and no feed regulation is required. If the vibration level exceeds these limits the feed is adjusted in order to restore the desired machining conditions. The VM algorithm has been implemented on the NATIONAL INSTRUMENTS NI DAQ 9188 device with an analog input module for data acquisition and an analog output module for feed tuning. The sensor was the DYTRAN 3213M6 triaxial piezoelectric accelerometer already used to collect the vibration data in the first part of the activity. A state machine with the architecture shown on figure 27 has been used for the implementation of the monitoring task. After the initialization, the system is kept in the idle state where the data acquisition is performed. At regular time intervals, defined by the update frequency of the analysis window, the evaluation state is enabled. In this operative condition the squared RMS value is computed and compared with the two vibration thresholds. If the vibration index exceeds the upper limit, the system reduces the commanded feed of a fixed amount. Vice versa if the vibration level is too low, the axis speed is increased in order to fall inside the desired range. The rate of feed variation can be different for the decreasing and the increasing phases. In the experiments the feed reduction rate was higher in order to produce a fast variation on the machine dynamics, while the increasing rate was selected in order to smoothly approach to the desired range. The response of the su-

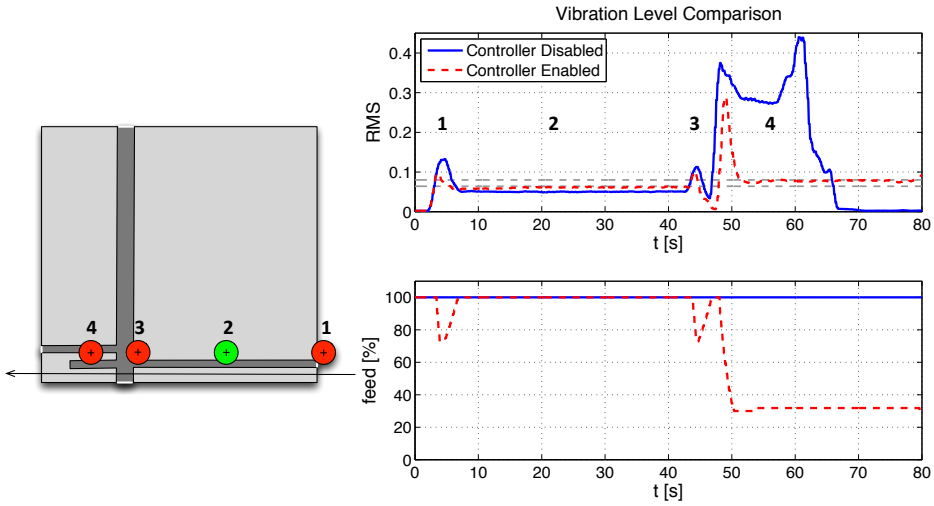


Figure 28: *Workpiece scheme and the effect of the supervisory system in the control of vibrations.*

pervisory system to the vibratory effect can be seen in figure 28 (red dashed line) where it is shown the result of one of the machining tests. The experimental activity consisted of a face milling operations on a steel workpiece. The working area was a flat surface with a vertical cut on the left side and two horizontal cuts at the bottom. The proposed results concern a working operation executed moving the tool from right to left over the two cavities as shown in the left scheme of figure 28. This test is meaningful since there are four different cutting conditions where the system action can be recognized in comparison with a traditional machining operation (blue solid line). According to the results the occurred cutting conditions can be described as follows:

1. The mill engages the workpiece with a programmed feed of $1000\,^{mm}/_{min}$ and the vibration index raises overcoming the upper threshold. The supervisory system reduces the axis velocity settling at $730\,^{mm}/_{min}$ ($73\,\%$).

2. After the mill has fully entered the workpiece, vibrations decrease under the safety range. As a consequence the feed is increased up to the nominal feed. In this case the feed increase was constrained to the commanded value but the system can be programmed to exceed it in order to increase the working speed.

3. The mill reaches the vertical cut and starts exiting from the workpiece. Vibrations slightly increase and the controller pushes the feed down to $730\,^{mm}/_{min}$.

4. The mill passes straddling the two horizontal slots and begins to work in interrupted cutting conditions. This is a particularly severe operation for the tool whose cutting edges repeatedly enter and exit the material with impacts and tears. This leads to high tool vibrations and

consequently the system commands a marked feed reduction. The feed value is brought to $320\,mm/min$ (32 %) leading the vibration level within the desired range.

The results show that the supervisory system can successfully mitigate the excessive vibratory effects that may occur during the working operations. In the next section will be described the implementation of the VM monitoring task in the *monitor* unit used in the distributed monitoring system.

3.1.5 *Vibration Monitoring Algorithm*

Thanks to the result obtained by the supervisory system described in the previous section, the algorithm based on the RMS vibration index has been implemented for the VM task of the *monitor* nodes. The implementation has been organized in two activities. The first one concerns the control of vibration level while the second one is the learning phase used to configure the vibration thresholds according to the working operation and the machining conditions. Then the VM task has been organized in the following steps:

DATA ACQUISITION Vibration measurements are collected for each accelerometer and immediately processed to obtain the vibration index.

FEATURE EXTRACTION The resultant RMS value is computed for a set of data and continuously updated.

INTERVENTION the obtained feature is periodically compared with the vibration limits and the feed variation is commanded to the machine.

LEARNING The computed vibration index is used as a reference to set the upper and lower thresholds.

Data Acquisition and Feature Extraction

When the machine operator enables the VM task, the *monitor* node performs the setup of sensors, loading the required configuration and then, it checks if sensors are correctly connected. If a malfunction is detected, a warning is sent to the MMI and the number of available sensor is updated. In any case the VM procedure is initiated measuring the offsets of data for each sensor and storing them in the `vm_offset` matrix. At this point the system is ready and the acquisition is enabled (`vm_isr` $= 1$). The acquisition and the feature extraction phase is inserted inside an interrupt routine triggered by the sensors when a new set of data is ready for the transmission. Signal feature is computed as soon as the acquisition is performed. Also in this case the vibration index is extracted from a signal interval updated with new data at a fixed rate. Since the resultant RMS^2 is obtained from a sum of values, this observation window has been implemented as a circular buffer where each element contains the squared sum of acceleration data acquired within the update time interval (line 3 of listing 2). Moreover for each set of data the offset compensation is also performed before the window update.

The buffer element is filled until the number of acquired sample is equal to the VM_BUFFER_UPDATE constant. At this point the actual feature extraction begins (line 13 of listing). The last updated element buffer is added to the current RMS^2 value given by the rms2 variable. At the same time the oldest buffer element is subtracted from rms2[i] and subsequently it is reset in order to be used for the next acquisitions. Once the new RMS^2 value has been computed the data ready flag is enabled (vm_drd = 1 at line 26) making the vibration index available in the main routine for the intervention phase.

```
/*VIBRATION MONITORING DATA PROCESSING*/
void VM_processSample()
{       /*ACQUISITION*/
        for (uint8_t i=0; i<N_ACC; i++){
                int_sq_vals[vm_index & (VM_BUFFER_SIZE - 1)][i] +=
                (uint32_t)(rawData.aX - vm_offset[i][0]) * (uint32_t)(rawData.aX -
                vm_offset[i][0]) + (uint32_t)(rawData.aY - vm_offset[i][1]) *
                (uint32_t)(rawData.aY - vm_offset[i][1]) + (uint32_t)(rawData.aZ -
                vm_offset[i][2]) * (uint32_t)(rawData.aZ - vm_offset[i][2]);
        }
        /*UPDATE COUNTER*/
        counter++;
        if(counter == VM_BUFFER_UPDATE)
        {       /*RESET COUNTER*/
                counter = 0;
                /*UPDATE RMS BUFFER*/
                for(uint8_t i = 0; i<N_ACC; i++){
                        rms2[i] = rms2[i] - int_sq_vals[(vm_index+1) &
                        (VM_BUFFER_SIZE - 1)][i] + int_sq_vals[vm_index &
                        (VM_BUFFER_SIZE - 1)][i];
                        int_sq_vals[(vm_index+1) & (VM_BUFFER_SIZE - 1)][i] = 0;
                }
                /*UPDATE BUFFER INDEX*/
                vm_index++;
                /*SWITCH DATA READY FLAG ON*/
                vm_drd = 1;
                /*CHECK LEARNING REQUEST*/
                if (gpio_pin_is_low(GPIO_PUSH_BUTTON_6)){
                        /*SWITCH LEARN FLAG ON*/
                        vm_learn = 1;
                }
        }
        /*RESET THE INDEX VALUE*/
        if(vm_index == 10*VM_BUFFER_SIZE){vm_index = 0;}
}
```

Listing 2: *Source code for data acquisition and vibration index computation*

Vibration Monitoring Action

When the vibration index is updated, the evaluation of the vibration level is performed. First the actual RMS^2 is computed, dividing it by the number of samples in the observation window. Then the function `VM_get_feed` is called and the code showed in listing 3 is executed. The RMS^2 value is compared with the predefined vibration limits. If the index fall inside the two thresholds no action is required, but if the limits are exceeded the decrease or increase command is sent to the NC that will change the feed value according to the programmed rates.

```
/*FEED CONTROL OUTPUT*/
int8_t VM_getFeed(float*rms)
{
        /*CHECK VIBRATION INDEX VALUE*/
        for(uint8_t i = 0; i<N_ACC; i++)
        {
                /*VIBRATIONS EXCEEDING UPPER LIMIT*/
                if (rms[i]>vm_highLim)
                {
                        ioport_set_pin_low(LED3);
                        printf("Warning Upper Limit Exceeded\n");
                        return -1;
                }
                /*VIBRATIONS BELOW THE LOWER LIMIT*/
                else if (rms[i]<vm_lowLim)
                {
                        ioport_set_pin_low(LED1);
                        return 1;
                }
                /*VIBRATIONS WITHIN THE LIMITS*/
                else
                {
                        ioport_set_pin_low(LED2);
                        return 0;
                }
        }
}
```

Listing 3: *Source code for the intervention of the vibration monitoring system*

Vibration Limits Learning

A generic cutting operation can be executed under very different machining condition that can affect the allowed vibration limits. For example vibrations generated during a roughing operation can not be accepted in finishing, and the vibration level that can be tolerated by a stubby mill will be higher than that one tolerated by a slender mill. In addition the machine operator could prefer to increase the productivity of the machine (acting on the cutting parameters) at the expense of more severe cutting conditions. For these reasons the vibration threshold can be considered inadequate to the current process and then it can be required to change the default limits. The VM task has been provided with a learning functionality that can be used whenever the vibration limits are not suitable. When the machine operator wants to change the allowed vibration range, he has to perform a working operation at the machining conditions that considers appropriate. As soon as the proper conditions are reached, he commands the learning function. The request enables the learning flag (e. g. line 3 of listing 2) and in the main routine the function `VM_startLearning` is called (listing 4). The controller action is then disabled and the computed vibration index is collected for a predefined amount of time. At the end the averaged value of the RMS^2 is used as the upper threshold while the lower limit is set as a fraction of the previous one. The learning procedure can be commanded at the start up of the VM task or during its execution and can be repeated every time is needed.

```
/*VIBRATION THRESHOLD LEARNING*/
void VM_startLearning()
{
        uint16_t j = 0;
        float learnt_rms = 0.0;

        /*START LEARNING*/
        while (j<N_DATA)
        {
                while(vm_drd == 0){}
                learnt_rms += ((float)rms2[0]/(float)VM_N) * resolution;
                j++;
                vm_drd = 0;
        }
        /*SET LIMITS*/
        vm_highLim = learnt_rms/((float) N_DATA);
        vm_lowLim = vm_highLim*(1-0.2);
}
```

Listing 4: *Source code for the learning phase*

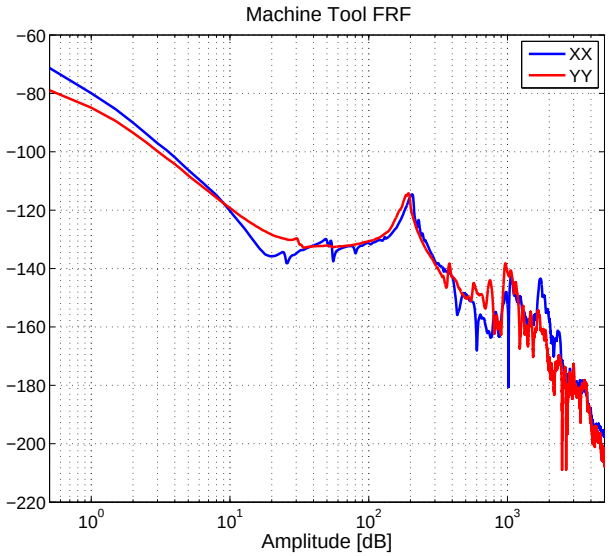


Figure 29: *Frequency response of the machine tool obtained by a tapping test along* X *and* Y *directions*

Further Improvements

The proposed approach provides an effective method to assess the vibration level of the machining operation. The implemented algorithm allows to manage a large number of data while keeping a low computational burden that makes it suitable for a wide range of microcontrollers with limited memory and performance. For example the algorithm testing has been performed on a observation window of 3200 samples ($\approx 0.64\,s$) that required only 128 *Bytes* of memory for each sensor. In addition the same vibration index can be also used to monitor the occurrence of chatter. Chatter is a self-excitation mechanism originated by the chip formation that leads the system to instability causing a sudden and continuous growth of the cutting vibrations. When chatter occurs the system is excited at the chatter

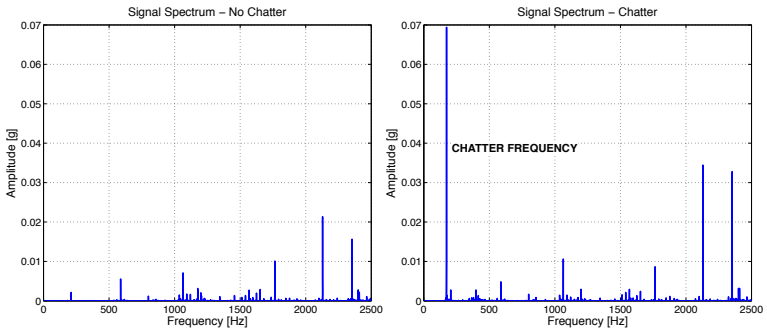


Figure 30: *Signal spectrum in two different machining conditions: without chatter and with chatter. In the second case the structural mode is excited by chatter*

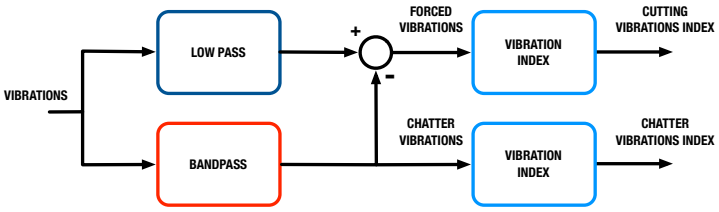


Figure 31: *Vibration monitoring architecture for the detection of chatter vibrations*

frequency that usually is close to one of the dominant structural modes of the machine. Therefore by knowing the frequency response of the machine tool it is possible to assess the frequency that is more likely to be affected by regenerative chatter. Figure 29 shows the FRF of the machine tool in X and Y directions obtained through an impact test at the tool tip. Both the FRFs shows a resonance at $\approx 200\,Hz$. Experimental tests run in different machining conditions showed that the chatter occurrence excites that structural mode that becomes more relevant than the remaining harmonics how can be seen analyzing the spectrum of vibrations in stable and unstable cutting conditions shown in figure 30. In this case the dominant frequency of the vibratory effect is unrelated to the cutting harmonics and is very close to the structural resonance experimentally found around $200\,Hz$. Given the FRF of the machine tool is then possible to extract information on chatter occurrence from the acceleration signal by using the RMS-based vibration index. The contribution of forced vibrations and self excited vibrations can be separated by using a low pass filter to extract the cutting harmonics and a bandpass filter to isolate the chatter frequency as shown in figure 31. The application of this method to measured data shows (Fig. 32) that the source of vibratory effect can be easily distinguished between forced and self excited vibrations.

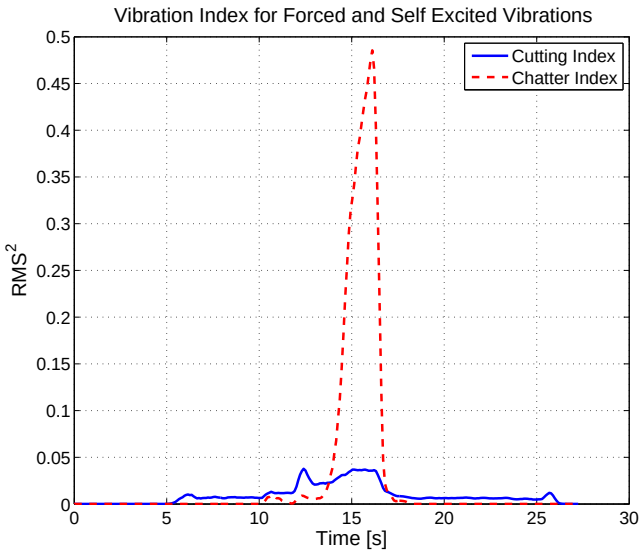


Figure 32: *Discrimination between high vibration level and chatter occurrence*

This allows to treat separately the two contributions by designing a specific intervention strategy for each of them. For example in case of chatter the VM monitoring action can be more effective by tuning the spindle speed rather than the axes feed. As for the forced vibrations the use of the RMS-based index reveals a more efficient method to promptly detect undesired working conditions compared to techniques based on spectral analysis.

3.2 CONDITION MONITORING

CM of machine elements refers to a set of techniques that, starting from the measurement of parameters related to the operative condition of the machine , allows to assess the health status of its parts. The parameters are usually vibrations, operating temperature, pressure of fluids and so on. The CM task consists of the evaluation of machine health and the detection of faulty elements. If the a damage is found the CM can determine the causes and its severity. The obtained information is used to develop the maintenance plan considering the actual condition of the machinery and replacing the traditional run-to-breakdown and preventive maintenance approaches. The assessment of the actual machine status, performed by the CM activity, leads the so called condition-based maintenance approach that allows an optimal planning of the maintenance extending the operational life of the plant. At the same time it allows also to avoid catastrophic failures that could cause severe damages to the machine and even dangerous situation for the machine users. As reported on [97] a well designed condition-based maintenance program leads to a reduction of downtime by $35-45\,\%$, an elimination of breakdowns by $70-75\,\%$, a reduction of maintenance costs of $25-30\,\%$ and a production increase of $20-25\,\%$. Usually simple tasks performed by machine operator as visual inspections and annotations about anomalous sounds, vibrations or odors can be useful to notice a possible failure of the machine, but however can not be considered an effective way to evaluate the machine health. CM tasks are intended to provide accurate and useful information on the existing faults in a timely manner. The activities involved in CM can be grouped as follows:

DETECTION It is the result of the measurement, collection and analysis of machine parameters that are used to estimate the machine condition. The current condition is compared with that one of the new machine or with predefined limits and is used to build trends and the history of machine status.

DIAGNOSIS Consists of the recognition of damage development and of its severity.

PROGNOSIS Provides the expected time to failure and allow the maintenance planning.

Other complementary activities include failure root-cause investigation, damage analysis and modeling, that altogether allow to improve the process condition, the monitoring strategy and even the machine design. Despite

the advantages related to the use of CM tasks there are also drawbacks due to the need of skilled personnel and costly equipment. Moreover it is also required an accurate planning to limit the costs of monitoring activities that can make CM disadvantageous. In order to overcome these limitations the use of the distributed monitoring system has been considered for the assessment of the machine status. In this work the aim was to implement a CM algorithm capable of detect damage on the spindle bearings by processing the vibrations signals gathered from the accelerometers and provide diagnostic information on the fault. The CM task has been designed as an activity that can be periodically run by the machine operator during the non-working time of the machine (e. g. before starting a working operation). The resulting information can be easily interpreted by the operator that does not require a specific knowledge on the topic. According to the monitoring system response the machine operator consults maintenance technicians that can access to more detailed data stored on the system for further analysis. Therefore CM activities can be split in two phases, one more frequent that is performed by the operator and involves a fast check of the machine status and the other one that is executed only when a significant damage is detected and a more accurate analysis is needed.

3.2.1 *Vibration Based Condition Monitoring*

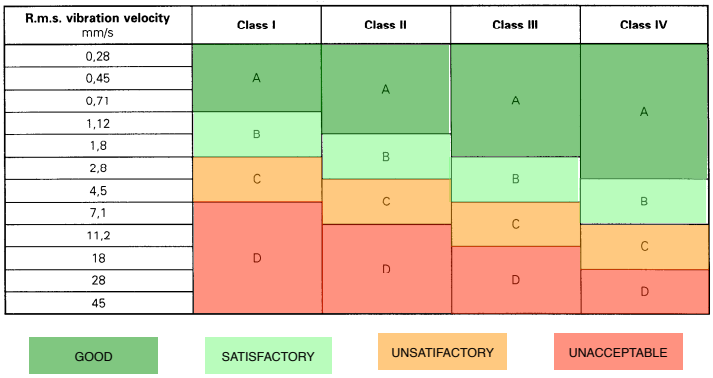


Figure 33: *Vibration limits according to ISO 10816-1 norm*

The assessment of machine condition from vibration measurement has been widespread used ant there exist numerous techniques for machinery diagnostic both in time and frequency domain ([98, 99]). The frequency spectrum of a rotating machine provides information on damaged elements allover the frequency band that can be classified as follows:

LOW FREQUENCY RANGE Frequency components due to misalignment, bent shafts and unbalances usually amplify the firsts harmonics of the shaft speed. Mechanical looseness appears in the low frequency band with interharmonic components.

MEDIUM FREQUENCY RANGE In this range the tooth mesh components of gearboxes modify the spectrum as the wear or damage grows. Moreover the increased deflection due to a broken tooth will increase the low level sidebands around the toothmesh peaks.

HIGH FREQUENCY RANGE Resonance frequencies of the machine structure are usually located on the high frequency range of the spectrum. Structural resonances amplify the components coming from a damaged bearing that usually have low energy. In this frequency range is then possible to detect a faulty bearing more easily than in the lower side of the spectrum.

Standardized methods for the detection of machine faults involves mainly the comparison of RMS values of vibration velocity with previous measurement or with predefined limits in the $10 - 1000\,Hz$ frequency range. Figure 33 shows the vibration limits for different class of rotating machines according to the ISO 10816 norm [100]. The standard guidelines provide a good indication on the current status of the machinery but are not able to recognize condition change and the fault evolution. Moreover the broadband analysis of vibrations can early detect severe misalignment or bent shafts while lower energy signals as gearbox or bearings faults will not be detected until the generated vibrations will overcome the dominating low frequency signals. For these reason acceptance limits are often defined to discover changes of nominal conditions. These limits are used to early detect variation on machine behavior not related to normal operative conditions and can be set for a specific machine resulting in a more accurate monitoring. Figure 34 shows an example of acceptance limits applied to the frequency spectrum of the machine. These limits create a synthesized spectrum that is used as a boundary for the measured one. The prompt discover of anomalies and the avoidance of false alarms are the two aspects considered in the choice of acceptance limits that are usually defined by experience and through statistical methods. Every time a limit is exceeded, a further analysis of the acquired signals must be performed in order to find the cause of vibration rise. The diagnosis of the possible damage starts from the analysis of the band where the anomalous component is located. According to the classification provided above the frequency range give a first indication on the fault nature and then specific techniques can be used to identify and assess the damage. In the next section will be described the developed monitoring algorithm for the CM task whose purpose is the detection and diagnosis of faults in spindle bearings. For this reason the Envelope Analysis (EA) of vibration has been implemented. EA is a well known diagnostic tool that allows to isolate the component of interest from the whole vibration signal and to analyze its frequency content to find the damage source. This technique is particularly suitable for the extraction of low energy pulses, due to a damaged bearing or a cracked gear tooth, from the background vibrations.

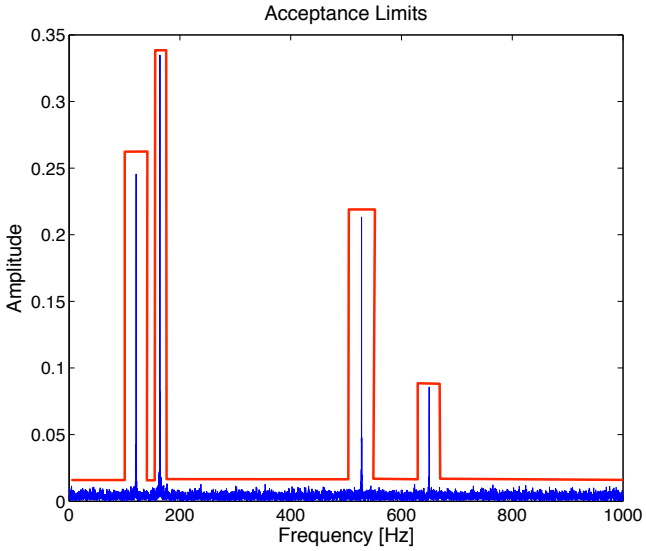


Figure 34: *Acceptance limits based on synthesized machine spectrum*

3.2.2 *Condition Monitoring of Bearing*

Bearing Failure Modes

Faults on rolling element bearings originate from localized damage on the raceways or on one rolling element due to metal fatigue that produces cracks and corrosion pits (Fig. 35a). The damage grows over time resulting in widespread spalling and material detachment till to the catastrophic failure of the bearing. An other issue is given by excessive static loads that cause the indentation of raceways (*brinelling*) and the deformation and looseness of the cage (Fig. 35b). Finally, in bearing equipping electric motors, stray currents can produce electrical erosion of the race surface (Fig. 35c). The passage of the rolling elements over the damage produces small pulses repeating at the ball pass frequency. Pulses are transmitted through the bearing housing, producing a vibration signature characterized by impulsive events with defined frequencies. As the damage grows the amplitude of vibration related to roller element harmonics increases and, when the fault

(a) (b) (c)

Figure 35: *Bearing failure modes: spalling 35a, brinelling 35b and electric erosion 35c*

becomes very large, it decays while the broadband noise increases resulting in the end of bearing life. The effects of bearing faults are increasing vibration, noise and temperature and finally catastrophic breakage. Monitoring bearings health is then an important task since their failure can cause more severe damages to the machine. However, especially in its early stage, the pulses caused by the damage have low energy and are buried by the higher vibrations of other machine parts. The signal is then hardly noticeable without a proper processing as the EA. Neugebauer [95] analyzed different monitoring techniques to assess bearings condition on machine tool spindle. He found that CM activities based on AE measurements or EA of vibration data were capable to detect faults well in advance. In particular AE-based method registered a condition deterioration after only $4000\,hours$ of operation ($\approx 50\,\%$ of the bearing life) due to lack of lubrication. The EA identified the damage after $6000\,hours$ ($\approx 15\,\%$ of the bearing life) just at the beginning of race wear. Other methods based on RMS measurement of vibration and temperature measurement were able to detect the fault only when the bearings were severely damaged. EA then is still recognized as the most effective method for bearing diagnostic. In the recent work of Bediaga [93] the EA (computed with the *Hilbert Transform*) resulted much more accurate in diagnosis than Amplitude Demodulation while other techniques as Cepstrum Analysis, FFT proved to be inadequate to that specific task.

Envelope Analysis of Bearings

EA of bearing is a signal processing technique that provides the spectrum of the bearing vibrations through the demodulation of the high frequency components of the machine vibrations. The peaks on the spectrum allows the diagnosis of the fault by comparison with the characteristic frequencies of the bearing given by equations 11-14:

BALL PASS FREQUENCY OF THE OUTER RACE

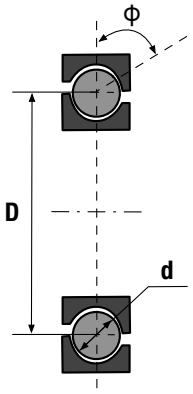
$$BPFO = \frac{n\,f_r}{2}(1 - \frac{d}{D}\cos\phi) \tag{11}$$

BALL PASS FREQUENCY OF THE INNER RACE

$$BPFI = \frac{n\,f_r}{2}(1 + \frac{d}{D}\cos\phi) \tag{12}$$

FUNDAMENTAL TRAIN FREQUENCY

$$FTF = \frac{f_r}{2}(1 - \frac{d}{D}\cos\phi) \tag{13}$$



BALL SPIN FREQUENCY

$$BSF = \frac{D}{2\,d}\,f_r(1 - \frac{d}{D}\cos\phi)^2 \tag{14}$$

where n is the number of roller elements of diameter d , f_r is the rotational

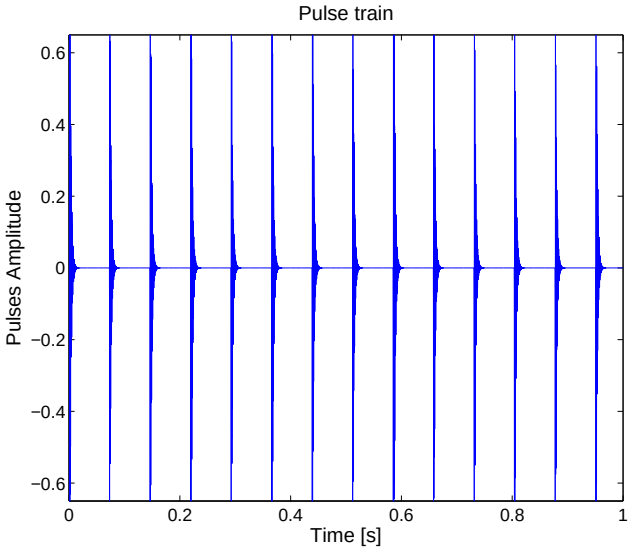


Figure 36: *Simulated damage signal with BPFO repetition frequency*

frequency, D is the pitch diameter and ϕ is the load angle. Bearing frequencies are generally low (within the first harmonics of shaft speed) and their harmonics are masked by other vibrations. However thanks to their impulsive nature it is possible to find the bearing harmonics in bands at high frequency where the contribution of other sources is neglectable. The effect of a single impact of the rolling element with the damaged can be modeled as the impulse response of a single Degree of Freedom (DOF) system and the vibration signal can be obtained as the system response to a train of pulses. In figure 36 is shown the simulated signal of a bearing with a damage in the outer race exciting the resonance of the modeled system set at $1.5\,kHz$. The rotational speed of the shaft is $200\,rpm$ and the Ball Pass Frequency of the Outer race (BPFO) is equal to $13.68\,Hz$. In addition the repetition frequency of the pulse train randomly varies of $\pm 1\,\%$ around the BPFO in order to include in the simulation the effect of slip of roller elements that is always present in real cases. These fluctuations cause a smear of the characteristic bearing frequencies making them even more difficult to detect in the raw spectrum. The whole vibration signal is then simulated by adding to the bearing signal the contributions of a small shaft unbalance ($3.33\,Hz$), of a gearbox with a tooth mesh frequency of ($180\,Hz$) and of random noise, that represent the signal of the healthy machine (blue solid line of figure 37). EA begins with the selection of the high frequency band that contains the pulses of the damaged bearing by means of band pass filtering of the original signal. The choice of the filter center frequency and bandwidth is a non trivial aspect of the analysis since a wrong filtering can compromise the final result. In common practice the filter is tuned according to the result of an impact test on the bearing housing or by monitoring the FRF of the machine. Alternatively the filter parameters are chosen in the band where a substantial rise of the high frequency spectrum is detected. In figure 37 the lower plot shows the spectra of the healthy machine (blue solid line) and

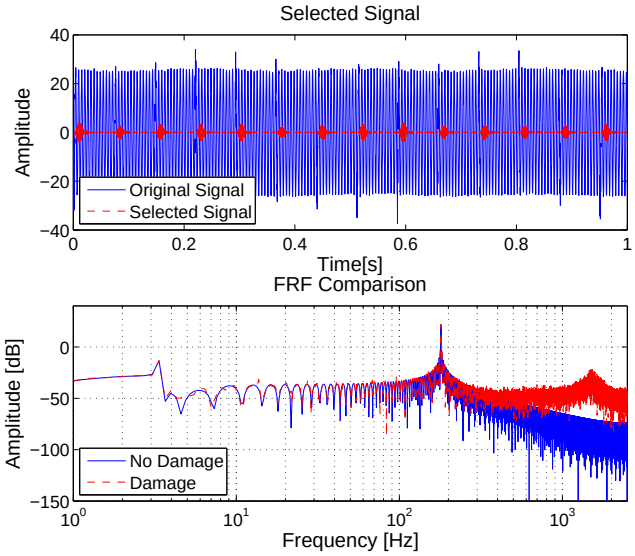


Figure 37: *Original and selected signal according to spectrum variation*

of damaged one (red dashed line). The signal of the faulty bearing causes an increase of spectrum amplitude at high frequency and the appearance of the resonance peak at $1.5\,kHz$. Then the bandpass filter is centered on this peak in order to extract the impulsive signal. The resulting filtered signal is shown in the upper plot of figure 37 together with the original one. The selected signal can be seen as an amplitude modulated signal constituted by the high frequency vibration (carrier component) and the low frequency impact repetition (modulating component). The next phase is then the amplitude demodulation that returns the modulating component as the envelope of the impact signal. The diagnosis is then performed comparing the peaks frequencies of the envelope spectrum with the characteristic frequencies of the bearing. If there exist a damage there will be a matching between one peak and the frequency related to the fault location. Figure 38 show the result of EA of the simulated damage. The upper plot show the bandpass filtered signal and the resulting envelope while in the lower graph the diagnosis of the damage on the outer race is performed in the envelope spectrum. In the next section will be discussed the implementation of the EA method in the CM algorithm together with an automated technique for the signal filtering.

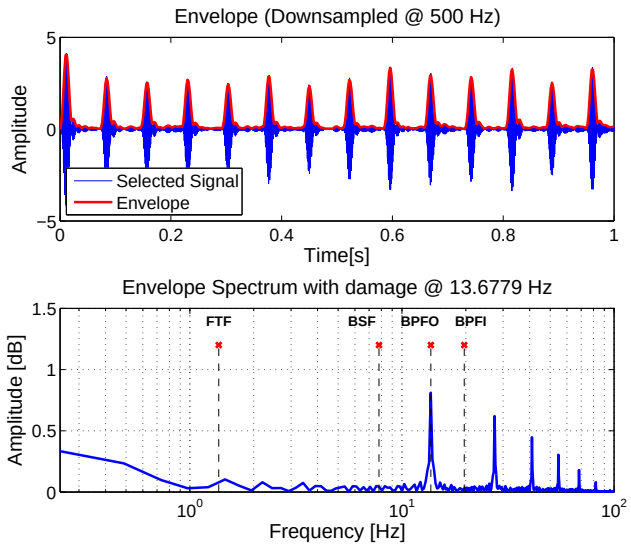


Figure 38: *Signal envelope and bearing diagnosis*

3.2.3 *Condition Monitoring Algorithm*

The CM algorithm developed for the monitoring system is based on the EA of vibration signal. The implementation has been organized in the following steps:

VIBRATION MEASUREMENT Vibrations are measured by the accelerometers placed on the bearings while the spindle is rotating at constant speed. Data are stored in the external RAM for further processing.

DAMAGE DETECTION Measurements are analyzed and the diagnosis is performed if a damage is detected.

DIAGNOSIS If the previous step gives a positive response, the signal is filtered and the EA is executed.

Data Acquisition

At the beginning of CM task, machine tool spindle is operated at constant speed while the machine is kept in standstill position. This is required in order to avoid wrong diagnosis due to the presence of cutting harmonics. The setup operation define the memory location for the acquired data (`cm_dataAddr`), provide the actual sampling frequency of each sensor and enables the acquisition (`cm_isr = 1`). The acquisition routine described in listing 5 is controlled through interrupts generated by sensors (cf. §2.3.1) that call the `CM_processSample` function. When the interrupt is triggered, the system reads the data from one accelerometer at a time and stores the vibration value on the external RAM (lines 5-13). The operation is repeated until the counter reach the acquisition limit and then the interrupt flag is disabled (`cm_isr = 0`) and the counter reset. For this application the limit

is set to 16384 data that corresponds approximately to a $3\,s$ acquisition, depending on the actual sampling frequency of each sensor.

```
/*CONDITION MONITORING DATA ACQUISITION*/
void CM_processSample(accData_raw_t*accData)
{
        /*ACQUIRE ACCELERATION DATA FOR EACH ACCELEROMETER*/
        for (uint8_t i=0; i<N_ACC; i++)
        {
                /*GET THE ACCELERATION DATA*/
                getAcc_raw(i,(unsigned char*) &accData);
                /*STORE VALUES IN THE EXTERNAL RAM*/
                hugemem_write16((hugemem_ptr_t) cm_dataAddr,accData->aZ);
                /*UPDATE THE MEMORY ADDRESS*/
                cm_dataAddr+=2;
        }

        /*UPDATE THE COUNTER*/
        counter++;
        if (counter == CM_N)
        {
                /*EXIT FROM ISR ROUTINE*/
                cm_isr = 0;
                /*RESET COUNTER*/
                counter = 0;
        }
}
```

Listing 5: *Acquisition of vibration data source code*

Damage Detection

Once the acquisition is completed, the collected data for each sensor are processed in order to detect the damage. According to the description of EA provided in section 3.2.2 the assessment of a possible fault is performed by the maintenance engineer that tunes the envelope detector or the CM software relying both on machine natural frequencies and significant changes on the vibration spectrum. Then the traditional approach is not suitable for an automated application since it would require adjustments by an experienced user and a database with the response of the healthy machine. Due to the impulse-like nature of bearing defects, in the past were proposed detection techniques based on indicators sensitive to the sharpness of peaks as the crest factor or kurtosis of signal [101]. However in some case these methods failed to assess the actual bearing condition, especially when noise

level was high and the damage at its early stage. Recently a novel approach based on Spectral Kurtosis (SK) technique is becoming very popular for damage detection and diagnostic applications. Starting from the work of Antoni [102], that demonstrated the effectiveness of SK to describe the frequency content of transient signals, and the further research of Antoni and Randall [103, 104] that proposed the use of SK as a tool for monitoring and diagnosis of faults on rotating machines, several authors exploited this method for the development of CM techniques (e. g. [105, 106]). SK is a statistical tool that overcomes the limits of PSD in the evaluation of the frequency content of non-stationary components in a signal. Kurtosis is a statistic index used to measure the shape of the probability distribution of a random variable or, in other words, its deviation from the normal distribution. Kurtosis is computed as the ratio between the fourth moment of the mean normalized with respect to the fourth power of the standard deviation (Pearson's index 15) minus three:

$$\beta_2 = \frac{\mu_4}{\sigma_4} = \frac{E\left[(X-\mu)^4\right]}{\left(E\left[(X-\mu)^2\right]\right)^2} \tag{15}$$

$$\gamma_2 = \beta_2 - 3 \tag{16}$$

The Pearson's index of a normal distribution is $\beta_2 = 3$ and then the its Kurtosis is equal to zero. According to the definition the Kurtosis can be useful to detect non-Gaussian components of a signal as a repeated impulse. However a low signal to noise ratio will mask the impulsive signal making its detection impossible. The SK of a signal is the computation of Kurtosis for each frequency band of the signal spectrum and then it allows to distinguish which frequencies contain impulse-like transitory components and which a stationary signal. SK is obtained starting from the computation of the STFT that provides the Fourier Transform of the signal $x(t)$ in a moving window along the time axis $X(t,f)$. If we denote the Power Spectrum (PS) at time t with $|X(t,f)|^2$ then the PSD can be computed as the average of the PS over time:

$$\begin{aligned} PS &= |X(t,f)|^2 \\ PSD &= \tfrac{1}{T}\int_0^T |X(t,f)|^2\, dt = \langle |X(t,f)|^2 \rangle \end{aligned} \tag{17}$$

The SK is then computed as follows:

$$K(f) = \frac{\langle |X(t,f)|^4 \rangle}{\langle |X(t,f)|^2 \rangle^2} - 2 \tag{18}$$

where the minus two is needed to have $K(f) = 0$ when $X(t,f)$ is the Fourier Transform of a stationary (Gaussian) signal. This means that SK is capable to detect the existence of a impulsive component in a signal (high $K(f)$ value) and to provide the frequency where this is more significant. SK has been successfully used, in the aforementioned works, for diagnostic of rotating machine and especially for those elements whose damage causes sharp

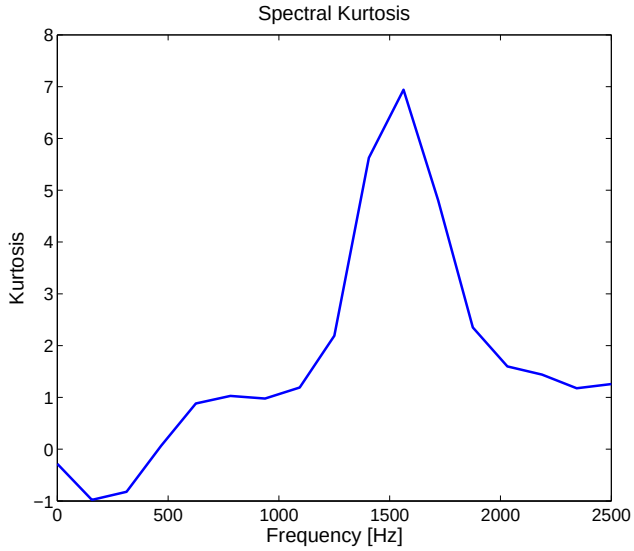


Figure 39: *Spectral Kurtosis of the simulated signal. The detected fault frequency is greater than the actual one due to the low frequency resolution.*

repetitive pulses as bearings and gearboxes. In figure 39 is shown the SK of the simulated bearing fault signal. The Kurtosis value indicates an impulsive component at 1562 Hz that is greater than the actual frequency because of the low frequency resolution resulting from the parameter selection of SK algorithm. Due to the advantages brought by SK, this method has been chosen for the execution of the automated fault detection in the proposed CM algorithm. In this way, the analysis of historical data of the machine is no more needed, as well as knowing the structural frequencies of the system. The last point is particularly important since the use of a traditional method would have required the dynamic characterization of the bearing houses for each machine configuration and accessory head. Instead with the SK, as center frequency of the bandpass filter is chosen that one with the maximum Kurtosis value. Listing 6 shows the damage detection and diagnosis phase of the CM algorithm. After the configuration of the parameters, at line 11 the `SpecKurt` function is called. The function receives as parameters the sensor id, the memory location of the raw data and the destination address of the SK values, the number of data, and the STFT parameters (window size and update frequency), the measured sampling frequency of the sensor and finally the vector containing the resulting center frequency. The SK computation is performed in the following steps:

1. FFT ($X(t_i, f)$) computation of the windowed signal.
2. PS ($|X(t_i, f_j)|^2$) and PS^2 ($|X(t_i, f_j)|^4$) computation for each frequency bin.
3. Add the computed PS values to numerator and denominator of SK.
4. Update the window and repeat steps $1 - 4$ until the end of signal is reached.

5. When the entire signal has been processed the Kurtosis is computed for each frequency bin as:

$$K(f) = \frac{\frac{dt}{T}\sum_{i=0}^{N}|X(t_i,f)|^4}{\left(\frac{dt}{T}\sum_{i=0}^{N}|X(t_i,f)|^2\right)^2} \tag{19}$$

6. The maximum Kurtosis is searched.

The `SpecKurt` returns 0 if $K(f) \leq 2$ or the frequency corresponding at the maximum Kurtosis value if $K(f) > 2$. If `SpecKurt` output is zero, the damage status is set to -1 and a negative message is sent for the operator(line 13). Otherwise the operator is informed that a damage has been detected and the EA procedure is started. At the beginning the bandpass filter and a lowpass filter are designed according to the frequency value returned by the `SpecKurt` function (line 23), then the `envelope` function is called (line 26). The envelope is computed in three steps:

1. The impulsive fault signal is extracted from vibration data by means of the IIR filter obtained by the `bpIIR` function.
2. The filtered signal is squared in order to perform its demodulation (*Square-Law demodulation*). Squaring the signal shifts half of its energy towards DC (the modulating component) and half of the energy to higher frequencies (carrier signal). The squared signal is then multiplied by two to compensate the energy loss.
3. The resulting signal is low pass filtered to remove high frequencies components obtaining the envelope of the original signal.
4. The envelope is downsampled and rescaled by performing the square root.

Finally the diagnosis is performed (line 28) according to the following steps:

1. FFT computation of the envelope.
2. Research of the peak closest to one of the bearing characteristic frequencies.

At the end of the computation a warning is set to the MMI indicating the detection of a fault on the examined bearing and a code that identify its location: 0 = FTF, 1 = BSF, 2 = BPFO, 3 = BPFI.

```
/*DAMAGE DETECTION AND DIAGNOSIS*/
void CM_computeData(uint8_t device_id)
{
        /*DOWNSAMPLED DATA SIZE*/
        uint32_t dataSize = CM_N/ENV_DOWNSAMPLE;
        /*FREQUENCY RESOLUTION*/
        float fRes = cm_device_params[device_id].sampling_frequency/CM_N;
        /*RESET SK MEMORY ADDRESS*/
        cm_dataAddr = ADDR0;
        cm_skAddr = ADDR1;
        /*SK AND DAMAGE DETECTION*/
        valSK[device_id] = SpecKurt(device_id,cm_dataAddr,cm_skAddr,CM_N,SK_WINDOW
    ,SK_WUPDATE,cm_device_params[device_id].sampling_frequency,&centerFreq[
    device_id]);
        /*SK OUTPUT*/
        if (centerFreq[device_id] == 0) {
                damageKey[device_id] = -1;
                printf("SK VAL%f => no damage detected on bearing = %d\n",valSK[
    device_id],device_id);
        }
        else {
                printf("SK VAL%f => possible damage detected on bearing = %d\n
    Performing Envelope Analysis",valSK[device_id],device_id);
                /*RESET ENVELOPE ADDRESS*/
                cm_dataAddr = ADDR0;
                cm_envAddr = ADDR2;
                /*FILTERS DEFINITION*/
                lpF = single_lpIIR(4,(uint16_t) centerFreq[device_id]/2 ,
    cm_device_params[device_id].sampling_frequency/ENV_DOWNSAMPLE);
                bpF = bpIIR((uint16_t) centerFreq,cm_device_params[device_id].
    sampling_frequency,ENV_BW);
                /*ENVELOPE ANALYSIS*/
                envelope(device_id,cm_dataAddr,cm_envAddr,CM_N,&bpF,&lpF,
    ENV_DOWNSAMPLE);
                /*DIAGNOSIS*/
                damageKey[device_id] = diagnosis(cm_envAddr,dataSize,fRes);
                printf("damage on bearing = %d\n",*fault_code);
                }

}
```

Listing 6: *Diagnostic algorithm source code*

3.2.4 *Experimental Results*

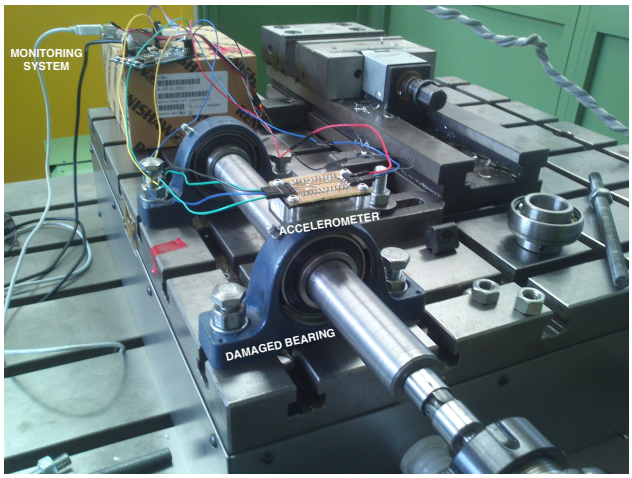


Figure 40: *Bearing test bench for* CM *algorithm testing*

The experimental validation of the CM algorithm has been performed on the test bench showed on figure 40 consisting of a shaft supported by two bearings and operated by an electric motor. One of the bearings has been disassembled and the inner race has been damaged with a grinder. The accelerometer has been mounted on top of its housing and the diagnostic algorithm has been used to assess the bearing status testing both the detection and the diagnostic tasks. The parameters and the characteristic frequencies computed with equations provided in section 3.2.2 are reported on Tab. 4.

BEARING	
PARAMETERS	
PITCH DIAMETER D	46 mm
BALL DIAMETER d	9.525 mm
LOAD ANGLE ϕ	0 °
ROTATIONAL FREQUENCY f_r	3.33 Hz
CHARACTERISTIC FREQUENCIES	
FTF	1.3216 Hz
BSF	7.7039 Hz
BPFO	11.8940 Hz
BPFI	18.1060 Hz

Table 4: *Bearing parameters and characteristic frequencies*

Healthy Bearing

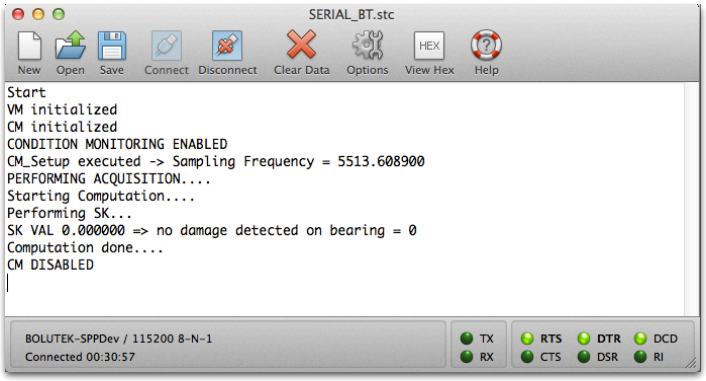
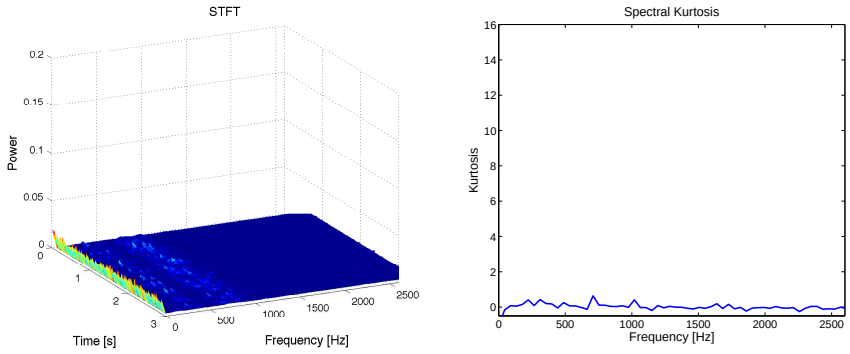


Figure 41: *CM task output with no damage detection*

The CM task has been initially tested with a new and healthy bearing. The performed analysis did not detect any damage returning a negative response as shown in the console output on figure 41. In order to understand this result the STFT of the measured signal is shown in figure 42a. The higher peaks are located in the low frequency region and are related to the shaft rotational frequency. As a consequence the SK computed in the fault detection phase is almost zero in the whole frequency range (Fig. 42b). This means that the signal content is mostly related to stationary signals and noise.



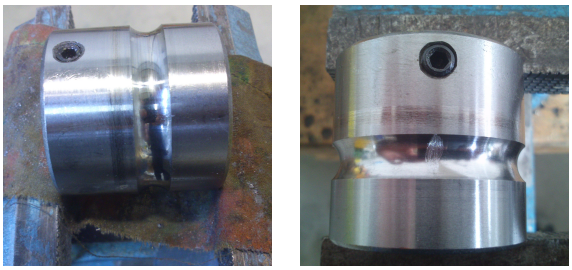
(a)

Figure 42: *STFT (a) and SK (b) of the new bearing*

Damaged Bearing

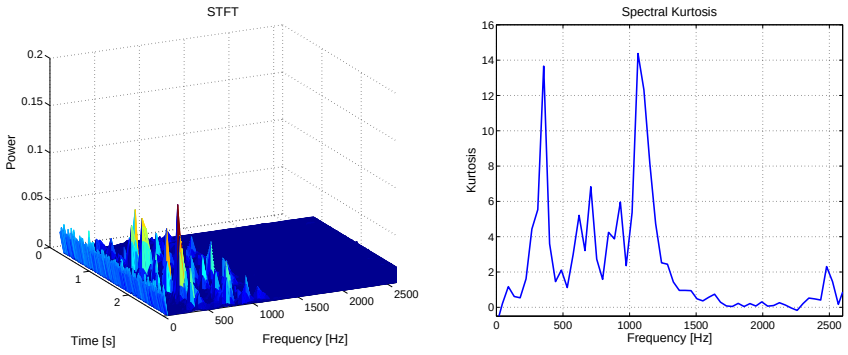
After the first analysis without the fault, the healthy bearing has been disassembled and the inner race has been slightly damaged with a grinding tool (Fig. 43b). Compared to the previous case the STFT of figure 44a shows the onset of peak at medium and high frequencies. These peaks are mainly related to the contribution of damage as proved by the result of the SK analysis

on figure 44b. Looking at the SK plot it seems that the effect of the damage is relevant at several frequencies, however the EA is performed selecting the signal at the frequency with the higher Kurtosis at $1063\,Hz$. The resulting envelope spectrum and the bearing diagnosis is showed in figure 45. The result of the EA allows to explain also the numerous peaks of the SK providing a better insight of the system status. Regarding the bearing diagnostic the most relevant peaks are those one at $1.38\,Hz$ and $17.99\,Hz$. The $17.99\,Hz$ frequency matches the BPFI frequency and then is compatible with the damage on the inner race. The other peak is very close to the FTF frequency, meaning that also the bearing cage is faulty. This is likely since it can have been damaged during the bearing disassembly and reassembly. The highest peak in the envelope spectrum is related to the rotational frequency that can be caused by a misalignment between the motor and the bearing shaft and the irregularities in the motion transmission due to the universal joint. In conclusion, the analysis of the envelope spectrum allowed to identify several damage sources in the test bench, even if the automated diagnostic routine recognized only the most likely.



(a) (b)

Figure 43: *Inner race of the healthy (a) and damaged bearing (b)*



(a) (b)

Figure 44: *STFT (a) and SK (b) of the damaged bearing*

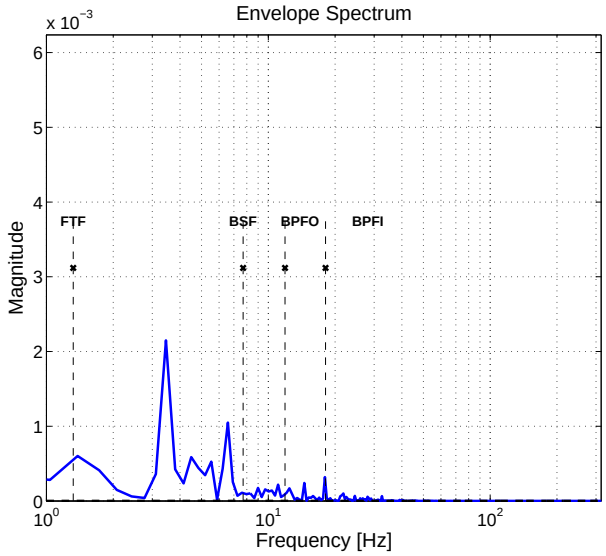


Figure 45: *Envelope spectrum of the damaged bearing showing the BPFI related peak*

3.3 COLLISION DETECTION

During machining operation, collisions can be due to the impact between the workpiece and the tool or even the spindle. The first case entails the tool breakage and the damage of the workpiece while in the second case the machine can be severely damaged. The causes of collision usually lie in programming errors or in maneuvering errors committed by the operator during manual movement of axis. The only available action is the machine halt by means of the emergency push-button on the operator panel. This kind of intervention is however too slow and the collision is most of the time unavoidable. The consequences range form the tool change to the reject of the damaged part and in the worst case in a long production down-time for the inspection and repair of the machine tool. In this context the monitoring system can provide a faster reaction in case of emergency by using the onboard sensors to assess the occurrence of a collision and halt the machine by switching the level of a digital line. It is worth to notice that in this application the *monitor* behaves as a passive safety system since its action is performed after the collision. However with the use of additional sensors, such as proximity sensors, the device can be easily converted to an active safety system. The CD algorithm is enabled both when no monitoring tasks are executed, and then the monitoring system is in the inactive state, both when the VM function is running. Instead if the CM task is executed the CD function is disabled since the accelerometer configuration is different. However this situation is reasonable since the CM activity is performed with the machine in standstill position.

3.3.1 *Collision Detection Algorithm*

The assessment of a collision occurrence is performed in the following steps:

VIBRATION MEASUREMENT Vibrations are collected for each accelerometer and at a sampling rate of $5\,kHz$.

SIGNAL PROCESSING Acquired data are compared with a predefined limit.

INTERVENTION The detection of a collision causes the activation of the emergency digital line.

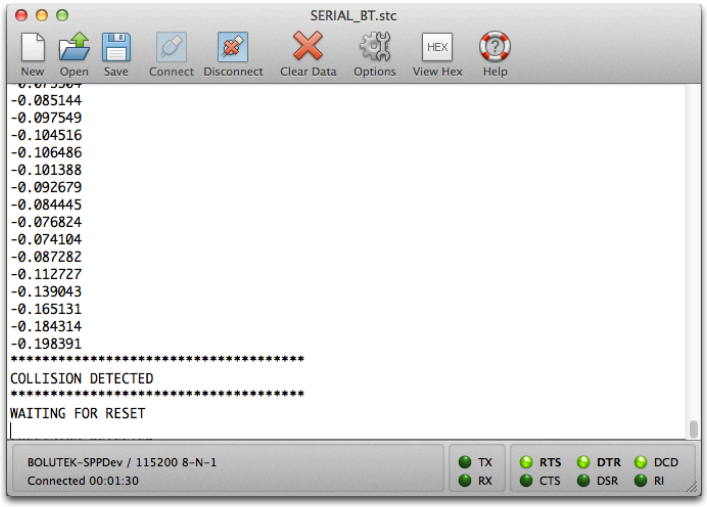


Figure 46: *Message sent after collision detection*

The CD algorithm measures the resultant energy of vibrations on the three axes components computed inside a narrow window of around $13\,ms$. The computed value is expressed in integer math to avoid the increase of CPU overhead related to the use of single precision values (line 11 of code 7). The signal energy is compared with the safety limit and if the value exceeds it the digital line intended for the machine halt is activated by switching its logical level (line 15). The evaluation of the vibration energy content is performed at each sample time and then the system provides the fastest available reaction. At this point the machine is stopped, a warning is sent to the MMI and the monitoring system is set in the error state waiting for the operator intervention (Fig. 46).

```
/*COLLISION DETECTION*/
void CD_run()
{
        uint32_t tmp;
        for (uint8_t i=0; i<N_ACC; i++){
                getAcc_raw(i, (unsigned char *) &rawData);
                tmp = ((uint32_t) rawData.aX * (uint32_t) rawData.aX)
                        + ((uint32_t) rawData.aY * (uint32_t) rawData.aY)
                        + ((uint32_t) rawData.aZ * (uint32_t) rawData.aZ);
                /*SIGNAL ENERGY*/
                cd_val[i]+= tmp - int_cd_vals[cd_index &(CD_BUFFER_SIZE - 1)][i];
                /*COLLISION ASSESSMENT*/
                if (cd_val[i] > CD_LIMIT){
                        /*SET EMERGENCY PIN HIGH*/
                        COLLISION(ALARM_PIN);
                        /*RESET FLAGS*/
                        cd_isr = 0;
                        cd_detected = 1;
                }
                /*UPDATE BUFFER*/
                int_cd_vals[cd_index & (CD_BUFFER_SIZE - 1)][i] = tmp;
        }
        /*UPDATE BUFFER INDEX*/
        cd_index++;
        /*RESET THE INDEX VALUE*/
        if(cd_index >= 10*CD_BUFFER_SIZE){cd_index = 0;}
}
```

Listing 7: *Source code of the collision detection algorithm*

3.4 OFF-LINE EVENT LOGGER

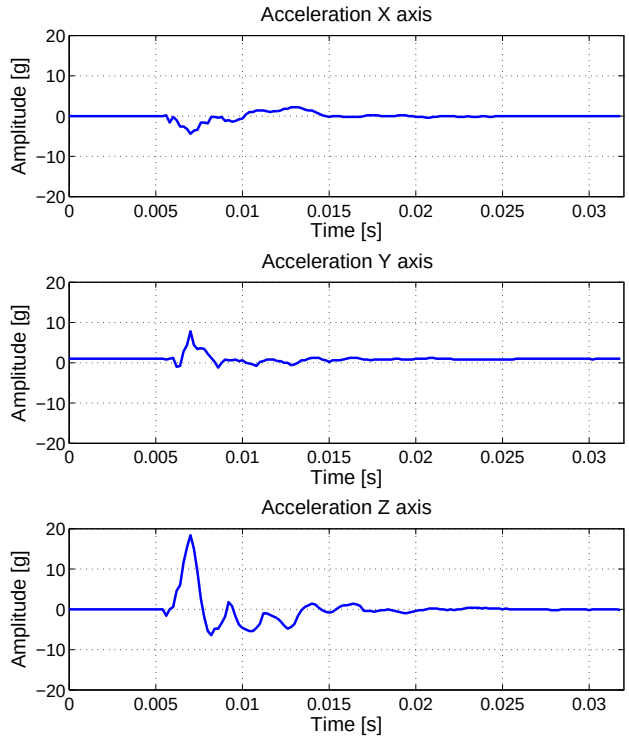


Figure 47: *Detection of impulsive event in sleep mode*

The *monitor* node equipping the accessory heads has to perform a further task in addition to those described in the previous sections. Accessory heads can be damaged not only during machining but also during the phase of coupling or release from the machine tool. In this regard, it is needed that the *monitor* node on the accessory head can detect and record the occurrence of potentially detrimental events, even if the head is not being used. In this case *monitors* have to operate detached from the machine and then the required power supply is provided by a backup battery. As a consequence the monitoring task can not be continuously run in order to avoid an excessive power consumption. Given this requirement, the *monitor* node in non-operative conditions is activated only when a significative event occurs. When the accessory head is placed on the tool storage and released from the machine tool, the NC commands the *monitor* to disconnect. The power supply is switched from the machine power to the backup battery and the microcontroller is set in sleep mode. Program execution is halted and all the clock sources are stopped, minimizing the power consumption. In this configuration the microcontroller can be reactivated only by interrupts on the TWI line and on specific I/O pins (asynchronous ports). The

connected accelerometers are then used to wake up the *monitor* exploiting their functionality to generate event-driven interrupt signals. Since the microcontroller is disabled, sensors are configured to acquire data and store them in an embedded ring buffer. In addition an internal threshold is set to trigger an interrupt if the acceleration values exceed it. Therefore if a severe event, such as a collision, is detected the accelerometer immediately wakes the microcontroller. At the wake moment the *monitor* starts reading the stored data gradually emptying the buffer. In this way also the acceleration measured before the interrupt are acquired allowing the reconstruction of the entire triggering event. Unlike the previous monitoring tasks, in this case the acquisition is timed by one of the microcontroller timer rather than by the accelerometer, since it has a different interrupt configuration. When the microcontroller wakes the interrupt routine is initiated and the SPI buffer is initialized in order to communicate with the sensors. Then the timer is configured to set the sampling fequency at $5\,kHz$ (line 6 of listing 8). After the configuration phase the *monitor* starts transferring the data from the accelerometer buffer (line 15). The acquisition is executed for a limited amount of time (few milliseconds) and at the end the recorded signal is stored in an external memory together with additional information on the occurred event. At the end of acquisition the program re-enables the sleep mode and waits for a new event. The result of an example test is shown in figure 47 where the accelerometer has been hit along the z direction. Plots show the recorded signals after the microcontroller wake proving that the *monitor* node was capable to detect the impulsive event.

```
/*MCU WAKE INTERRUPT CALLBACK FUNCTION*/
void wake_callback()
{
        /*ENABLE TIMER FOR ACQUISITION AFTER WAKEUP*/
        PR.PRPC &=  ~SYSCLK_TC1;
        /*SET OVERFLOW EVERY 5 kHz*/
        TCC1.PER = 6400;
        TCC1.CTRLA = TC_CLKSEL_DIV1_gc;
        /*CONFIGURATION*/
        wake_data = &data_array[0][0];
        uint16_t k = 0;
        /*DISABLE LIS3DH INTERRUPT*/
        clear_int(LIS3DH_DEVICE_0);
        /*START ACQUISITION*/
        while (k<WAKE_BUFFER){
                /*WAIT TIMER OVERFLOW*/
                while ((TCC1.INTFLAGS & TC1_OVFIF_bm) == 0){}
                /*CLEAR INTERRUPT FLAG*/
                TCC1.INTFLAGS = TC1_OVFIF_bm;
                ioport_toggle_pin(DEBUG_PIN0);
                /*GET AND STORE DATA*/
                getAcc_raw(LIS3DH_DEVICE_0,&rawData);
                *wake_data++ = rawData.aX;
                *wake_data++ = rawData.aY;
                *wake_data++ = rawData.aZ;
                k+=3;
                }
        /*RESET DATA POINTER*/
        wake_data = &data_array[0][0];
        /*ACQUISITION TERMINATED RETURN TO THE MAIN ROUTINE*/
        ...
}
```

Listing 8: *Microcontroller wake event source code*

3.5 EVALUATION BENEFITS AND DRAWBACKS

Global Monitoring System

The purpose of the proposed monitoring system is to equip CNC milling machines in order to supervise the working operations and assess the status of the machine tool. The requirements that must be fulfilled aim to ensure the proper operation of the system in several industrial applications. The monitoring system should be able to adapt to different working conditions without the need of detailed handmade tuning during the process execution. This is a key aspect that has been considered since a time-consuming system configuration can void the benefits of adopting the system itself. Moreover a complex setup will require additional skills to the machine user that on the other hand is not supposed to know the underlying working principles. In addition the whole system, including sensors and the data acquisition and processing units has to be fully integrated in the machine tool architecture sending the monitoring information through the MMI. The developed system provides a framework where the monitoring nodes are deployed on the machine according to the specific task. This allows to build a scalable system that easily fits different machine tools and applications and allows to expand it if new task are required or if the machine modifies its architecture as in the case of the accessory heads mounting. The choice of microcontrollers as monitoring devices and the use of digital MEMS sensors provide an inexpensive and small-sized solution compared to common monitoring systems that usually consists of signal conditioning units, data acquisition boards and a computer to execute the process control. The microcontroller-based design allows to assign a specific task to each monitoring node providing high flexibility on the setup of the supervisory system. On the other hand the use of microcontrollers represents also the main limitation of the proposed approach because of the reduced performances in term of memory and computational power. However thanks to the advances in electronics this issue can be easily overcome by using more powerful devices with slight cost increase. Since each node is connected to a standard TWI bus the whole monitoring system can be constituted by microcontrollers with different performances according to the performed task. The core software architecture based on a state machine paradigm requires very few inputs by the machine user that has mainly to select the monitoring activity to be executed. However the reduced interaction possibilities does not prevent from further extension of the system functionalities. In this sense also the software architecture is characterized by high flexibility and modularity that can be exploited during the design phase.

Vibration Monitoring Algorithm

The VM algorithm based on the RMS indicator, revealed to be the most efficient method to monitor the vibration level on the machine acquiring data from up to four accelerometers at high sampling rate. This implementation provides information on the actual machining conditions and can be used

as a support for the machine operator, by sending warning messages when an excessive vibration level is sensed suggesting the corrective action. In addition the developed algorithm allows also minimal attended working operation by automatically adapting the feed rate in order to keep the measured vibration index within a safety range. In this case the supervisory system does not require any interaction with the machine operator and ensures that the process is performed in the desired manner regardless the external factors that can affect the working operation. According to the need of adapting the system to different working conditions a learning function has been implemented in order to configure the intervention threshold of the monitoring algorithm. In this case the machine operator has to bring the machine to the desired working point and then execute the learning function. In this way the system automatically adjusts the configuration parameters and then starts the monitoring task. The existing tradeoff with the proposed monitoring algorithm is its reactive behavior, meaning that the corrective action is not optimized with respect to the actual working conditions. Therefore, since the controlled parameter is the feed rate, a too conservative response can be penalizing for the productivity of the machine. However the use of this system provides an enhancement of the machining operation by keeping the process execution under nominal conditions that, at the same time, affects the quality of the machined part and the tool life.

Condition Monitoring Algorithm

The proposed CM approach for the detection and diagnosis of faulty spindle bearings is based on the well known Envelope Analysis of bearing vibration signature. It is then a reliable tool that returns a detailed and clear information on the system health. In addition the fault detection algorithm based on the Spectral Kurtosis computation provides a fast and accurate method to evaluate the existence of a damage without requiring complex analysis of the vibrations spectra. However detailed information on the acquired data are available for maintenance engineers. The results obtained from the CM algorithm require in any case a history of faults in order to properly estimate the entity of damage and consequently to prepare the maintenance plan. The main drawback of the proposed algorithm is the need of a significative computational effort to perform the analysis and this cause a slow response of the system. However since the CM task can be run periodically, the required time does not affect the productivity of the machine. In addition this issue can be easily overcome by adopting a more powerful microcontroller.

Collision Detection Algorithm

The CD task does not require any interaction with the machine operator except when, after a collision, it brings the system to the error state waiting for the reset. The implementation is based on the comparison between the signal energy and a threshold. This represents the main limitation of the algorithm since setup of the threshold can be far from trivial since false alarms

can cause unexpected halt of the machine with subsequent slowdowns of the production. On the other hand the proposed algorithm can be tuned also to detect particularly severe working conditions and halt the machine if the action of the VM monitoring is not enough to restore an acceptable vibratory level. This provides a more robust response of the monitoring system guaranteeing the safety of the machining process.

4

ADVANCED MONITORING AND CONTROL

This chapter describes the use of the Extremum Seeking control strategy for the adaptive optimal tuning of machine parameters. First it will be presented the inline performance optimization of a linear motion system in terms of accuracy and disturbance rejection. The control of the linear motion system will be enhanced by a variable gain architecture in order to overcome the intrinsic limitations of linear controllers. Then the Extremum Seeking will be included on the system in order to optimize the variable gain controller according to the characteristics of the measured error. The proposed method has been successfully tested on an experimental linear motion system based on a magnetically levitated axis and the results will be presented in the following chapter.

4.1 INTRODUCTION

Linear controllers such as the Proportional Integrative Derivative (PID) controllers, are nowadays the most frequent solution for the control of linear motion system. The ease of design, tuning and implementation and, above all, the capability of manage model uncertainty make linear controllers very suitable for a large variety of applications. However linear controllers designer has to deal with inherent performance limitations that impose a trade-off among conflicting features of the desired controller response. This is the case of the waterbed effect that causes the increase of system sensitivity at high frequencies, as a consequence of the attenuation of low frequency disturbances. This is a common issue on linear controller design that can not be overcome by using more advanced design techniques. Therefore in the controller design the choice of parameters has to balance the fulfillment of the required performance on one hand, and the reduction of the disturbances effect on the other hand. The design limitations related to the waterbed effect are a key issue in linear motion systems, where the improvement of the positioning accuracy is affected by the contribution of measurement noise and disturbances at high frequencies. In order to reduce these limitations a Variable Gain Controller (VGC) approach has been proposed. The VGC strategy allows to selectively increase the controller gain according to the magnitude of the feedback error, improving the system response both in terms of reduction of tracking error and sensitivity to disturbances. The tuning of the VGC parameters depends on the characteristic of the error signal and of the noise affecting the system. Usually this task is performed in an heuristic way or by using a disturbance model to optimize the controller performance for a specific application. However the identification and modeling of the disturbances is often difficult and can not entirely capture the actual effects acting on the system. A different approach has been proposed

in order to tune automatically the variable gain controller parameters by using an extremum seeking control strategy. Extremum seeking is an adaptive control technique that optimizes a certain performance measure according to the steady state output of the system in real-time, by automated and continuous adaptation of the system input-output pair. This method only uses the measured output of the plant to compute the steady-state performance regardless of the knowledge on the disturbances and on the system dynamics. The variable gain controller and the extremum seeking strategy has been applied to an experimental setup that consists on a magnetically levitated linear axis controlled on 6 degree of freedom. The activities that have been carried out, involved the design of both the linear and variable gain controller, the assessment of the tradeoff due to the waterbed effect and finally the performance optimization of the variable gain controller by means of the extremum seeking technique. This work has been conducted in collaboration with the University of Eindhoven and the Philips Innovation Services.

4.2 PROBLEM FORMULATION

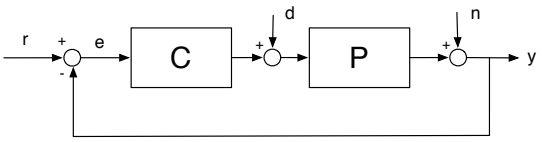
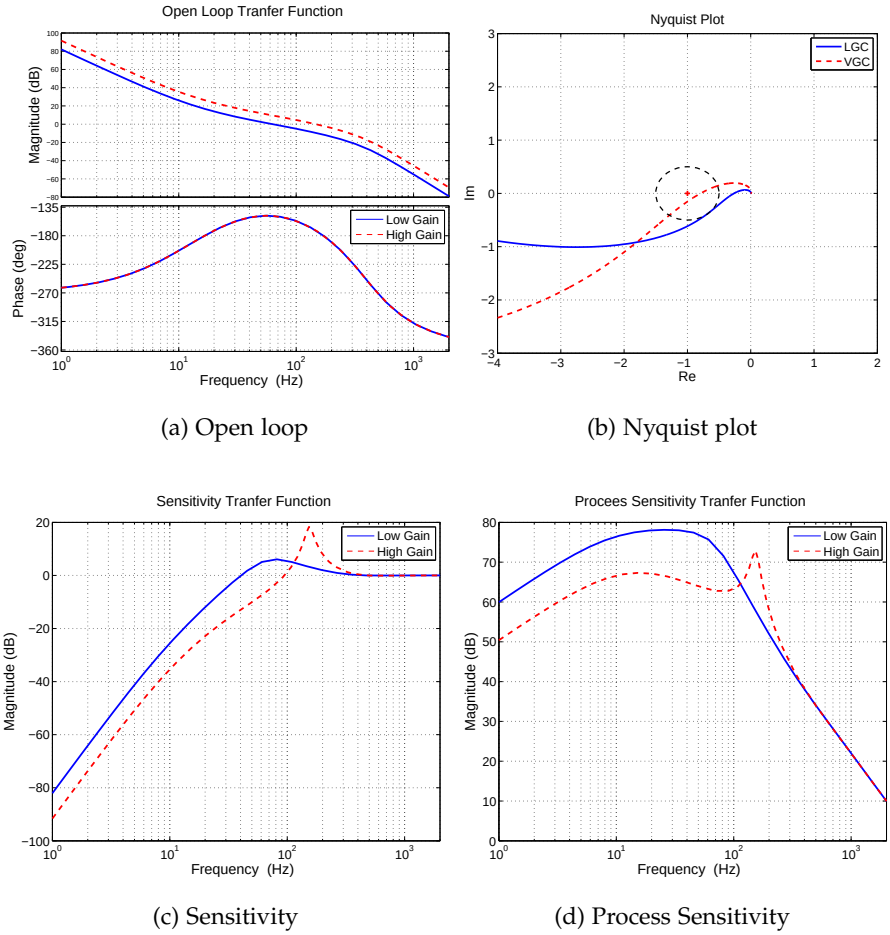


Figure 48: *Linear feedback controller scheme*

Considering the control of a linear motion system, the closed loop structure shown in figure 48 consists of the plant $P(s)$ (modeled as a mass) and the linear controller $C(s)$. The reference position r is sent through the controller to the plant that receives as inputs a force signal and the force disturbance d. The plant output is a position signal that is fed back providing the positioning error e. The control performance can be improved by increasing the controller gain and consequently the bandwidth of the system. The effect of gain increase can be seen in figures 49 where it is shown how the system response varies with an high gain configuration (red dashed line) compared to a low gain configuration (blue solid line). The high gain controller has a larger bandwidth that reduces the error sensitivity and improves the disturbance rejection within the bandwidth (Fig. 49c and 49d). However the high frequency noise in the range $100 - 200\ Hz$ is amplified because of the waterbed effect (cf. §A.1) as can be seen in figure 50. The plot shows the positioning error obtained for a $10\ mm$ back and forth motion of the mass (low frequency reference) that is followed by the error during the standstill position where a $160\ Hz$ sinusoid is used as disturbance input (high frequency input). The controller with the higher gain provides a better attenuation of the error during the motion phase than the low gain controller but also the disturbance results amplified by the increased gain. A more effective way to overcome this limitation is given by the variable gain con-



(a) Open loop

(b) Nyquist plot

(c) Sensitivity

(d) Process Sensitivity

Figure 49: *Comparison between the transfer functions of low and high gain controller configuration*

trol (or *N-PID* control) strategy [107]. The basic idea is to add a non linear function ϕ to the existing linear controller in order to selectively increase the gain according to the controller error. The close loop architecture of figure 48 is then modified with an other loop that consists of a nonlinear element $\phi(e)$ function of the error and a shaping filter $F(s)$ (Fig. 51). The selected shape of nonlinearity $\phi(e)$ is the dead-zone function

$$\phi(e,\delta) = \begin{cases} 0 & \text{if } -\delta \leq e \leq \delta \\ \alpha(e-\delta) & \text{if } e \leq -\delta \\ \alpha(e+\delta) & \text{if } e \geq \delta \end{cases} \tag{20}$$

that applies the extra gain α if the error exceeds the dead-zone length δ. For linear motion systems low frequency inputs (e. g. the reference singal) cause larger errors than those ones due to high frequency inputs (e. g. measurement noise). Therefore the dead-zone function of equation 20 compensates the waterbed effect and increases the overall controller performance by improving the tracking property at low frequencies and significantly reducing the sensitivity to high frequency disturbances. Given the architecture of fig-

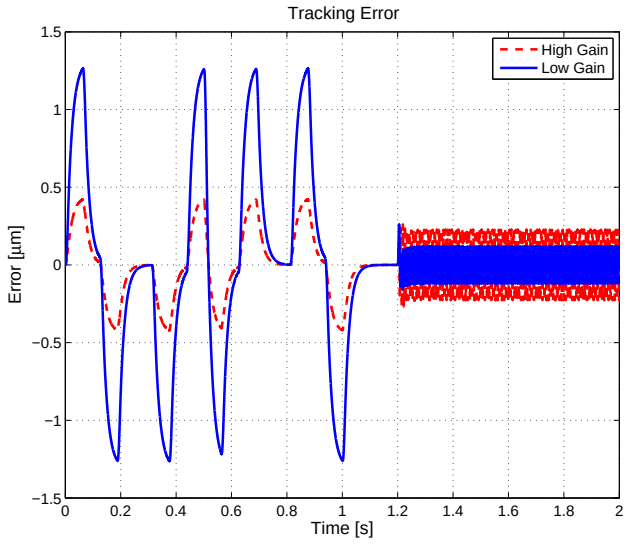


Figure 50: *Error signal with two different configuration of the controller*

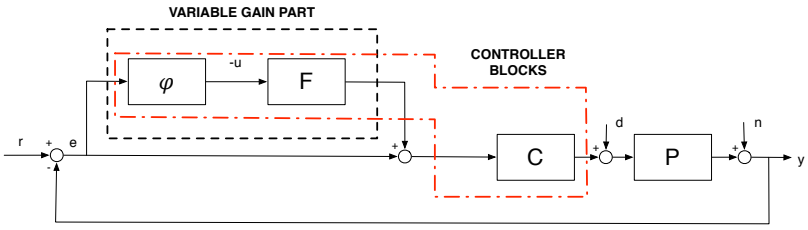


Figure 51: *Variable gain controller scheme*

ure 51, the VGC can be modeled as a *Lur'e* type system (Fig. 52) of the form:

$$\begin{cases} \dot{x}(t) &= A\ x(t) + B\ u + B_r\ r(t) + B_d\ d(t) \\ e &= C\ x(t) + D_r(t)\ r + D_d\ d(t) \\ u &= -\phi(e, \delta) \end{cases} \tag{21}$$

where the non linear part is the dead-zone function ϕ and the linear part is the transfer function between the dead-zone output u and the error e:

$$G_{eu} = C(s\ I - A)^{-1}\ B = \frac{P(s)C(s)F(s)}{1 + P(s)C(s)} \tag{22}$$

The stability of the nonlinear controller is assessed by satisfying the circle criterion conditions and the incremental sector condition for the convergence of solution (cf. Appendix A.2 and [108]). According to those definitions, the nonlinear system is globally asymptotically uniformly stable if:

- $G_{eu}(s)$ is *Hurwitz*: verified since both $\dfrac{PC}{1+PC}$ and F are *Hurwitz* by design.

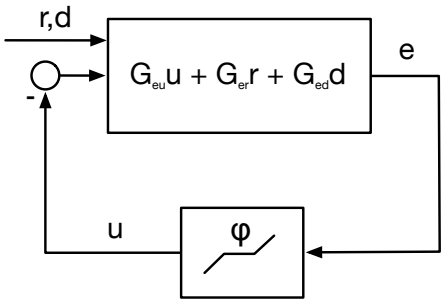


Figure 52: Lur'e *scheme of the controller*

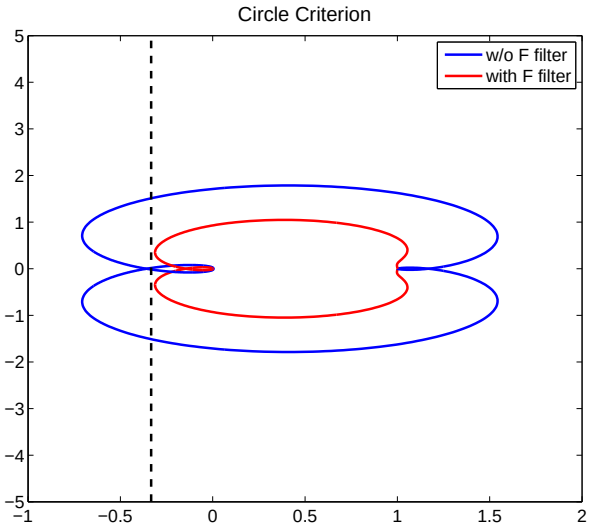


Figure 53: *Nyquist plot of* $G_{eu}(s)$

- $Re(G_{eu}(j\omega)) > -\frac{1}{\alpha}\,\forall\omega \in \mathbb{R}$: can be achieved by tuning F.
- $1 + \alpha G_{eu}(\infty) > 0$: verified since $\lim_{s\to\infty} \frac{PC}{1+PC} = 0 \implies 1 > 0$.
- $\phi \in [0, \alpha]$: is true for the dead-zone function.
- $0 \leq \dfrac{\phi(e_2) - \phi(e_1)}{e_2 - e_1} \leq \alpha,\ \forall e_2 \neq e_1$: is true for the dead-zone function.

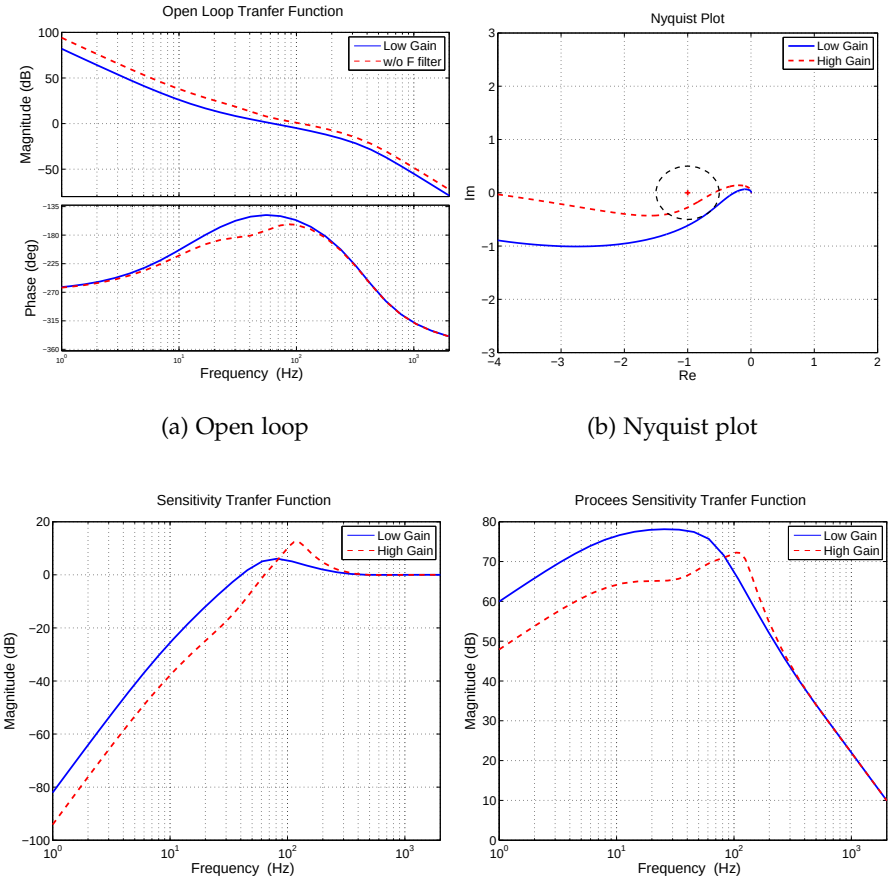
From the stability conditions two important conclusions can be drawn:

1. The stability of the VGC, given the linear controller $C(s)$ is obtained by a proper design of the shaping filter $F(s)$.
2. The stability is not affected by the dead-zone length δ

It follows that, the design of $F(s)$ aims to satisfy the circle criterion condition $Re(G_{eu}) > -\frac{1}{\alpha}$ as shown in figure 53 and it is needed to guarantee the stability of the system. On the other hand, the dead-zone amplitude is the key parameter for the achievement of high controller performance since it

allows to change the system sensitivity according to the error magnitude. The nonlinear controller is given by equation 23, where $\kappa = 0$ if the error amplitude is smaller than the dead-zone length, while $\kappa = \alpha$ if the error is greater than δ.

$$(1 + \kappa F(s))C(s) \tag{23}$$



(a) Open loop

(b) Nyquist plot

(c) Sensitivity

(d) Process Sensitivity

Figure 54: *VGC transfer functions for the low gain configuration* $\kappa = 0$ *and the high gain configuration* $\kappa = \alpha$

When the error is within the dead-zone length the additional gain in zero and the system has the same response of the linear case with the low gain configuration (blue solid line in figure 54). In the opposite case, thanks to the extra gain, the system has a reduced sensitivity to error and disturbances (red dashed line in figure 54). The VGC response combines the advantages of the two configurations according to the dead-zone value. The optimal δ value strictly depends on the disturbances acting on the system and then the proper dead-zone length has to be selected after the assessment of the noise level and the identification of disturbances in the analyzed application. Considering the previously described error signal (Fig. 50), it is evident that

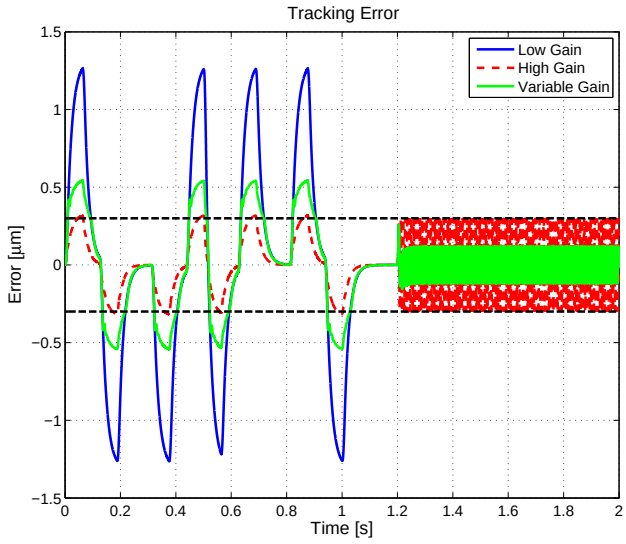


Figure 55: *Comparison among the errors of the linear and VGC controllers*

a more desirable response of the controller would be a mix between the improved error tracking of the high gain controller, during the motion phase, and the noise attenuation of the low gain controller in the standstill position. This result can be achieved by setting the dead-zone length equal to the maximum value of the disturbance error for the high gain controller. The response of the VGC with this configuration is showed in figure 55 where the black dashed line is the dead-zone amplitude and the green solid line is the error of the VGC. The system response is a tradeoff between the low and high gain configurations that leads to an overall improvement of the performance. The tuning of δ parameter is usually performed in an heuristic manner or through the offline optimization of the VGC, relying on models of the plant and of the external disturbances. However this approach requires an accurate modeling of the disturbances that is usually a challenging task. To avoid this issues in the next section will be proposed an online optimization method based on the Extremum Seeking (ES) control strategy to find the optimal dead-zone length according to a pre-established performance criterion. The ES is an adaptive control approach that does not require a precise knowledge on the system and on the disturbances and allows to continuously operate with the optimal configuration of the system. The proposed method is particularly suitable for the calibration of linear motion systems, such as the axes of machine tools, in order to achieve the best positioning accuracy. In addition the ES allows to maintain the desired performance even if the operative conditions slowly vary in time, compensating for example the effects of temperature variations or wear.

4.3 EXTREMUM SEEKING

ES is an adaptive control strategy used to optimize the steady state input-output characteristic or the performance of a dynamic system. This technique requires the existence of an input-output relationship with at least one extremum but does not need any explicit knowledge about the system to optimize. The algorithm provides a gradient estimate of the steady state input-output map and then the direction towards the optimum point. ES is generally used for optimization of plants with constant steady state outputs. However recently it has been adopted also to optimize systems with time varying outputs and in particular with periodic steady state outputs [109]. Linear motions systems are often applied in repetitive tasks, so this last method is particularly suitable for optimize the system performance understood as minimizing the tracking error and the contribution of external disturbances to it. The general architecture for the ES algorithm is showed in figure 56. The output of the system is used for the computation of a performance index J that multiplied by a dither signal (usually a sinusoidal signal) provides gradient estimation of the static map in a neighborhood of the given input-output pair. This estimate is used in the optimization block to move the input value to the direction that minimize (or maximize) the performance. According to the explanation provided by Tan et al. work

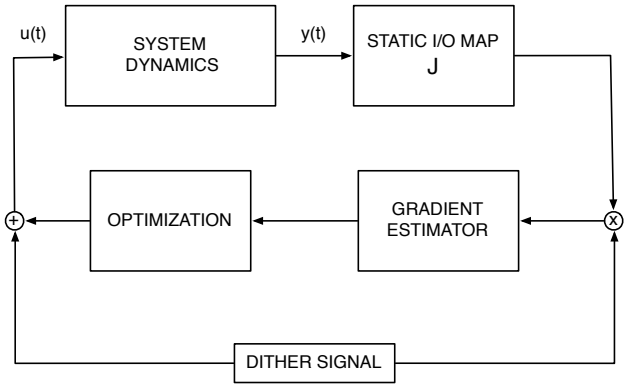


Figure 56: *Extremum seeking main blocks*

[110] let us consider a system whose dynamics can be modeled as:

$$\begin{cases} \dot{x} &= f(x,u) \\ y &= h(x) \end{cases} \tag{24}$$

Let define the static performance map J as:

$$J = g(u) = g(h(x)) \tag{25}$$

with an extremum located at u^* (Fig. 57). We further assume that there exists a steady state characteristic defined by a differentiable function $l : \mathbb{R} \to \mathbb{R}^n$ such that

$$f(x,u) = 0, \quad iff \quad x = l(u) \tag{26}$$

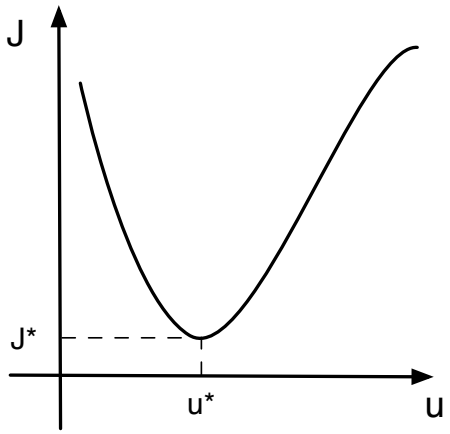


Figure 57: *Static performance map*

and the equilibrium point $x = l(u)$ is globally uniformly asymptotically stable in u. These assumptions ensure that in steady state conditions the system performance is given by:

$$J = g(h(x)) = g(h(l(u))) = J(u) \tag{27}$$

4.3.1 *Time Scale Separation*

The simplest ES algorithm used to optimized the performance to the extremum value J^* is shown in figure 58. In this configuration the parameters that rule the behavior of the algorithm are:

- The amplitude of the dither signal a.
- The frequency of the dither signal ω.
- The gain of the integrator c.

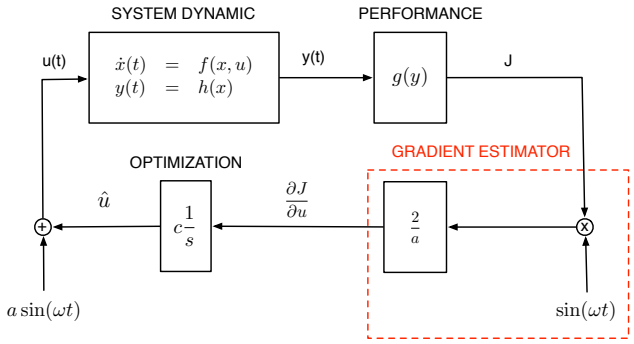


Figure 58: *Minimal extremum seeking block diagram*

According to this scheme the dynamics of the system can be described by:

$$\begin{cases} \dot{x} &= f(x, \hat{u} + a\sin(\omega t)) \\ \dot{\hat{u}} &= c\,\frac{2}{a}\,J\ \sin(\omega t) = c\,\frac{2}{a}\,g(h(x))\ \sin(\omega t) \end{cases} \tag{28}$$

These equations represent respectively the x dynamics and the learning dynamics of the ES algorithm. Assuming that the transients in the x dynamics are faster than the variations in time of both $\hat{u}$ and the dither signal, than the solution x can be expressed as:

$$x = l(\hat{u} + a\sin(\omega t)) + \xi(t) + \eta(t) \tag{29}$$

where $\xi(t)$ is the transient response of the dynamical system and $\eta(t)$ is a function of a, ω, c that can be made neglectable by choosing them sufficiently small. Then, considering also the equation 27, the learning dynamics can be expressed as:

$$\dot{\hat{u}} = c\,\frac{2}{a}\,J(\hat{u} + a\ \sin(\omega t) + \xi(t) + \eta(t))\ \sin(\omega t) \tag{30}$$

These equations highlight the time scale separation determined by the three dynamics existing in the ES algorithm:

FAST DYNAMICS It is given by the transients of the system states x described by the term $\xi(t)$

MEDIUM DYNAMICS It is the gradient estimate dynamics due to the dither signal and then is related to its frequency ω.

SLOW DYNAMICS The slowest evolution of the system is ruled by the convergence dynamics that is related to the learning rate c.

The rationale behind the time scale separation can be understood considering that the ES algorithm aims to find the extremum point of the input-output characteristic in their steady state condition. This means that the whole ES dynamics must be slower that the system dynamics so that, when a new input $u = \hat{u} + a\sin(\omega t)$ enters in the system, the transient runs out without affecting the ES action. In addition, reaching the extremum point requires that the gradient estimation and the optimization are performed in sequence. Therefore first the dither signal has to explore the neighborhood of the current input estimate $\hat{u}$, in order to assess the gradient direction and, subsequently, the learning dynamic slowly moves towards the extremum u^*. The time scale separation is an essential property that ensures the convergence of the algorithm. If the set of parameters a, ω, c is chosen sufficiently small, the time scale separation among the plant, the gradient estimator and the convergence dynamics is preserved and then the steady-state measured performance will remain close to the static performance map.

4.3.2 *Gradient Estimator*

The main idea behind the gradient estimator is that the product between the variation of J and the variation of u is similar to the slope of the $J - u$ curve in the neighborhood of u. In other words if J increases when u increases than the slope will be positive. Otherwise if J decreases when u increases the gradient will be negative. In both cases the slope will be close

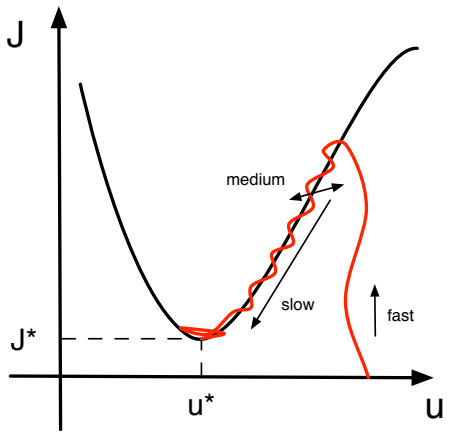


Figure 59: *Time scale separation and convergence*

to $J(\hat{u} + a\sin(\omega t))\sin(\omega t)$. In order to demonstrate this general concept let us consider the approximate gradient equation:

$$\widetilde{\frac{\partial J}{\partial u}} \approx \frac{2}{a} J(\hat{u} + a\sin(\omega t) + \xi(t) + \eta(t))\sin(\omega t) \tag{31}$$

Since $\xi(t)$ quickly goes to zero and $\eta(t)$ is small if a, ω and c are small, then those terms can be neglected. The Taylor expansion up to the first derivative of $J(\hat{u} + a\sin(\omega t))$ gives:

$$\begin{aligned}
\widetilde{\frac{\partial J}{\partial u}} &\approx \frac{2}{a}(J(\hat{u}) + a\frac{\partial J}{\partial u}\sin(\omega t))\sin(\omega t) + O(a) \\
&\approx \frac{2}{a} J(\hat{u})\ \sin(\omega t) + 2\frac{\partial J}{\partial u}\sin^2(\omega t) + O(a) \\
&\approx \frac{2}{a} J(\hat{u})\ \sin(\omega t) + \frac{\partial J}{\partial u}(1 - \cos(2\omega t)) + O(a)
\end{aligned} \tag{32}$$

This result not seems to provide a good estimate of the gradient, but averaging its response over the time gives:

$$\frac{1}{T}\int_{t}^{t+T} \widetilde{\frac{\partial J}{\partial u}}\, d\tau = \frac{\partial J}{\partial u} + O(a) \tag{33}$$

According to equation 33, the accuracy of the gradient is related only to a and then the smaller is the dither amplitude the better will be the gradient estimate.

4.3.3 *Optimizer*

The average behavior described in the previous section allows to define the average learining dynamic as:

$$\dot{\tilde{\hat{u}}} = c\,\widetilde{\frac{\partial J}{\partial u}} = c\left(\frac{\partial J}{\partial u} + O(a)\right) \tag{34}$$

The value of the dither amplitude a and the learning rate c are responsible for the accuracy of the Extremum Seeking algorithm since they define the convergence domain size for the performance J and the dead-zone amplitude $delta$. By selecting a sufficiently small, the J value will converge to a $O(a)$-neighborhood of J^*. Furthermore, $\hat{u}$ remains in a $o(c)$-neighborhood of $\widetilde{\hat{u}}$. Then the convergence of $\hat{u}$ and $y(t)$ will be defined in a $O(a) + o(c)$ limited domain surrounding respectively u^* and y^*.

4.3.4 *Extremum Seeking with Periodic Outputs*

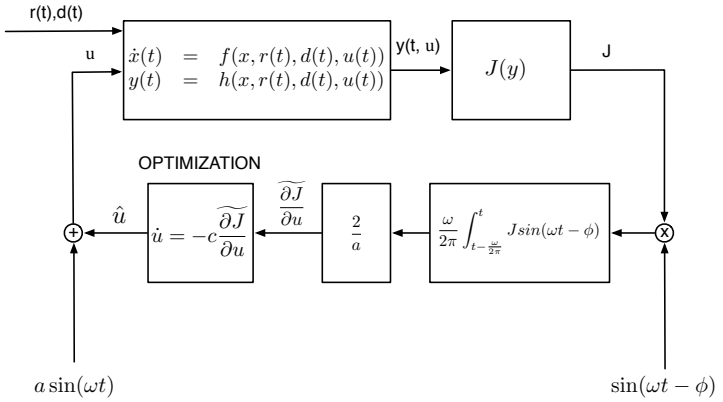


Figure 60: *Extremum seeking algorithm with periodic outputs*

Many engineering systems execute repetitive tasks and this implies working conditions with periodically varying steady state outputs. Linear motion systems such as pick and place machines, wafer scanners and machine tools are applications where appears this kind of behavior and where the extremum seeking algorithm can reveal very useful to improve their performance. However so far, the extremum seeking optimization has been used mostly in plants with constant outputs but in the past few years the attention moved also to systems with periodic steady state outputs. In [109] it has been proposed a novel extremum seeking method applied to generic non linear systems with periodic steady states outputs that is suitable for a motion control context. Given the plant:

$$\begin{aligned} \dot{x}(t) &= f(x, r(t), d(t), u) \\ y(t) &= h(x(t), d(t), u) \end{aligned} \tag{35}$$

where the system dynamics depends on the disturbance $d(t)$ and the parameter u. Moreover the output $y(t)$ is periodic with period T. Let us consider that the performance of the plant (35) is optimized by the extremum seeking scheme of figure 60 acting on the parameter u. In [109] it has been demonstrated that, if the following assumptions are valid:

1. The input pair $r(t)$, $d(t)$ is bounded and periodic with a constant T period.

2. For every fixed parameter u the system (35) is globally asymptotically stable and its steady state solution $x(t,u)$ is unique and $T-$periodic.

3. The steady state performance map J has an unique global minimum (or maximum) for $u = u^*$.

Then the extremum seeking scheme is semi-globally practically asymptotically stable (SGPAS). This stability condition ensures that the optimization loop converges to an an arbitrarily small neighborhood of u^* by properly selecting a sufficiently small set of the dither amplitude a, the dither frequency ω and the learning rate c. Then, whatever the initial conditions, the parameter u will converge to the optimal point u^* with the desired accuracy (and speed), determined by the choice of a, ω and c. Moreover in the same work it has been also proposed the use of the moving average filter of equation (36) that provide a more accurate estimation of the gradient $\frac{dJ}{du}$.

$$\frac{\omega}{2\pi} \int_{t-T_d}^{t} J \sin(\omega\tau + \phi) d\tau \tag{36}$$

The filter deletes the oscillations due to the dither signal and provide an estimate of the average slope of the performance in a neighborhood of u as explained in section 4.3.2. In this gradient estimator is also introduced the fourth parameter ϕ that can be used to avoid the phase shifts due to the system dynamics and the performance computation. In the following sections we will always refer to this method for the estimation of the gradient.

4.3.5 *Variable Gain Controller Optimization*

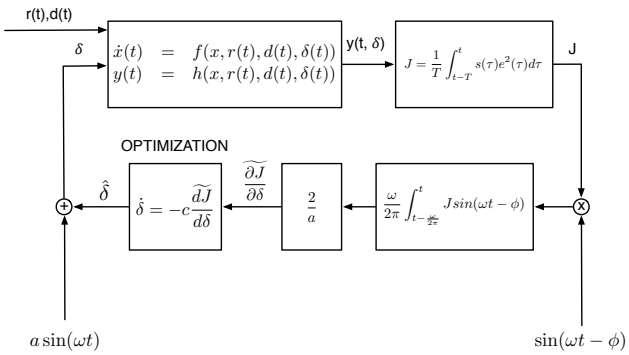


Figure 61: *Extremum seeking algorithm block diagram for linear motion control*

The use of a VGC showed in section 4.2 gives a considerable improvement of the system accuracy. The controller and the nonlinearity can be tuned to reach the required performance for a desired task. Usaually the parameter choice is done in a heuristic way, but this approach can not be always valid since a nominal operative condition is affected by a set of disturbances (such as noise, thermal variations, wear and so on) that changes the system response. The extremum seeking algorithm can be added to the controller

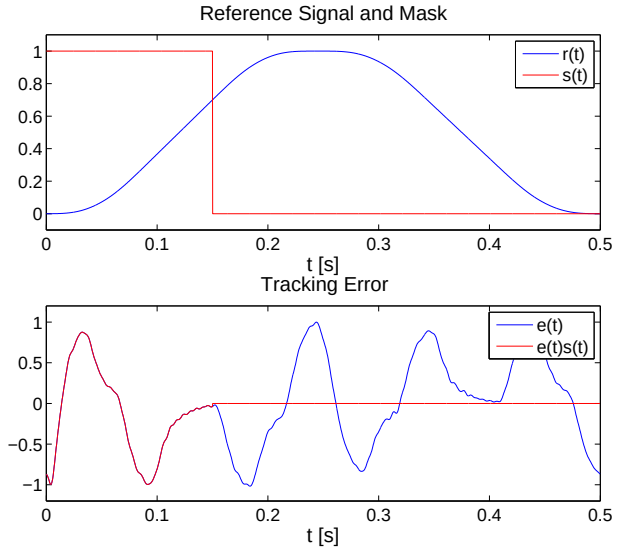


Figure 62: *Mask application in the wafer scanner example*

loop in order to keep the system performance as close as possible to the optimal level, independently from the external factors that affect the system behavior. This is possible since the extremum seeking functioning is related only to the measured system output and then to its actual working conditions. As long as the input-output relationship does not change and there exist an extremum point in the system performance the algorithm will converge on it despite the disturbances acting on the system. In [111] this approach has been proposed for the performance optimization of the motion control of wafer stages. The presented method uses an extremum seeking algorithm to find the dead-zone amplitude that minimizes the cumulative error of the linear motion system. The extremum seeking architecture (Fig. 61) derives from that one described in the previous section and it has been applied to the variable gain controller scheme (4.2). The dead-zone amplitude is considered as a further input of the system and is the parameter to optimize while the positioning error is the measured output that is used for computing the performance of the system. The structure of this scheme has already been explained in the previous sections, however it is worth to focus on the computation of the performance J:

$$J = \frac{1}{T} \int_{t-T}^{t} e(\tau)^2 s(\tau) d\tau \tag{37}$$

The performance can be computed in different ways (signal amplitude, root mean square, infinity norm...) according to the application. In this case it has been chosen the integral of the squared error multiplied by a piecewise function $s(t)$. The utility of $s(t)$ is to select and weight some relevant portions of the error signal in order to increase their contribution to the performance. For example, considering the wafer scanner case, a silicon wafer covered with a photosensitive layer is moved under a laser beam that exposes a specific area. This operation is repeated for each section of the

wafer that needs to be exposed. Then a normal operative condition consists of the positioning phase, during which the wafer is moved under the beam, and the scanning phase where the stage moves at constant speed and the wafer area is exposed. Then the stage stops and moves to a new position where the process is repeated. It is clear that the most relevant part, from the controller performance point of view, is the scanning phase. The mask function $s(t)$ then should weight the constant velocity part during the scanning and the acceleration part before it (Fig. 62), in order to take into account also the positioning errors at the beginning of the operation. The remaining part of the movement can be neglected in the performance computation.

4.4 EXPERIMENTAL ACTIVITY

The variable gain controller and the extremum seeking techniques have been tested on the experimental setup shown in figure 63). The motion system is a magnetically levitated linear axis controlled in all of its six DOF. It is actuated by 6 linear motors with 2 DOF each, while the position measurements are performed by a 6 DOF interferometer with a resolution of $0.625\ nm$. Such a motion platform could be used for example in applications such as pick-and-place machines or wafer scanners. The main DOF is along the x axis, allowing a stroke of $80\ mm$, that will be used for in the experimental activity. The entire system is placed on a vibration isolation table that make it very insensitive to external disturbances. The controller design required to stabilize a MIMO system with 6 outputs and 6 inputs that become 7 when the variable gain controller has been implemented. The controller design has been performed on the following steps:

PLANT IDENTIFICATION The plant frequency response ha been computed by the measured FRFs of each axis.

CONTROLLER DESIGN The linear controller has been designed by using the experimental FRF of the plant and the loop shaping technique [112, 113] aided by a *ShapeIt*-like tool [114].

EXTREMUM SEEKING OPTIMIZATION The dead-zone length has been optimized with the ES control strategy.

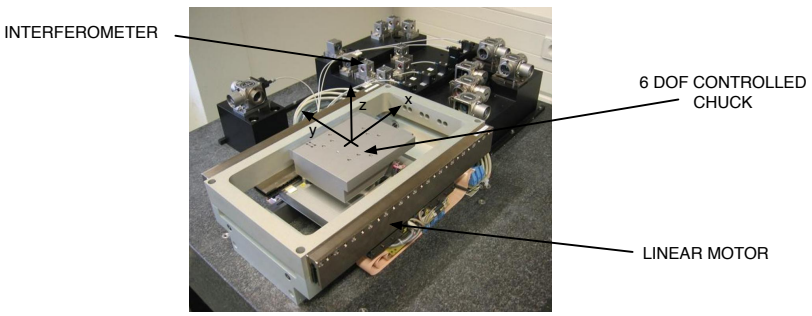


Figure 63: *Nforcer experimental setup*

4.4.1 *Frequency Response Analysis*

The first phase needed for the design of the *Nforcer* controller was the plant identification using the frequency response of the axes. Since an unknown controller had already been implemented a three point indirect method has been used (cf. §B.1). A white noise force disturbance has been injected in the plant and both plant input and output has been measured for each axes. The experiments have been repeated for different carriage positions along the entire stroke of the x axis with steps of 10 mm. Figures from 64 to 69 show the frequency response of the identified plant. The Bode plots show a single mass behavior with a slope of -40 $^{dB}/_{decade}$ in the low frequency part and the first mass decoupling over 100 Hz that is typical for linear motion systems. Moreover several resonance peaks appear at higher frequencies.

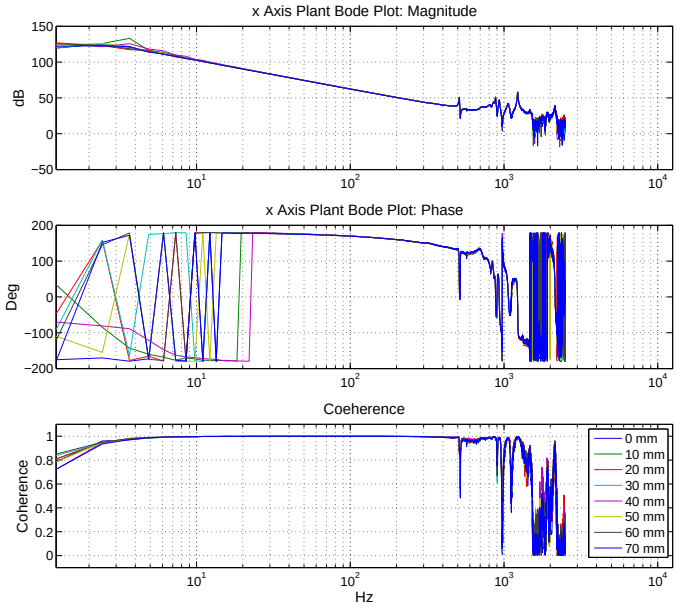


Figure 64: *Nforcer frequency response of x axis*

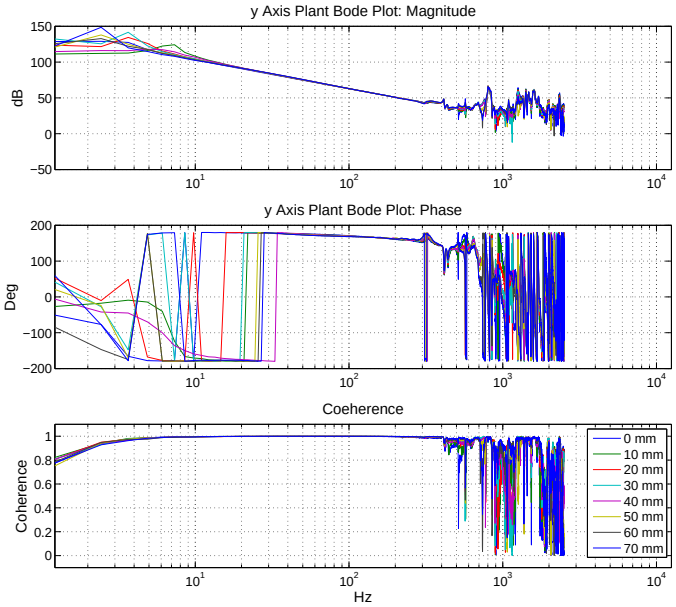


Figure 65: *Nforcer frequency response of y axis*

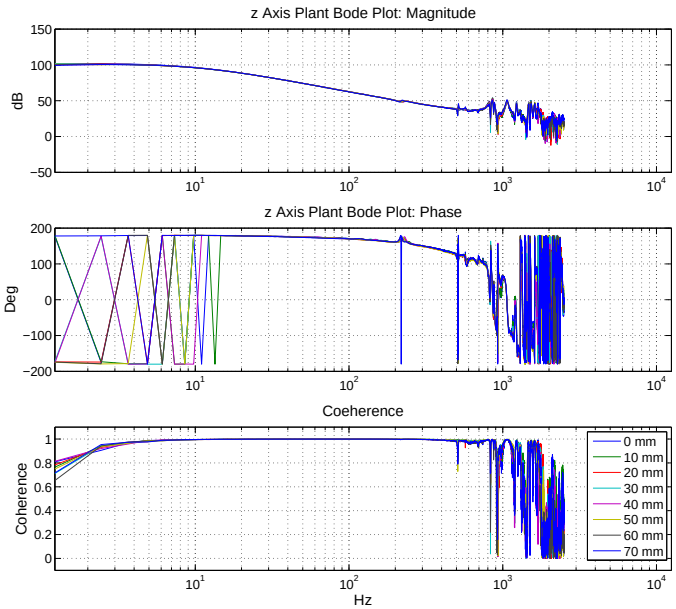


Figure 66: *Nforcer frequency response of z axis*

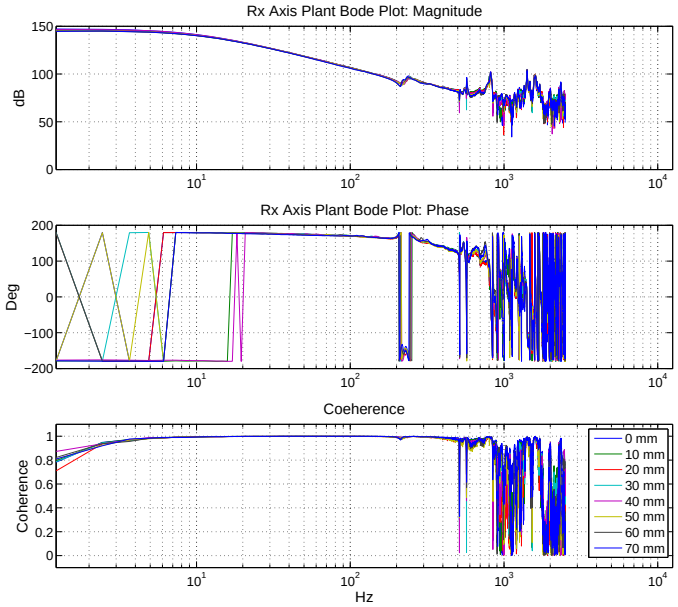


Figure 67: *Nforcer frequency response of R_x axis*

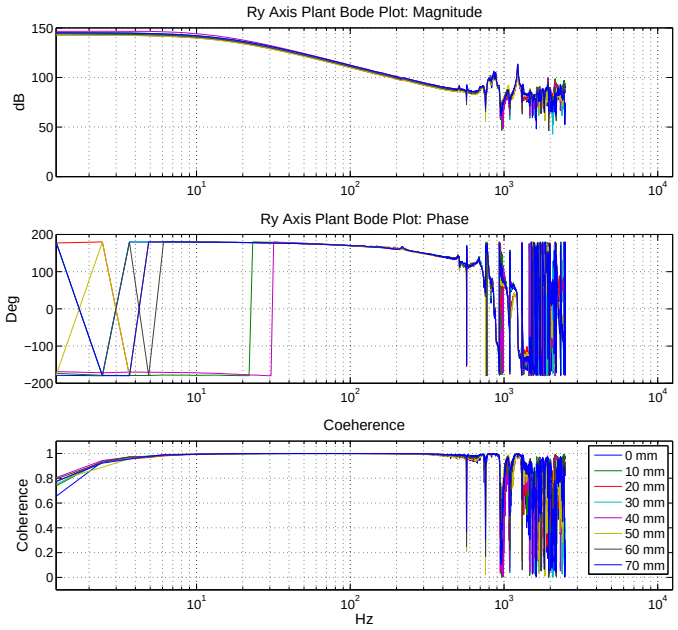


Figure 68: *Nforcer frequency response of R_y axis*

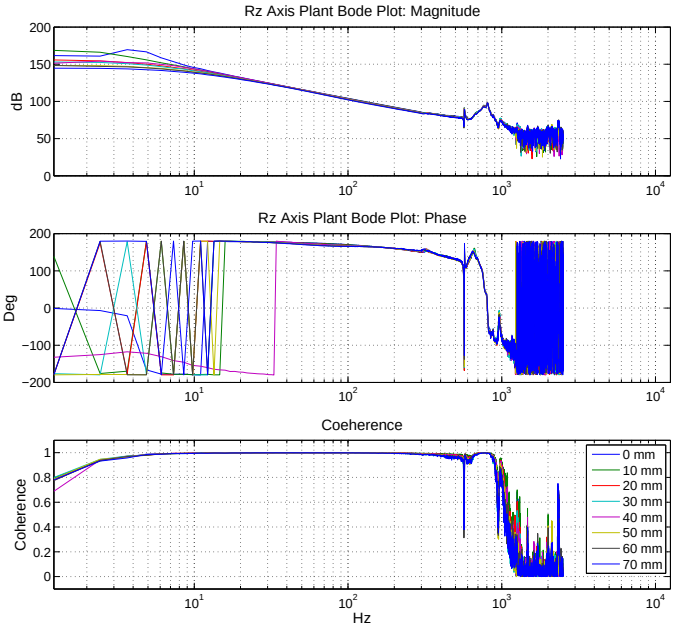


Figure 69: *Nforcer frequency response of R_z axis*

4.4.2 *Controller Design*

Linear Controller

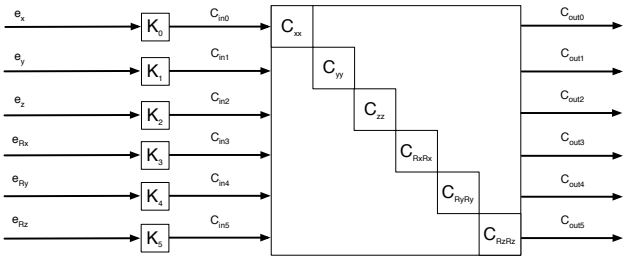


Figure 70: MIMO *controller structure*

The controller of the MIMO system has been designed following the sequential loop closing approach [115]. The method consists of develop the MIMO controller by designing a single SISO controller at time for each input-output pair. This is possible since there are no interactions among the control loops, that is, the controller transfer function is diagonal as can be seen in figure 70 where the controller structure implemented on the experimental motion system is shown. Then the multivariable nature of the plant can be neglected and each control loop can be closed sequentially without affecting the stability of the whole system. However, since the plant inputs and outputs are cross-coupled (Fig. 71), it is needed to take into account the effect of the closed loops in the controller design. The plant transfer function has to consider the contribution of the single SISO controllers and of the cross-

coupling effect between plant inputs and outputs, providing for each DOF an equivalent plant transfer function.

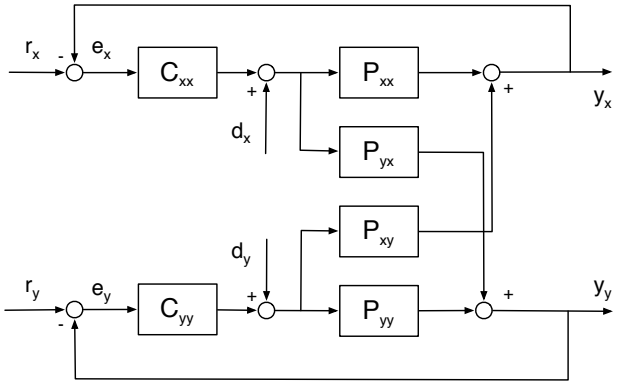


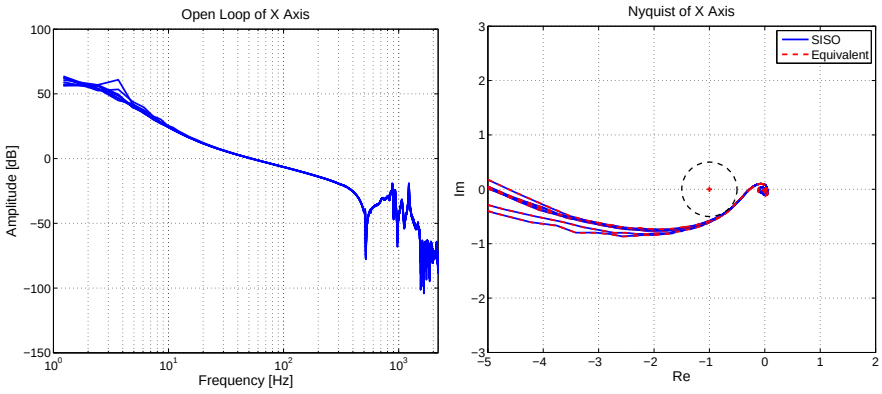
Figure 71: MIMO *system with interconnection*

Referring to the 2×2 MIMO system of figure 71, the equivalent plant is given by:

$$P_{xx}^{eq} = P_{xx} - P_{xy}C_{xx}(I + P_{yy}C_{yy})^{-1}P_{yx} \tag{38}$$

Once that each loop has been closed, the 6 SISO controllers are tuned using the equivalent plant, as it was the real plant, in order to reach the stability and the desired robustness of the MIMO system. It is worth to notice that the design of the controller did not require any model of the plant but it based only on the measured frequency response of the system. This allows to overcome modeling inaccuracies and deal with the actual system behavior. In the following sections will be provided the controller features for each axis of the experimental motion system showing the differences between the real and the equivalent plant.

x Axis



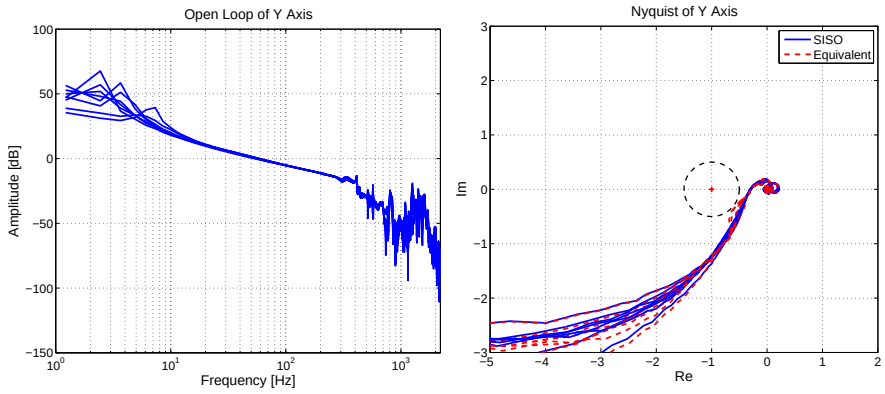
(a) *x* axis open loop transfer function (b) *x* axis Nyquist plot

Figure 72: *x axis* SISO *controller*

CONTROLLER BLOCKS	
GAIN	5.6×10^{-5}
LEAD FILTER	$\frac{\frac{1}{2\pi 15}s+1}{\frac{1}{2\pi 250}s+1}$
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 500)^2}s^2+\frac{2\cdot 0.4}{2\pi 500}s+1}$
INTEGRATOR	$\frac{s+2\pi 15}{s}$
NOTCH FILTER	$\frac{\frac{1}{(2\pi 514.6)^2}s^2+\frac{2\cdot 0.005}{2\pi 514.6}s+1}{\frac{1}{(2\pi 600)^2}s^2+\frac{2\cdot 0.5}{2\pi 600}s+1}$

Table 5: *x controller elements*

y Axis



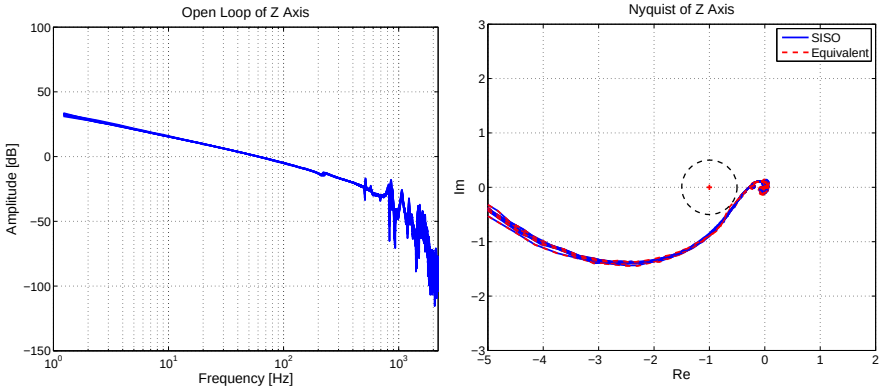
(a) *y* axis open loop transfer function (b) *y* axis Nyquist plot

Figure 73: *y axis* SISO *controller*

CONTROLLER BLOCKS	
GAIN	1.92×10^{-5}
LEAD FILTER	$\frac{\frac{1}{2\pi 5}s+1}{\frac{1}{2\pi 480}s+1}$
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 350)^2}s^2+\frac{2\cdot 0.5}{2\pi 350}s+1}$
INTEGRATOR	$\frac{s+2\pi 10}{s}$
NOTCH FILTER	$\frac{\frac{1}{(2\pi 804)^2}s^2+\frac{2\cdot 0.05}{2\pi 804}s+1}{\frac{1}{(2\pi 804)^2}s^2+\frac{2\cdot 0.5}{2\pi 804}s+1}$

Table 6: *y controller elements*

z Axis



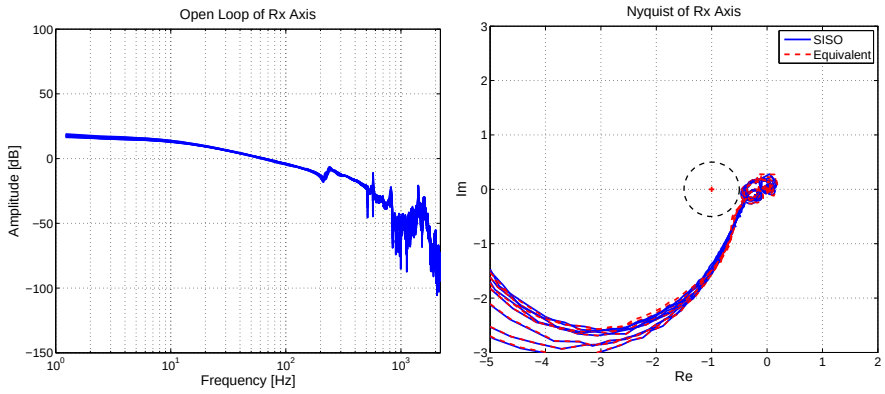
(a) z axis open loop transfer function (b) z axis Nyquist plot

Figure 74: *z axis* SISO *controller*

CONTROLLER BLOCKS	
GAIN	4.70×10^{-5}
LEAD FILTER	$\frac{\frac{1}{2\pi 10}s + 1}{\frac{1}{2\pi 180}s + 1}$
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 500)^2}s^2 + \frac{2 \cdot 0.5}{2\pi 500}s + 1}$
INTEGRATOR	$\frac{s + 2\pi 10}{s}$

Table 7: *z controller elements*

R_x *Axis*



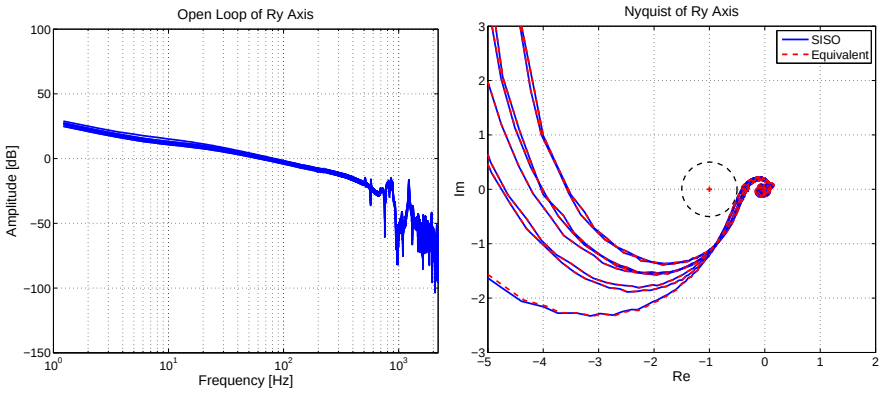
(a) R_x axis open loop transfer function (b) R_x axis Nyquist plot

Figure 75: *R_x axis* SISO *controller*

CONTROLLER BLOCKS	
GAIN	3.03×10^{-7}
LEAD FILTER	$\frac{\frac{1}{2\pi 10}s+1}{\frac{1}{2\pi 300}s+1}$
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 500)^2}s^2+\frac{2\cdot 0.5}{2\pi 500}s+1}$
INTEGRATOR	$\frac{s+2\pi}{s}$
NOTCH FILTER	$\frac{\frac{1}{(2\pi 828.4)^2}s^2+\frac{2\cdot 0.1}{2\pi 828.4}s+1}{\frac{1}{(2\pi 828.4)^2}s^2+\frac{2\cdot 0.5}{2\pi 828.4}s+1}$

Table 8: *R_x controller elements*

R_y Axis



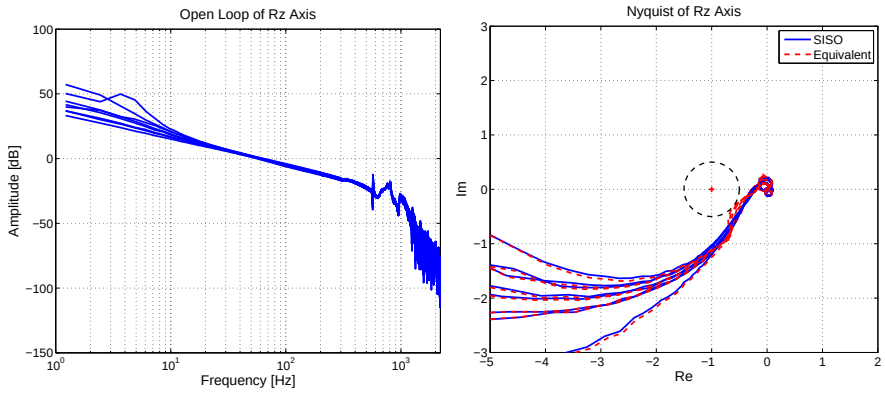
(a) R_y axis open loop transfer function (b) R_y axis Nyquist plot

Figure 76: *R_y axis* SISO *controller*

CONTROLLER BLOCKS	
GAIN	1.58×10^{-7}
LEAD FILTER	$\frac{\frac{1}{2\pi 8}s+1}{\frac{1}{2\pi 400}s+1}$
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 453)^2}s^2+\frac{2\cdot 0.5}{2\pi 453}s+1}$
INTEGRATOR	$\frac{s+2\pi 10}{s}$
NOTCH FILTER	$\frac{\frac{1}{(2\pi 1040)^2}s^2+\frac{2\cdot 0.1}{2\pi 1040}s+1}{\frac{1}{(2\pi 1040)^2}s^2+\frac{2\cdot 0.5}{2\pi 1040}s+1}$

Table 9: *R_y controller elements*

R_z *Axis*



(a) R_z axis open loop transfer function (b) R_z axis Nyquist plot

Figure 77: *R_z axis* SISO *controller*

CONTROLLER BLOCKS	
GAIN	3.20×10^{-7}
LEAD FILTER	$\frac{\frac{1}{2\pi 8}s+1}{\frac{1}{2\pi 300}s+1}$
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 500)^2}s^2+\frac{2\cdot 0.5}{2\pi 500}s+1}$
INTEGRATOR	$\frac{s+2\pi 10}{s}$
NOTCH FILTER	$\frac{\frac{1}{(2\pi 800)^2}s^2+\frac{2\cdot 0.1}{2\pi 800}s+1}{\frac{1}{(2\pi 1100)^2}s^2+\frac{2\cdot 0.1}{2\pi 1100}s+1}$

Table 10: *R_z controller elements*

Variable Gain Controller

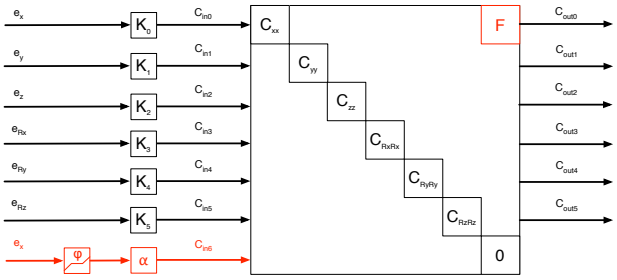
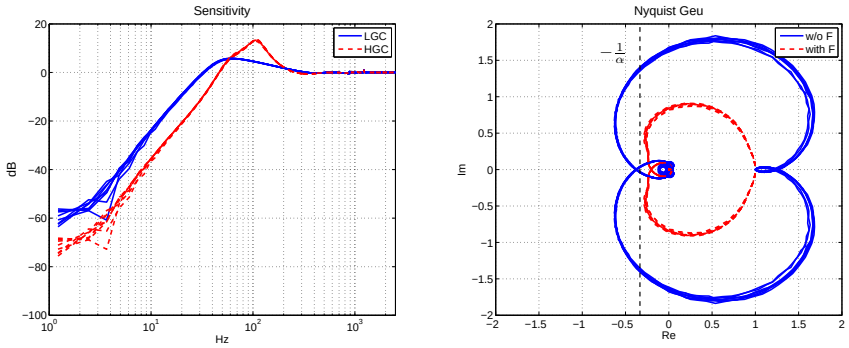


Figure 78: *VGC implementation in the* MIMO *controller structure*

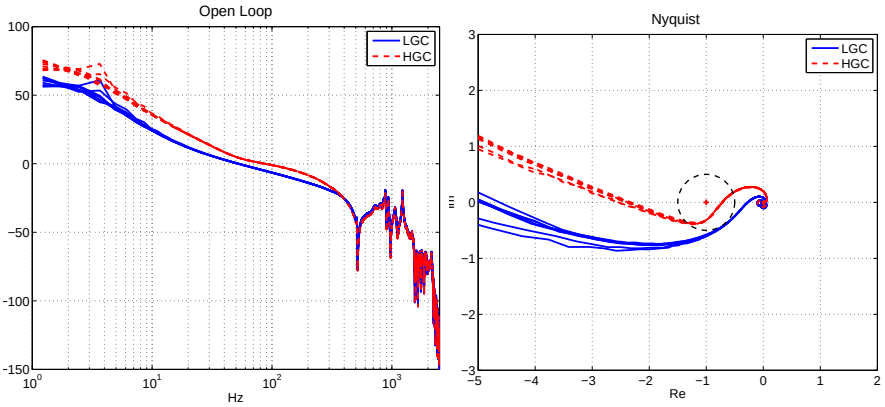
The variable gain controller implementation is obtained starting from the linear controller architecture, by adding a new input (C_{in6} in figure 78) that takes the tracking error in the x direction and processes it through the non-linear function ϕ. The result is then filtered by the loop shaping filter $F(s)$ and added to the x controller output (Fig. 78). The variable gain controller design requires that the extra gain and the filter $F(s)$ has to be selected. Figure 79b shows the G_{eu} transfer function both for the case without shaping filter $F(s) = 1$ and for the case with the designed filter. If no $F(s)$ is used the maximum extra gain allowed according to the circle criterion (cf. §4.2) is $\alpha = \frac{1}{0.63} = 1.59$. The designed shaping filter involved the use of a notch filter to increase the allowable gain and a low pass filter to reduce the high frequency noise (see table 11 for the filter features). In this way the available additional gain is increased to $\alpha = \frac{1}{0.26} = 3.8$ and for the considered application the extra gain $\alpha = 3$ was selected in order to have more robustness (red dashed line in figure 79b). As for the linear case, the stability has been assessed using the measured frequency response data. Compared to the linear case the VGC bandwidth is increased form $50\ Hz$ to $80\ Hz$ as it can be seen in figure 79a the sensitivity of the two limit configurations is shown. The response of the low gain controller (blue solid line) is obtained by choosing $\delta = \infty$ while the response of the high gain controller (red dashed line) is obtained with $\delta = 0$. The selection of intermediate values of the dead-zone length provides a tradeoff between this two characteristic and it will be optimized by the ES action.



(a) Variable Gain Controller sensitivity transfer function

(b) Circle criterion check

Figure 79: *Controller performance and stability*



(a) Variable Gain Controller open loop transfer function

(b) Variable Gain Controller Nyquist plot

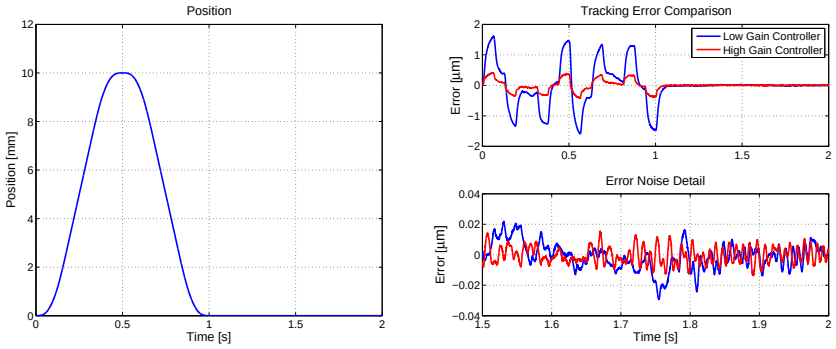
Figure 80

VGC BLOCKS	
GAIN α	3
LOWPASS FILTER	$\frac{1}{\frac{1}{(2\pi 300)^2}s^2 + \frac{2\cdot 0.7}{2\pi 300}s + 1}$
NOTCH FILTER	$\frac{\frac{1}{(2\pi 60)^2}s^2 + \frac{2\cdot 0.5}{2\pi 60}s + 1}{\frac{1}{(2\pi 40)^2}s^2 + \frac{2\cdot 1}{2\pi 40}s + 1}$

Table 11: *Shaping filter elements*

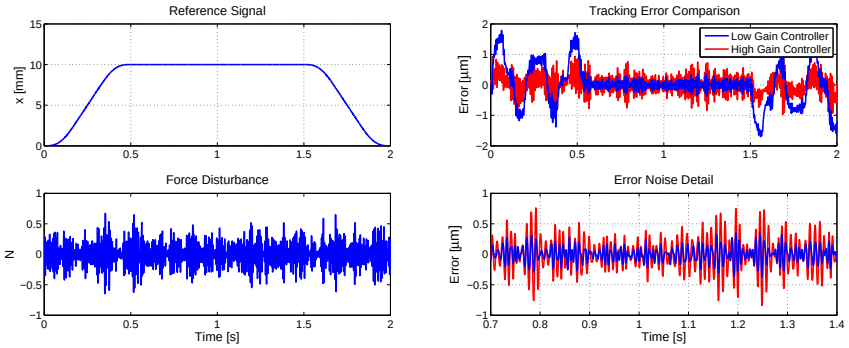
Tradeoff Assessment

The VGC approach is effective only if the tradeoff between low and high gain controller shows up. For this reason a set of preliminary tests has been conducted in order to evaluate the effect of gain variations on the measured error. Figure 81 shows the result of the first test where a third-order motion profile has been used as reference signal. The resulting error proves that the high gain controller is the best solution since the noise level is so low that no significative amplification is introduced. This behavior has not been unexpected since the experimental setup design and the use of the isolation table are intended to avoid as much as possible the effect of external disturbances. Then a second test has been necessary in order to emulate a real application and make evident the tradeoff between the controller setup. Machines operating in an industrial environment are affected by high-frequency force disturbances (e. g. the cross-talk from other machine components, the measurement noise, the contribution of adjacent machines...). To obtain such effect an high frequency colored disturbance, obtained by filtering white noise in the $100 - 200$ Hz band, has been injected in the system. A third-order displacement profile similar to the previous one has been used for the reference input (Fig. 82a). Figure 82b shows that the injected noise produces the desired effect. The better performance during the motion phase of the high gain controller is counterbalanced by the noise amplification in the standstill position. In this situation the controller yield can be effectively improved by the VGC with the proper tuning of the dead-zone length. In this regard will be presented the experimental results regarding the use of the extremum seeking strategy to optimize the variable gain controller as proposed in section 4.3.5.



(a) Reference signal (b) Error signal

Figure 81: *Preliminary tradeoff test*



(a) Reference signal (b) Error Signal

Figure 82: *Tradeoff test with injected noise*

4.4.3 *Extremum Seeking Tests*

The purpose of experimental activity on the extremum seeking control implemented in the experimental motion system is twofold: on one hand the optimization of the dead-zone length of the VGC and, on the other hand, to provide a better knowledge on the effects of the ES parameters choice (a, ω, c). As stated in section 4.3.1 the dither frequency ω, the learning rate c and indirectly also the dither amplitude a contribute to the time scale separation among the system dynamics and the ES dynamics which is the key aspect of the optimization procedure. However there are no specific rules about the choice of these parameters that usually is done with a trial-and-error approach. The tests, executed with different configuration of the ES algorithm, provided a better insight of the role of each parameter leading to more specific guidelines on their selection. The experimental activity that will be presented has been divided in three parts. First the static performance map as been built from the measured data in order to verify the extremum seeking convergence. Then some tests have been performed to prove the optimization of the dead-zone amplitude and finally the last experiments have been conducted to understand the effect of each parameter on the ES yield.

Input Signals

The input signals used in the experiments are the reference signal, the high frequency force disturbance and a mask function $s(t)$. The choice of these signal has been related to the case of a pick and place operation: the system is moved in the commanded position in $0.5\ s$ where is kept standstill for placing the component. After $1\ s$ the carriage is moved back to the starting point in $0.5\ s$ to pick a new component and then the operation is repeated resulting in a $T = 2\ s$ periodic path. The reference signal is the third-order motion profile used in the tradeoff assessment experiments with a $10\ mm$ stroke. The system is moved with a velocity of $0.032\ m/s$, an acceleration

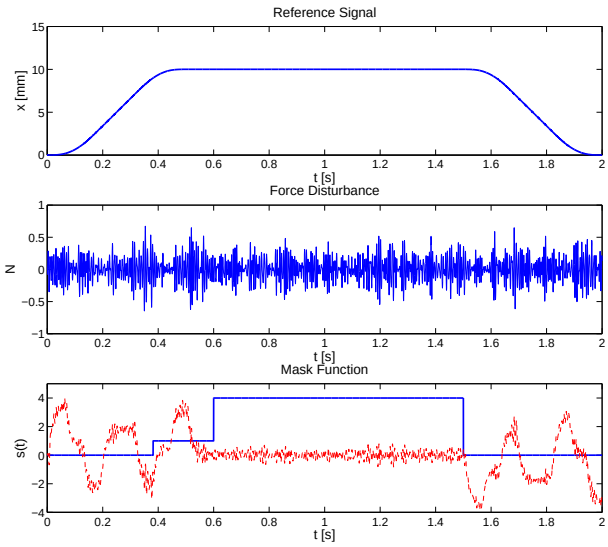


Figure 83: *Input signals*

of $0.256\ m/s^2$ and the resulting jerk is $4.096\ m/s^3$ (Fig. 83). The force disturbance is white noise filtered in the $100 - 200\ Hz$ band. The frequency content of the disturbance is well separated from the $0.5\ Hz$ frequency of the reference signal and then it can be approximatively considered as a periodic signal with period $T = 2\ s$. As a consequence the $T-$periodic bounded reference $(r(t))$ and disturbance $(d(t))$ inputs satisfy the first stability assumption of the ES strategy (cf. §4.3.4). The mask function $(s(t))$ is needed to tailor the controller performance to the specific application. In the case of the pick and place task the machine should guarantee high positioning accuracy while the component is being placed. For this reason the mask applied to the signal has mainly to highlight the contribution of the error during the approaching phase and mostly when the system is kept in the standstill position, during the placing phase. All the other contributions can be neglected in the performance computation. The selected mask function is shown in the last plot of figure 83 together with the normalized error (red dashed line). It is worth to notice that the use and the choice of the mask function is completely arbitrary being related to the specific case. A different mask as that one of section 4.3.5 would lead to a different performance map and then to a different optimization result. However this does not affect the convergence of the algorithm until an extremum point can be found.

Static Performance Map

The static performance map (Fig. 84) has been built by measuring the steady-state performance for 100 different values of δ in the range between $0\ \mu m$ (high gain controller) and $2.5\ \mu m$ (low gain controller). Since the operating conditions of the experimental motion system slightly changed during

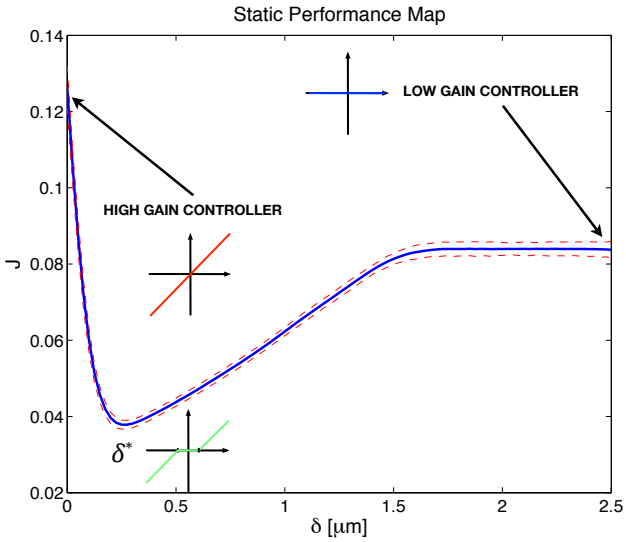


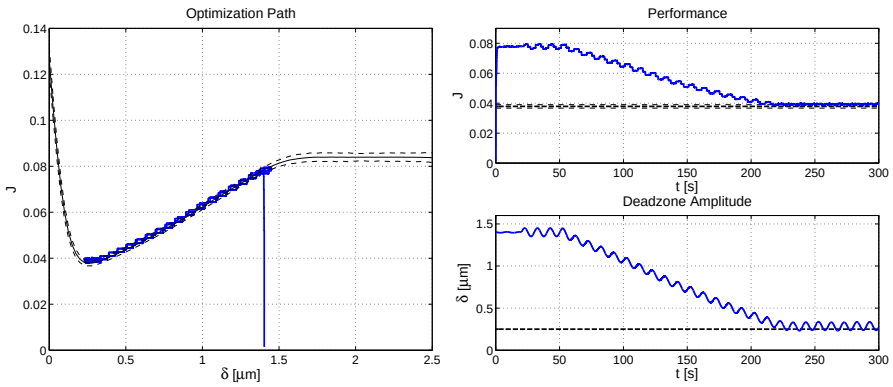
Figure 84: *Static performance map from experimental test*

the tests, the static map has been measured ten times and then the averaged curve (blue solid line) and the confidence interval $\pm 3\sigma$ (red dashed line) have been used for the extremum seeking validation. The changes in the system response does not constitute an issue for the optimization since they are slower than the convergence dynamics and indeed they provide a good proof of the extremum seeking capability of continuously adapt to the optimal solution.

Dead-zone Optimization

The extremum seeking convergence has been tested setting the the dither amplitude $a = 0.05\ \mu m$, the dither period $T_\omega = 10\ s$ and the adaptation rate $c = 0.3$. The optimal dead-zone δ^* has been found starting from the initial dead-zone amplitude $\delta = 1.4\ \mu m$. At the beginning the adaptation has been disabled ($c = 0$) for $30\ s$ in order to let the stabilized plant to converge to its steady-state performance. The convergence of the plant dynamics is constrained by the input period since at least T seconds of data history are needed to build up the information on the steady-state performance. Moreover also the dither period contribution has to be taken into account since T_ω seconds of history are needed to perform the gradient estimate. Then the minimum time required before starting the adaptation is equal to $T + T_\omega$ seconds. Figure 85 shows that extremum seeking converges in a neighborhood of the optimal dead-zone $\delta^* = 0.25\ \mu m$ in around $200\ s$. The time scale separation can be seen in the convergence detail of figure 86. The fast dynamic is included in the $2\ s$ T-period, the medium dynamics is highlighted by the T_ω-periodic oscillations of the performance and the slow dynamics is the whole convergence time. In the right plot of figure 86 it is also possible to notice the effect of the mask function with the varying part

related to the weights applied to the tracking error and the constant values related to the neglected part of the signal.



(a) Dead-zone Optimization (b) Performance and δ convergence

Figure 85: *Extremum seeking optimization*

PARAMETERS	
DITHER AMPLITUDE $[a]$	0.05 μm
DITHER PERIOD $[T_\omega]$	10 s
LEARNING RATE $[c]$	0.3

Table 12: *Extremum seeking parameters of the optimization experiment*

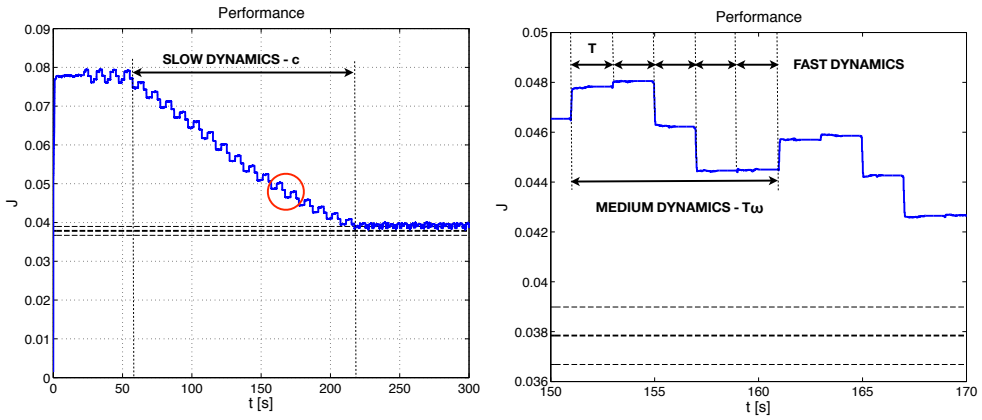


Figure 86: *Time scale separation*

The tracking error resulting from the optimized dead-zone is shown in figure 87 where it can be seen that the optimized VGC clearly outperforms the linear controller. The extra gain applied only during the motion phase reduces the tracking error of the low gain controller while the high frequency noise is not amplified thanks to the proper selection of the dead-zone amplitude.

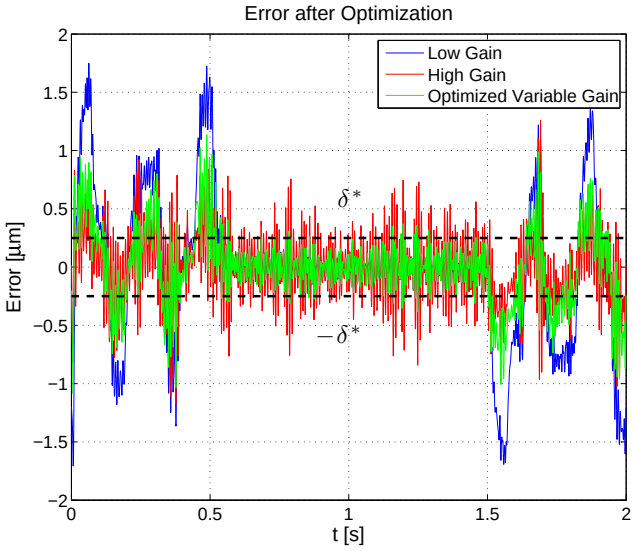


Figure 87: *Optimization result on the tracking error*

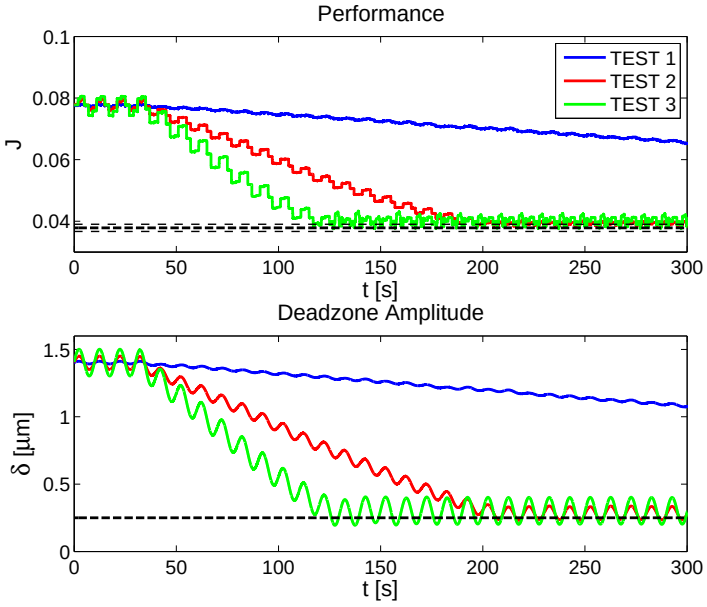


Figure 88: *Dither amplitude and adaptation gain effect*

PARAMETERS

	a	c	T_ω
TEST 1	0.01 μm	0.1	10 s
TEST 2	0.05 μm	0.3	10 s
TEST 3	0.1 μm	0.5	10 s

Extremum Seeking Parameters Choice

The extremum seeking parameters affect both the convergence time and the accuracy of the algorithm. However they are also responsible for keeping the time scale separation which ensures the convergence of the optimized variable. This implies that the parameters are related to each other and can not be freely selected. The dither amplitude a is directly related with the neighborhood size of optimal δ^* value to which the ES converges and then it defines the convergence accuracy of the algorithm. However it can not be chosen too small without reducing the adaptation parameter c as well. The learning rate in fact determines the slowest dynamics of the algorithm and should be kept small enough compared to the faster dither dynamics. If c is too high the adaptation will move quickly out of the δ neighborhood explored by the dither signal causing a wrong gradient estimate and, in the worst case, the break down of the time scale separation. In figure 88 are shown three tests conducted with the same dither period and different

values of the $a - c$ pair. Compared to the other experiments the first test, conducted with a dither amplitude of $0.01\ \mu m$, will converge in a smaller neighborhood of the solution resulting to be more accurate. However it requires also a smaller adaptation gain and then the convergence time is increased as well (units 17 minutes). Similarly also the dither period must be properly selected according to the plant dynamics. The period T_ω rules the gradient estimate dynamics that must be slower than the system response, so that the measured performance is kept close to the steady-state static map. In this specific case, that involves periodic inputs, the system dynamics is determined not only by the plant transients but also by the period T of the input signal. The dither period should be higher than the input period to guarantee the time scale separation or in other words, it should be high enough to perceive δ as constant in one signal period. The effect of the dither period choice is shown in figure 89. Four tests has been conducted choosing T_ω as a multiple of the input period ($2T$, $3T$, $4T$, $5T$) and keeping the same dither amplitude and learning rate. In the first experiment T_ω was too close to the input period resulting in a completely wrong gradient estimate. The ES then moved in the opposite direction diverging from the optimal solution. Choosing $T_\omega = 3T$ made the algorithm to converge, however the time scale separation is not yet fully observed and the gradient is underestimated making the convergence slower than the remaining two tests. These two last cases show quite the same response due to a proper choice of the dither period. On the other hand, it is worth to notice that a too high T_ω will move the dither dynamics too close to the convergence dynamics, requiring as a consequence a smaller adaptation rate and then a slower optimization.

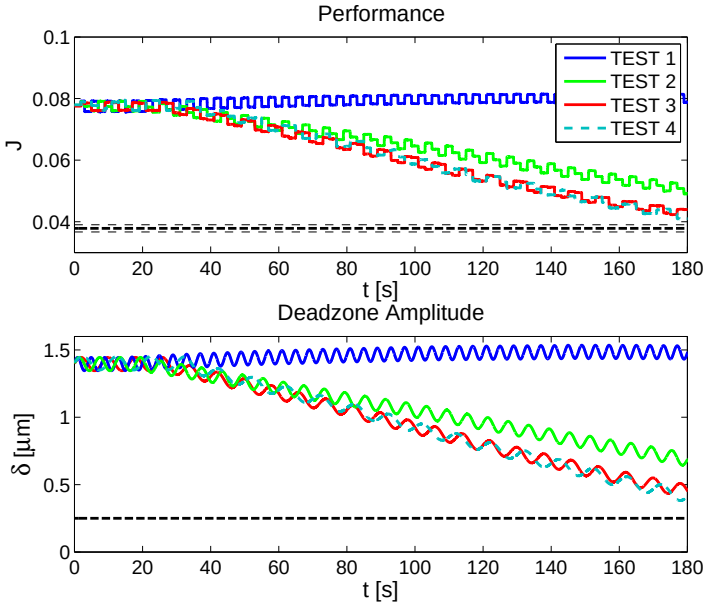


Figure 89: *Dither period effect*

PARAMETERS

	a	T_ω	c
TEST 1	0.05 μm	$2T$	0.3
TEST 2	0.05 μm	$3T$	0.3
TEST 3	0.05 μm	$4T$	0.3
TEST 4	0.05 μm	$5T$	0.3

4.4.4 *Discussion*

The proposed ES approach proved to be useful for the inline performance optimization of the VGC. The application of this method to the control of linear motion system allowed to achieve low tracking error and good disturbance rejection. The optimized VGC outperformed the linear motion controller avoiding the occurrence of the waterbed effect and providing an optimal tradeoff among the conflicting controller features. On the other hand the experimental results highlighted that the use of the ES requires an accurate selection of the parameters in order to guarantee the convergence to the extremum point. For this reason the knowledge of the system dynamics and a minimal assessment of the performance function are suggested so that the ES loop can be properly designed.

5

CONCLUSION

According to the current trends in machining manufacturing, that requires high production rate with high quality of the machined parts, the enhancement of the automation level in modern CNC machine tools is a compelling need. The key requirements are intended to prevent working defects, rejects and unexpected downtimes. In this context a monitoring system has been developed with the aim to supervise the working operation of CNC machine tools. The system has been designed as a distributed architecture with one or more nodes that perform the monitoring tasks and one node that manage the communication between the monitoring system and the NC of the machine. Each monitoring node is based on a microcontroller that acquires data from sensors, executes the programmed tasks communicating the results to the machine and, if required, commands a corrective action as the regulation of the cutting parameters. The proposed architecture is scalable, allowing the connection of new nodes to the existent system, and thanks to the reduced size, it is easily integrable in the machine structure. This guarantees an high flexibility for the choice and location of monitoring nodes allowing the upgrade of the system if new sensors and new monitoring task are required. The implementation of the monitoring algorithms has been performed with the twofold aim of robustness and ease of use. In the first case the system should properly operate in several working conditions including different cutting process, different tools and even different machine architecture. In the second case the developed monitoring algorithms do not require specific knowledge to the machine user providing easy to understand outputs and a limited set of configuration parameters. The developed monitoring tasks includes:

- Vibration Monitoring and vibration mitigation executed during the milling process with the purpose of containing the vibration level that can affect the surface finishing and the tool life.

- Condition Monitoring needed to assess the health of the spindle bearings. It detects a possible fault and provides the diagnosis of the faulty element.

- Collision Detection used to halt the machine in case of a dangerous situation as a collision. It provides a fast response that is not achievable by the machine operator preventing severe damages to the machine. It is also useful for minimally supervised machining operation. In addition a similar task has been developed for the accessory heads of the machine in order to allow the onboard node to record events when the device is not operative.

Finally these algorithms have been successfully tested with dedicated experiments and subsequently they have been embedded in a state machine

architecture that constitutes the main software framework. Each monitoring activity can be executed by the machine operator except for the Collision Detection task that is always active for safety reasons. The developed supervisory system is an innovative device in the machine tool field capable to adapt to different working conditions and perform several monitoring tasks, making it suitable for equipping machine tools operating in a industrial environment. The present work is concluded with the application of a novel control strategy to linear motion systems with the aim of optimizing the controller performance. A Variable Gain Controller scheme has been applied to plant in order to improve the response of the system in terms of tracking error and disturbance rejection. Then an Extremum Seeking-based algorithm has been used to optimize the variable gain of the controller. The proposed approach has been experimentally validated with success providing also a better insight on the tuning of Extremum Seeking loop. This study provided a promising strategy for the inline calibration of the machine axes in order to adapt the motion controller response to the actual operative conditions of the machine by compensating the contribution of slowly varying effects (e. g. temperature variation).

Future Developments

The developed supervisory system offers a flexible architecture for the monitoring of machining operations. The system can be easily adapted to different machine tools and machining processes and further monitoring task can be added. Future activities will aim to improve the whole system by enhancing the performance of monitor nodes with powerful microcontrollers and implementing more sophisticated algorithms. In addition the monitoring system will be integrated with a controller for the offline optimization of the working process with the purpose of achieving an intelligent manufacturing system capable of optimizing and controlling the process execution.

APPENDIX

A

CONTROL THEORY

A.1 THE WATERBED EFFECT

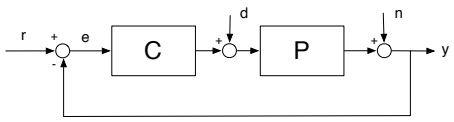


Figure 90: *Controller scheme*

The waterbed effect describes the main limitation of the performances in linear control design. A proper control design should satisfy requirements in terms of robustness, reference tracking and disturbance rejection. The sensitivity function S then plays an important role in the design since it is related both to the tracking error and to the disturbance attenuation. In fact considering the controller scheme of figure 90 the tracking error can be written as:

$$e = \frac{1}{1+PC} \cdot (r-n) - \frac{P}{1+PC} \cdot d = S \cdot (r-n) - P_s \cdot d \tag{39}$$

where S and P_s are respectively the sensitivity and the process sensitivity. According to equation 39 the desired sensitivity has to be small inside the bandwidth where we want high performance but it should be kept low also in other frequency ranges where disturbances and noise affect the controller response. This is an issue for most of the linear controllers since there exists an important constraint for the sensitivity function due to the Bode sensitivity integral.

Theorem 1 (Bode Sensitivity Integral) *Let L and S be respectively the open loop transfer function and the sensitivity function. If $L(s)$ has p_k poles on the right half-plane than the sensitivity function satisfies the following relation:*

$$\begin{aligned} \int_0^\infty log(S(i\omega))d\omega &= \int_0^\infty log\left(\frac{1}{1+L(i\omega)}\right)d\omega \\ &= \pi \sum_k Re(p_k) - \frac{\pi}{2} \lim_{s\to\infty} sL(s) \end{aligned} \tag{40}$$

The theorem shows that the integral of $|S(s)|$ remains constant allover the frequency band and this implies that the amount of error suppressed in a certain frequency range will be redistributed elsewhere. In other words, improving the error rejection of the controller (e.g. increasing the bandwidth)

will cause an amplification of the error in the high frequency region. If the system is stable (no poles in the right half-plane) and $L(s)$ has two more poles than zeros, how usually happens in motion systems, the equation 1 turns in:

$$\int_0^\infty log(S(i\omega))d\omega = 0 \tag{41}$$

This equation can be easily explained thinking to the Bode plot of the sensitivity function (Fig. 91). In fact, due to (41), the area beneath the curve within the controller bandwidth must be the same of the area beneath the curve beyond the bandwidth. In figure 91 it can be seen how this phenomenon affects the performance of the controller. An higher gain implies a wider band and a better attenuation of error, but beyond the crossover frequency the error will be much more amplified by the controller with the higher gain.

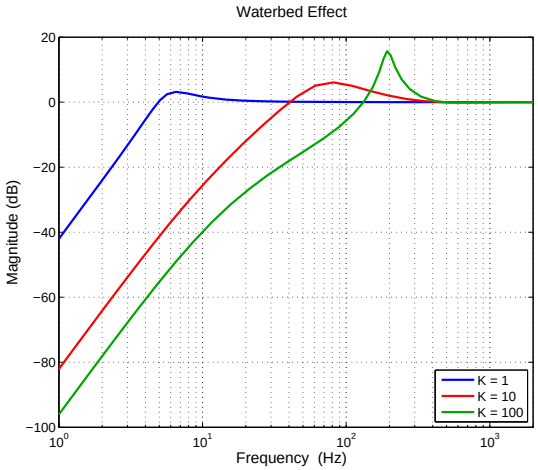


Figure 91: *Waterbed effect*

A.2 STABILITY OF NON LINEAR SYSTEMS

The stability of a linear control system is a condition uniquely determined and independent of the presence of inputs (bounded). The study of the stability leads to a steady state solution that has the following characteristics:

- Is unique and is valid both for the unperturbed and perturbed system.
- Is periodic if the system inputs are periodic.

These assumptions are no more valid in the study of non linear control systems. In general a non linear system has complex steady state solution depending on its inputs and on initial conditions. Studying the stability of a non linear system means to define two main properties:

INPUT TO STATE STABILITY determines if the system is stable with no inputs and if the system has a bounded or convergent response with bounded or convergent inputs.

CONVERGENCE OF SOLUTIONS determine if the steady state solutions of the system with different inputs converge to a common equilibrium point.

Input to State Stability

Considering a linear system give by the equation:

$$\dot{x} = Ax + Bw(t) \tag{42}$$

If the system is stable i.e. A is *Hurwitz* (poles are in the left half plane) and is perturbed by a bounded input ($w(t)$), then the general solution of the system is:

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t0}^{t} e^{A(t-\tau)}Bw(\tau)d\tau \tag{43}$$

and the output (state) x(t) is bounded according to:

$$\|x(t)\| \leq k\|x(t_0)\|e^{-\lambda(t-t_0)} + \frac{k}{\lambda}\|B\| \sup_{t_0 \leq s \leq t} \|w(s)\| \quad \text{with } k,\ \lambda > 0 \tag{44}$$

Equation 44 shows that the system response is given by the contribution of two elements; the first term represents the system response with no inputs (*zero-input stability*) while the second term describes the system evolution due to bounded or converging inputs (*asymptotic gain property*). For linear systems this two properties are equivalent and the existence of one of them implies also the existence of the other one. Similarly the stability of non linear system involves these two properties that however are not related to each other. The definition of the *input to state stability* property includes these two notion and allows to describe the stability of a non linear system both under unperturbed and perturbed conditions.

Theorem 2 *The non linear system:*

$$\dot{x} = f(x, w(t)), \qquad f(0,0) = 0 \tag{45}$$

is input to state stable *if there exist a* $\mathcal{K}$-function $\gamma(w)$ *(strictly increasing function) and a* $\mathcal{KL}$-function $\beta(x,t)$ *(strictly increasing in x, decreasing in t) such that:*

$$\|x(t)\| \leq \beta(\|x(t_0)\|, t - t0) + \gamma\left(\sup_{t_0 \leq s \leq t} \|w(s)\|\right) \tag{46}$$

The *input to state stability* implies that the equilibrium point $x = 0$ of the unperturbed system is globally uniformly asymptotic stable and that a bounded input leads to a bounded output. In general it is required to find a *ISS-Lyapunov* function in order to state whether a system is input to state stable and to compute the functions β and γ.

Theorem 3 () *The non linear system 45 is* ISS-*stable if and only if there exists a smooth* ISS-Lyapunov *function. A function $V(x,t)$ is an* ISS-Lyapunov *function, if there exist class $\mathcal{K}_\infty$-*functions *$\alpha_1(\|x\|)$, $\alpha_2(\|x\|)$ (a $\mathcal{K}$-*function *which goes to infinity with $x \to \infty$) and class $\mathcal{K}$-*functions *$\alpha_3(\|x\|)$, ρ such that:*

$$\begin{aligned} &\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|) \\ &\dot{V} = \frac{\partial V}{\partial x} f(x,w) \leq -\alpha_3(\|x\|), \ for \|x\| \geq \rho(\|w\|) \end{aligned} \tag{47}$$

Finding a *ISS-Lyapunov* function, generally is difficult, however for some classes of systems as *Lur'e* type systems there exist constructive conditions that simplify the definition of input to state stability.

Stability of Lur'e Type Systems

A non linear control system is treated as a *Lur'e* problem if it can be reduced to a system with a linear time invariant part in the feedforward path and a nonlinearity in the feedback loop (figure 92).

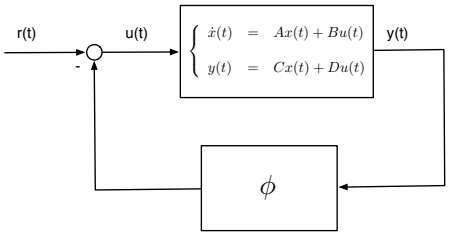


Figure 92: *Lur'e type system scheme*

Considering an unforced system ($r = 0$) described by:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \\ u &= -\phi(t,y) \end{aligned} \tag{48}$$

where $\phi(t,y)$ is *Lipschitz* in y and $G(s) = C(sI - A)^{-1}B + D$ is a rational proper transfer function of the system, then for all the non linearities $\phi(t,y)$ satisfying a sector condition, the origin $x = 0$ is a equilibrium point. The system 48 is absolutely stable if the origin is globally uniformly asymptotic stable for any non linearity in the given sector. For *Lur'e* type systems this conditions are verified by the *Circle Criterion*.

Theorem 4 (Circle Criterion) *The non linear system 45 is absolutely stable if one of this conditions holds:*

- *$\phi \in [0, \infty]$ and $G(s)$ is strictly positive and real (Fig. 93a).*
- *$\phi \in [K_1, \infty]$ and $G(s)(1 + K_1 G(s))^{-1}$ is strictly positive and real (Fig. 93b).*

- $\phi \in [K_1, K_2]$ *with* $K_2 > K_1$ *and* $(1 + K_2G(s))(1 + K_1G(s))^{-1}$ *is strictly positive and real (Fig. 93c).*
- $\phi \in [0, K_1]$ *and* $(1 + K_1G(s))$ *is strictly positive and real (Fig. 93d).*

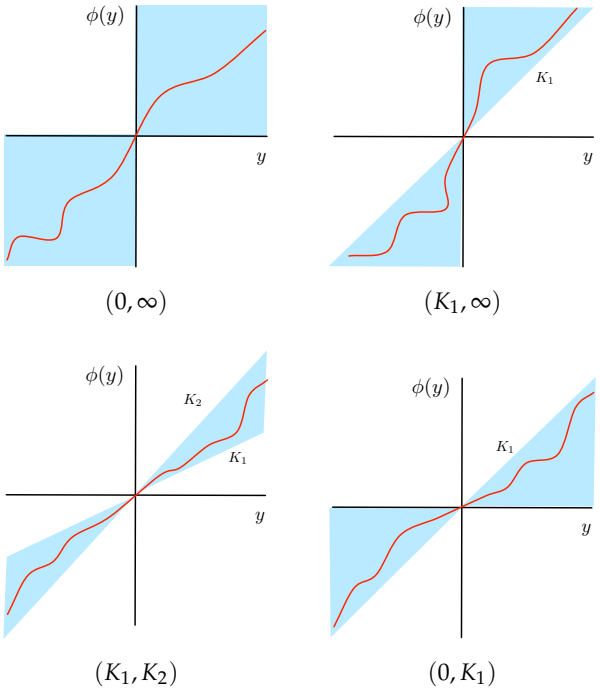


Figure 93: *Sector conditions for non linear functions*

Definition 1 *A proper rational transfer function* $H(s)$ *is strictly positive and real if and only if:*

1. $H(s)$ *has all poles in the left-half plane,* (Hurwitz).
2. $Re(H(j\omega)) > 0 \; \forall \omega \in \mathbb{R}$.
3. $H(\infty) > 0$

Convergence of Solutions

Unlike linear systems, the steady state response of a non linear system is not unique. In general different initial conditions lead to different equilibrium points and it is not possible to determine them a priori, making the analysis of the non linear system difficult. Then it would be advantageous to have a non linear system whose response deriving from any initial condition will converge to a unique steady state solution determined only by the input. The convergence property is defined by:

Definition 2 (Convergence) *A non linear system*

$$\dot{x} = f(x, w(t)) \tag{49}$$

is:

- Convergent *if:*
  * $\exists$ *a unique (defined and bounded) steady state solution* $\bar{x}_w : \sup_{t\in\mathbb{R}} \|\bar{x}_w(t)\| < +\infty$.
  * $\bar{x}_w(t)$ *is globally asymptotically stable.*
- Uniformly convergent *if* $\bar{x}_w(t)$ *is convergent and globally uniformly asymptotically stable.*
- Exponentially convergent *if* $\bar{x}_w(t)$ *is convergent and globally exponentially stable.*

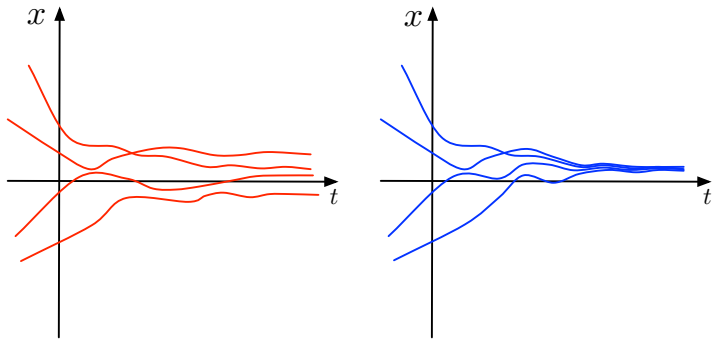


Figure 94: *Convergent and not convergent solutions*

The importance of the convergence property lies on the fact that we know what will be the steady state response of the system regardless of the initial conditions. A convergent non linear system will have a unique bounded globally asymptotically stable solution for any $t \in [-\infty, \infty]$ that depends on the input. In case of a constant input the response will be constant and with a periodic input the steady state behavior will be periodic with the same period. To check if the convergence property belongs to a given system it can be used a sufficient condition proposed by Demidovich in [116]:

Theorem 5 *Consider the nonlinear system (49) and let the function* $f(x, w(t))$ *be* C^1 *with respect to* x *and continuous with respect to* w. *If there exist two positive definite symmetric matrix* P, Q *such that*

$$P\frac{\partial f}{\partial x} + \frac{\partial f}{\partial x}^T P \leq -Q \; \forall x \in \mathbb{R}^n, \; w \in \mathbb{R}^m \tag{50}$$

then the system is exponentially convergent for all bounded $w(t)$

This theorem applies also to the conditions proposed by Yakubovich in [117] for global asymptotic stability of all solutions for Lur'e type systems. This method ensures the convergence property for systems of the form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_w w(t) \\ y(t) &= Cx(t) + Du(t) \\ u &= -\phi(t, y) \end{aligned} \tag{51}$$

if the conditions of the following theorem are met:

Theorem 6 *The Lur'e type system (51) is convergent if:*

- $0 \leq \dfrac{\phi(y_2) - \phi(y_1)}{y_2 - y_1} \leq \alpha, \ \forall y_2 \neq y_1.$
- $Re(G_{yu}(s)) < -\dfrac{1}{\alpha}.$
- $G_{yu}(s)$ *is* Hurwitz.

The first condition is called incremental sector condition and implies that the derivative of the nonlinearity ϕ must be always positive and at least equal to the constant value α. If the non linear system (49) can be described as a Lur'e type system and both the circle criterion and the incremental sector condition are valid, then there exists a unique globally asymptotically stable solution, bounded for $t \in [-\infty, \ \infty]$. Moreover if the system is subject to periodic inputs with period T, also the steady state response will be periodic with the same period.

B

FREQUENCY RESPONSE ESTIMATE

B.1 INDIRECT FREQUENCY RESPONSE MEASUREMENTS

Frequency response measurement are generally used in control design for the identification of the plant model. The simplest case is to identify the plant when it is in a open loop configuration.

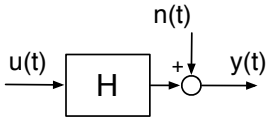


Figure 95: *Open loop scheme*

Then given a plant H approximately linear subject to an unknown disturbance $n(t)$ and to a known input $u(t)$, the measured output is obtained by:

$$y(t) = H \otimes u(t) + n(t) \tag{52}$$

that in frequency domain becomes:

$$Y(f) = H(f)U(f) + N(f) \tag{53}$$

Multiplying every therm of the equation with the complex conjugate of the Fourier transform of the input $U^*(f)$ we obtain:

$$Y(f)U^*(f) = H(f)U(f)U^*(f) + D(f)U^*(f) \tag{54}$$

that can be written as

$$S_{yu}(f) = H(f)S_{uu}(f) + S_{nu}(f) \tag{55}$$

where S_{ii} is the power spectral density of i and S_{ji} is the cross power spectral density between i and j. If the input $u(t)$ and the disturbance $n(t)$ are uncorrelated then:

$$S_{nu}(f) \approx 0 \tag{56}$$

and then the transfer function of the plant can be obtained by:

$$\frac{S_{yu}(f)}{S_{uu}(f)} \approx H(f) \tag{57}$$

Indirect Methods

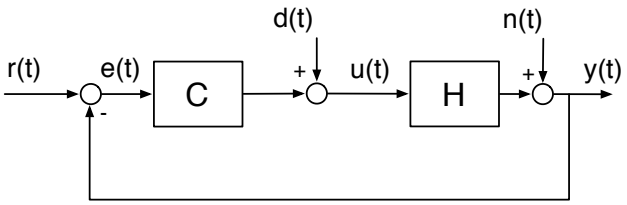


Figure 96: *Close loop scheme*

The direct method is not always applicable since there could be situation where the plant is already controlled or needs to be controlled before performing the measurements (e.g the system is unstable). Then indirect methods are used to identify the plant, computing a closed loop transfer function of the stabilized plant and deriving $H(f)$ from it. The first technique, called two points indirect method, is used when the controller transfer function $C(f)$ is known or measurable. It consist of applying a disturbance $d(t)$ to the system and measure the plant input $u(t)$. The closed loop transfer function between u and d is:

$$\begin{aligned} U(f) &= D(f) + C(f)R(f) - C(f)H(f)(U(f) + N(f)) \\ &= \frac{1}{1 + C(f)H(f)}D(f) + \frac{C(f)}{1 + C(f)H(f)}R(f) + \frac{1}{1 + C(f)H(f)}N(f) \\ U(f) &= S(f)R(f) + S(f)(D(f) + N(f)) \end{aligned} \tag{58}$$

where $S(f)$ is the sensitivity transfer function. Multiplying each term with the complex conjugate of the disturbance we will obtain:

$$S_{ud}(f) = S(f)S_{rd}(f) + S(f)(S_{dd}(f) + S_{nd}(f)) \tag{59}$$

All the other inputs are uncorrelated to the disturbance, then the sensitivity of the system can be obtained by:

$$S(f) \approx \frac{S_{ud}(f)}{S_{dd}(f)} \tag{60}$$

Since the controller $C(f)$ is known, the plant can be identified as:

$$H(f) \approx \frac{1 - S(f)}{C(f)S(f)} \tag{61}$$

In order to obtain a good estimate of $H(f)$ the disturbance $d(t)$ should be white noise and no reference should be given. In this way the disturbance and the other inputs will be strongly uncorrelated. If the controller acting

on the system is unknown the presented method is no more reliable and it is better to use so called three point indirect method. The procedure is the same of the two point method but in addition also the tracking error $e(t)$ or the system output $y(t)$ is measured. Knowing the disturbance, the controller input and the tracking error allows to derive both the sensitivity and the process sensitivity transfer function from which it is possible to identify the plant $H(f)$. Given the input $d(t)$ and the measured signals $u(t)$ and $e(t)$, the close loop transfer functions are given by:

$$\begin{aligned} U(f) &= \frac{1}{1+C(f)H(f)}D(f) + \frac{C(f)}{1+C(f)H(f)}R(f) + \frac{1}{1+C(f)H(f)}N(f) \\ E(f) &= \frac{1}{1+C(f)H(f)}R(f) - \frac{1(f)}{1+C(f)H(f)}N(f) - \frac{H(f)}{1+C(f)H(f)}D(f) \end{aligned} \tag{62}$$

That, if the reference and the noise are unrelated to the injected disturbance, becomes:

$$\begin{aligned} S_{ud}(f) &\approx S(f)S_{dd}(f) \\ S_{ed}(f) &\approx -P_s(f)S_{dd}(f) \end{aligned} \tag{63}$$

Then the plant transfer function can be derived as:

$$H(f) \approx \frac{P_s(f)}{S_(f)} = -\frac{S_{ed}(f)}{S_{ud}(f)} \tag{64}$$

BIBLIOGRAPHY

[1] H.K. Tonshoff et al. "Developments and Trends in Monitoring and Control of Machining Processes." In: *CIRP Annals - Manufacturing Technology* 37.2 (1988), pp. 611 –622.

[2] G. Byrne, David Dornfeld, and B. Denkena. "Advancing Cutting Technology." In: (2003).

[3] Delcam. *CAM Softaware*. 2013. URL: http://www.powermill.com/.

[4] Siemens. *CAM Softaware*. 2013. URL: http://www.plm.automation.siemens.com/.

[5] OPEN MIND. *CAM Software*. 2013. URL: http://www.openmind-tech.com/.

[6] CGTech. *Vericut CAM Software*. 2013. URL: http://www.cgtech.com/.

[7] Manufacturing Automation Laboratories. *CutPro Machining Software*. 2013. URL: http://www.malinc.com/.

[8] Caron Engineering. *TMAC Tool Monitoring Adaptive Control*. 2013. URL: http://www.caron-eng.com/.

[9] Heidenhain GmbH. *Efficiency Dynamics*. 2013. URL: http://www.heidenhain.com/de_DE/php/dokumentation-und-information/dokumentation/prospekte/popup/media/media/file/view/file-0671/file.pdf.

[10] Omative. *ACM Adaptive Control Monitor*. 2013. URL: http://www.omative.com/image/users/173890/ftp/my_files/English/Documents/Brochures_Presentations/OMATIVE_ACM_Brochure.pdf.

[11] Artis. *Tool and Process Monitoring*. 2013. URL: http://www.artis.de/en/products/artis-process-monitoring/ctm/.

[12] Nordmann. *Tool Monitoring and Process Control*. 2013. URL: http://www.nordmann.eu/usa/produkte.html.

[13] Omative. *VCM Vibration Control Monitor*. 2013. URL: http://www.omative.com/image/users/173890/ftp/my_files/English/Documents/Brochures_Presentations/VCM_Brochure.pdf.

[14] PROMETEC. *PROMOS2*. 2013. URL: http://www.prometec.de/download/datasheets_machining/PROMOS2.GB.pdf.

[15] K. Jemielniak. "Commercial Tool Condition Monitoring Systems." In: *The International Journal of Advanced Manufacturing Technology* 15.10 (1999), pp. 711–721.

[16] Renishaw. *Touch Probes*. 2013. URL: http://www.renishaw.com/en/machine-tool-probes-and-software--6073.

[17] R. Neugebauer, B. Denkena, and K. Wegener. "Mechatronic Systems for Machine Tools." In: *CIRP Annals - Manufacturing Technology* 56.2 (2007), pp. 657 –686.

[18] "An Overview of Modeling and Simulation of the Milling Process." In: *Journal of Engineering for Industry* 113.2 (May 1991), pp. 169–175.

[19] Y. Altintas. *Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design*. Cambridge University Press, 2012.

[20] J. V. Abellan-Nebot and F. Romero Subiron. "A review of machining monitoring systems based on artificial intelligence process models." In: *The International Journal of Advanced Manufacturing Technology* 47.1-4 (2010), pp. 237–257.

[21] Ashanira Mat Deris, Azlan Mohd Zain, and Roselina Sallehuddin. "Overview of Support Vector Machine in Modeling Machining Performances." In: *Procedia Engineering* 24.0 (2011). International Conference on Advances in Engineering, pp. 308 –312.

[22] Steven Y. Liang, Rogelio L. Hecker, and Robert G. Landers. "Machining Process Monitoring and Control: The State-of-the-Art." In: *Journal of Manufacturing Science and Engineering* 126.2 (July 2004), pp. 297–310.

[23] R. Teti et al. "Advanced monitoring of machining operations." In: *CIRP Annals - Manufacturing Technology* 59.2 (2010), pp. 717 –739.

[24] J. Tlusty and G.C. Andrews. "A Critical Review of Sensors for Unmanned Machining." In: *CIRP Annals - Manufacturing Technology* 32.2 (1983), pp. 563 –572.

[25] G. Totis and M. Sortino. "Development of a modular dynamometer for triaxial cutting force measurement in turning." In: *International Journal of Machine Tools and Manufacture* 51.1 (2011), pp. 34 –42.

[26] G. Totis et al. "Development of an innovative plate dynamometer for advanced milling and drilling applications." In: *Measurement* 49.0 (2014), pp. 164 –181.

[27] Süleyman Yaldiz et al. "Design, development and testing of a four-component milling dynamometer for the measurement of cutting force and torque." In: *Mechanical Systems and Signal Processing* 21.3 (2007), pp. 1499 –1511.

[28] Lei Ma, ShreyesN. Melkote, and JamesB. Castle. "PVDF sensor-based monitoring of milling torque." In: *The International Journal of Advanced Manufacturing Technology* (2013), pp. 1–12.

[29] A.A.D. Sarhan et al. "Compensation of machine tool spindle error motions in the radial direction for accurate monitoring of cutting forces utilizing sensitive displacement sensors." In: *Proceedings of the World Congress on Engineering* 1 (2011).

[30] Saurabh Aggarwal, Nenad Nešić, and Paul Xirouchakis. "Cutting torque and tangential cutting force coefficient identification from spindle motor current." In: *The International Journal of Advanced Manufacturing Technology* 65.1-4 (2013), pp. 81–95.

[31] Y. Kakinuma and T. Kamigochi. "External Sensor-Less Tool Contact Detection by Cutting Force Observer." In: *Procedia CIRP* 2.0 (2012). 1st CIRP Global Web Conference: Interdisciplinary Research in Production Engineering (CIRPE2012), pp. 44 –48.

[32] Matsubara Atsushi and Ibaraki Soichi. "Monitoring and Control of Cutting Forces in Machining Processes: A Review." In: *International Journal of Automation Technology* 3.4 (2009), pp. 445–456.

[33] Xiaoli Li. "A brief review: acoustic emission method for tool wear monitoring during turning." In: *International Journal of Machine Tools and Manufacture* 42.2 (2002), pp. 157 –165.

[34] Guillem Quintana and Joaquim Ciurana. "Chatter in machining processes: A review." In: *International Journal of Machine Tools and Manufacture* 51.5 (2011), pp. 363 –376.

[35] S. Dutta et al. "Application of digital image processing in tool condition monitoring: A review." In: *CIRP Journal of Manufacturing Science and Technology* 6.3 (2013), pp. 212 –232.

[36] Yuming Qin and S.S. Park. "Robust adaptive control of machining operations." In: *Mechatronics and Automation, 2005 IEEE International Conference*. Vol. 2. 2005, 975–979 Vol. 2.

[37] L. Rubio, M. De la Sen, and A. Bilbao-Guillerna. In: *Discrete-time adaptive control of milling forces using fractional order holds by on-line adjustment of the correcting gain*. 2007, pp. 1330–1335.

[38] Junhong Zhang, Guangjian Ni, and Jianping Zhang. "Neural network based pid control on nc milling process with high-speed." In: *Technology and Innovation Conference, 2006. ITIC 2006. International*. 2006, pp. 551–555.

[39] P.B. Huang et al. "An in-process adaptive control of surface roughness in end milling operations." In: *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on*. Vol. 3. 2010, pp. 1191–1194.

[40] Zhiyong Chen, Hai-Tao Zhang, and Han Ding. "Mitigation of chatter instability in milling processes by active fourier series compensation." In: *Automation Science and Engineering (CASE), 2012 IEEE International Conference on*. 2012, pp. 167–171.

[41] B.Srinivasa Prasad et al. "Condition monitoring of CNC machining using adaptive control." In: *International Journal of Automation and Computing* 10.3 (2013), pp. 202–209.

[42] Xifan Yao et al. "Machining force control with intelligent compensation." In: *The International Journal of Advanced Manufacturing Technology* 69.5-8 (2013), pp. 1701–1715.

[43] U. Zuperl, E. Kiker, and K. Jezernik. "Adaptive Force Control in High-Speed Machining by Using a System of Neural Networks." In: *IEEE International Symposium on Industrial Electronics*. Vol. 1. 2006, pp. 148–153.

[44] U. Zuperl, F. Cus, and M. Reibenschuh. "Modeling and adaptive force control of milling by using artificial techniques." In: *Journal of Intelligent Manufacturing* 23.5 (2012), pp. 1805–1815.

[45] S. Saikumar and M.S. Shunmugam. "Development of a feed rate adaption control system for high-speed rough and finish end-milling of hardened EN24 steel." In: *The International Jour-*

nal of Advanced Manufacturing Technology 59.9-12 (2012), pp. 869–884.

[46] Lieh-Dai Yang et al. "Fuzzy-nets-based in-process surface roughness adaptive control system in end-milling operations." In: *The International Journal of Advanced Manufacturing Technology* 28.3-4 (2006), pp. 236–248.

[47] Yan Tang, Robert G. Landers, and S.N. Balakrishnan. "Hierarchical optimal force position and contour control of machining processes." In: *Control Engineering Practice* 14.8 (2006), pp. 909 –922.

[48] Luis Rubio et al. "Model-based expert system to automatically adapt milling forces in Pareto optimal multi-objective working points." In: *Expert Systems with Applications* 40.6 (2013), pp. 2312 –2322.

[49] P. Bosetti, M. Leonesio, and P. Parenti. "On Development of an Optimal Control System for Real-time Process Optimization on Milling Machine Tools." In: *Procedia CIRP* 12.0 (2013). Eighth CIRP Conference on Intelligent Computation in Manufacturing Engineering, pp. 31 –36.

[50] B. Denkena and F. Flöter. "Adaptive Cutting Force Control on a Milling Machine with Hybrid Axis Configuration." In: *Procedia CIRP* 4.0 (2012). 3rd CIRP Conference on Process Machine Interactions, pp. 109 –114.

[51] Adam G. Rehorn, Jin Jiang, and Peter E. Orban. "State-of-the-art methods and results in tool condition monitoring: a review." In: *The International Journal of Advanced Manufacturing Technology* 26.7-8 (2005), pp. 693–710.

[52] A. Siddhpura and R. Paurobally. "A review of flank wear prediction methods for tool condition monitoring in a turning process." In: *The International Journal of Advanced Manufacturing Technology* 65.1-4 (2013), pp. 371–393.

[53] Chen Zhang and Jilin Zhang. "On-line tool wear measurement for ball-end milling cutter based on machine vision." In: *Computers in Industry* 64.6 (2013), pp. 708 –719.

[54] Soichi Ibaraki and Takuya Shimizu. "A long term control scheme of cutting forces to regulate tool life in end milling processes." In: *Precision Engineering* 34.4 (2010), pp. 675 –682.

[55] François Girardin, Didier Remond, and Jean-François Rigal. "Tool wear detection in milling: An original approach with a non-dedicated sensor." In: *Mechanical Systems and Signal Processing* 24.6 (2010), pp. 1907 –1920.

[56] L. Pejryd, J. Repo, and T. Beno. "Machine Tool Internal Encoders as Sensors for the Detection of Tool Wear." In: *Procedia CIRP* 4.0 (2012). 3rd CIRP Conference on Process Machine Interactions, pp. 46 –51.

[57] M. Ritou et al. "Angular approach combined to mechanical model for tool breakage detection by eddy current sensors." In: *Mechanical Systems and Signal Processing* 44.1-2 (2014). Special Issue on Instantaneous Angular Speed (IAS) Processing and Angular Applications, pp. 211 –220.

[58] Guofeng Wang, Zhiwei Guo, and Yinwei Yang. "Force sensor based online tool wear monitoring using distributed Gaussian ARTMAP network." In: *Sensors and Actuators A: Physical* 192.0 (2013), pp. 111 –118.

[59] C. Doukas et al. "On the Estimation of Tool-wear for Milling Operations based on Multi-Sensorial Data." In: *Procedia CIRP* 8.0 (2013). 14th CIRP Conference on Modeling of Machining Operations (CIRP CMMO), pp. 415 –420.

[60] PaulW. Prickett, RaeesA. Siddiqui, and RogerI. Grosvenor. "A microcontroller-based end milling cutter monitoring and management system." In: *The International Journal of Advanced Manufacturing Technology* 55.9-12 (2011), pp. 855–867.

[61] C.S. Ai et al. "The milling tool wear monitoring using the acoustic spectrum." In: *The International Journal of Advanced Manufacturing Technology* 61.5-8 (2012), pp. 457–463.

[62] Sohyung Cho, Sultan Binsaeid, and Shihab Asfour. "Design of multisensor fusion-based tool condition monitoring system in end milling." In: *The International Journal of Advanced Manufacturing Technology* 46.5-8 (2010), pp. 681–694.

[63] D.A. Tobon-Mejia, K. Medjaher, and N. Zerhouni. "CNC machine tool's wear diagnostic and prognostic by using dynamic Bayesian networks." In: *Mechanical Systems and Signal Processing* 28.0 (2012). Interdisciplinary and Integration Aspects in Structural Health Monitoring, pp. 167 –182.

[64] Y. Altintas and M. Weck. "Chatter Stability of Metal Cutting and Grinding." In: *CIRP Annals - Manufacturing Technology* 53.2 (2004), pp. 619 –642.

[65] Y. Altintas and E. Budak. "Analytical Prediction of Stability Lobes in Milling." In: *CIRP Annals - Manufacturing Technology* 44.1 (1995), pp. 357 –362.

[66] Hongrui Cao, Bing Li, and Zhengjia He. "Chatter stability of milling with speed-varying dynamics of spindles." In: *International Journal of Machine Tools and Manufacture* 52.1 (2012), pp. 50 –58.

[67] I. Zaghbani and V. Songmene. "Estimation of machine-tool dynamic parameters during machining operation through operational modal analysis." In: *International Journal of Machine Tools and Manufacture* 49.12-13 (2009), pp. 947 –957.

[68] Bin Li et al. "Estimation of CNC machine-tool dynamic parameters based on random cutting excitation through operational modal analysis." In: *International Journal of Machine Tools and Manufacture* 71.0 (2013), pp. 26 –40.

[69] HongQi Liu et al. "On-line chatter detection using servo motor current signal in turning." In: *Science China Technological Sciences* 54.12 (2011), pp. 3119–3129.

[70] GuangFei Jia et al. "A synthetic criterion for early recognition of cutting chatter." In: *Science China Technological Sciences* 56.11 (2013), pp. 2870–2876.

[71] M. Lamraoui et al. "Indicators for monitoring chatter in milling based on instantaneous angular speeds." In: *Mechanical Systems and Signal Processing* 44.1-2 (2014). Special Issue on Instantaneous Angular Speed (IAS) Processing and Angular Applications, pp. 72 –85.

[72] M. Lamraoui et al. "Cyclostationarity analysis of instantaneous angular speeds for monitoring chatter in high speed milling." In: *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*. 2012, pp. 3868–3873.

[73] E. Abele, T. Sielaff, and A. Schiffler. "Method for chatter detection with standard PLC systems." In: *Production Engineering* 6.6 (2012), pp. 611–619.

[74] Somkiat Tangjitsitcharoen and Narongsak Pongsathornwiwat. "Development of chatter detection in milling processes." In: *The International Journal of Advanced Manufacturing Technology* 65.5-8 (2013), pp. 919–927.

[75] Somkiat Tangjitsitcharoen, Tanintorn Saksri, and Suthas Ratanakuakangwan. "Advance in chatter detection in ball end milling process by utilizing wavelet transform." In: *Journal of Intelligent Manufacturing* (2013), pp. 1–15.

[76] Dong-Hoon Kim et al. "The development of embedded device to detect chatter vibration in machine tools and CNC-based autonomous compensation." In: *Journal of Mechanical Science and Technology* 25.10 (2011), pp. 2623–2630.

[77] P. Albertelli et al. "Spindle speed variation in turning: technological effectiveness and applicability to real industrial cases." In: *The International Journal of Advanced Manufacturing Technology* 62.1-4 (2012), pp. 59–67.

[78] N. J. M. van Dijk et al. "Automatic In-Process Chatter Avoidance in the High-Speed Milling Process." In: *Journal of Dynamic Systems, Measurement, and Control* 132.3 (Apr. 2010), pp. 031006–031006.

[79] Yasuhiro Kakinuma, Yui Sudo, and Tojiro Aoyama. "Detection of chatter vibration in end milling applying disturbance observer." In: *CIRP Annals - Manufacturing Technology* 60.1 (2011), pp. 109 –112.

[80] T. Yoneoka et al. "Disturbance Observer-Based In-process Detection and Suppression of Chatter Vibration." In: *Procedia CIRP* 1.0 (2012). Fifth CIRP Conference on High Performance Cutting 2012, pp. 44 –49.

[81] Hiromitsu Morita and Toru Yamashita. "Tracing and Visualizing Variation of Chatter for In-Process Identification of Preferred Spindle Speeds." In: *Procedia CIRP* 4.0 (2012). 3rd CIRP Conference on Process Machine Interactions, pp. 11 –16.

[82] B. Denkena et al. "Model based reconstruction of milled surface topography from measured cutting forces." In: *International Journal of Machine Tools and Manufacture* 54-55.0 (2012), pp. 25 –33.

[83] Jean Philippe Costes and Vincent Moreau. "Surface roughness prediction in milling based on tool displacements." In: *Journal of Manufacturing Processes* 13.2 (2011), pp. 133 –140.

[84] Guillem Quintana et al. "Using kernel data in machine tools for the indirect evaluation of surface roughness in vertical milling operations." In: *Robotics and Computer-Integrated Manufacturing* 27.6 (2011), pp. 1011 –1018.

[85] Christian Brecher et al. "Use of NC kernel data for surface roughness monitoring in milling operations." In: *The International Journal of Advanced Manufacturing Technology* 53.9-12 (2011), pp. 953–962.

[86] E. García-Plaza et al. "Surface Finish Monitoring in Taper Turning CNC Using Artificial Neural Network and Multiple Regression Methods." In: *Procedia Engineering* 63.0 (2013). The Manufacturing Engineering Society International Conference, MESIC 2013, pp. 599 –607.

[87] İlhan Asiltürk and Mehmet çunkaş. "Modeling and prediction of surface roughness in turning operations using artificial neural network and multiple regression method." In: *Expert Systems with Applications* 38.5 (2011), pp. 5826 –5832.

[88] Marek Vrabel et al. "Surface Roughness Prediction using Artificial Neural Networks when Drilling Udimet 720." In: *Procedia Engineering* 48.0 (2012). Modelling of Mechanical and Mechatronics Systems, pp. 693 –700.

[89] H.K. Abdul-Ameer, G.A. Al-Kindi, and H. Zughaer. "Towards computer vision feedback for enhanced CNC machining." In: *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*. 2011, pp. 754–760.

[90] J. Lee, M. Ghaffari, and S. Elmeligy. "Self-maintenance and engineering immune systems: Towards smarter machines and manufacturing systems." In: *Annual Reviews in Control* 35.1 (2011), pp. 111 –122.

[91] Jay Lee et al. "Prognostics and health management design for rotary machinery systems-Reviews, methodology and applications." In: *Mechanical Systems and Signal Processing* 42.1-2 (2014), pp. 314 –334.

[92] Jiao Wang and Meiling Qi. "Application of intelligent fault diagnosis technology in NC machine tool fault diagnosis." In: *Electronics and Optoelectronics (ICEOE), 2011 International Conference on*. Vol. 4. 2011, pp. V4–424–V4–427.

[93] I. Bediaga et al. "An integrated system for machine tool spindle head ball bearing fault detection and diagnosis." In: *Instrumentation Measurement Magazine, IEEE* 16.2 (2013), pp. 42–47.

[94] ClaudiuFlorinel Bisu et al. "Envelope dynamic analysis: a new approach for milling process monitoring." In: *The International Journal of Advanced Manufacturing Technology* 62.5-8 (2012), pp. 471–486.

[95] R. Neugebauer, J. Fischer, and M. Praedicow. "Condition-based preventive maintenance of main spindles." In: *Production Engineering* 5.1 (2011), pp. 95–102.

[96] Linxia Liao and Jay Lee. "Design of a reconfigurable prognostics platform for machine tools." In: *Expert Systems with Applications* 37.1 (2010), pp. 240 –252.

[97] *Study Federal Energy Management Program USA*. 2006. URL: `http://www1.eere.energy.gov/femp/pdfs/OM_5.pdf`.

[98] Cyril M Harris and Allan G Piersol. *Harris' shock and vibration handbook*. Vol. 5. McGraw-Hill New York, 2002.

[99] ChrisK Mechefske. "Machine Condition Monitoring and Fault Diagnostics." In: *Vibration and Shock Handbook*. CRC Press, 2005, pp. 25–1–25–35.

[100] ISO. *Mechanical vibration - Evaluation of machine vibration by measurements on non-rotating parts - Part 1: General guidelines*. ISO 10816-1:2007. Geneva, Switzerland: International Organization for Standardization, 2012.

[101] N Tandon and A Choudhury. "A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings." In: *Tribology International* 32.8 (1999), pp. 469 –480.

[102] Jérôme Antoni. "The spectral kurtosis: a useful tool for characterising non-stationary signals." In: *Mechanical Systems and Signal Processing* 20.2 (2006), pp. 282 –307.

[103] Jérôme Antoni and R.B. Randall. "The spectral kurtosis: application to the vibratory surveillance and diagnostics of rotating machines." In: *Mechanical Systems and Signal Processing* 20.2 (2006), pp. 308 –331.

[104] Robert B. Randall and Jérôme Antoni. "Rolling element bearing diagnostics-A tutorial." In: *Mechanical Systems and Signal Processing* 25.2 (2011), pp. 485 –520.

[105] Tomasz Barszcz and Robert B. Randall. "Application of spectral kurtosis for detection of a tooth crack in the planetary gear of a wind turbine." In: *Mechanical Systems and Signal Processing* 23.4 (2009), pp. 1352 –1365.

[106] Len Gelman et al. "Novel Technology Based on the Spectral Kurtosis and Wavelet Transform for Rolling Bearing Diagnosis." In: *International Journal of Prognostics and Health Management* 4.2 (2013).

[107] M.F. Heertjes, X.G.P. Schuurbiers, and H. Nijmeijer. "Performance-Improved Design of N-PID Controlled Motion Systems With Applications to Wafer Stages." In: *Industrial Electronics, IEEE Transactions on* 56.5 (2009), pp. 1347–1355.

[108] Hassan Khalil. *Nonlinear Systems*. Prentice Hall, Jan. 2002.

[109] Mark Haring, Nathan van de Wouw, and Dragan Nešić. "Extremum-seeking control for nonlinear systems with periodic steady-state outputs." In: *Automatica* 49.6 (2013), pp. 1883 –1891.

[110] Y. Tan et al. "Extremum seeking from 1922 to 2010." In: *Control Conference (CCC), 2010 29th Chinese*. 2010, pp. 14–26.

[111] B.G.B. Hunnekens et al. "Steady-state performance optimization for variable-gain motion control using extremum seeking." In: *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*. 2012, pp. 3796–3801.

[112] K.J. Aström and R.M. Murray. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, 2010.

[113] M. Steinbuch and M.L. Norg. "Advanced Motion Control: An Industrial Perspective." In: *European Journal of Control* 4.4 (1998), pp. 278 –293.

[114] D. Bruijnen, R. van de Molengraft, and M. Steinbuch. "Optimization aided loop shaping for motion systems." In: *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*. 2006, pp. 255–260.

[115] J. Maciejowski and G. Vinnecombe. *Multivariable Feedback Design*. Pearson Education Canada, 2000.

[116] Demidovich B.P. "Dissipativity of a system of nonlinear differential equations." In: *Ser.Mat. Mekh.* Part I—6 (1961) 19–27; Part II—1 (1962) 3–8 (in Russian) (1961).

[117] Yakubovich V.A. "Matrix inequalities method in stability theory for nonlinear control systems: I. absolute stability of forced vibrations." In: *Automation and Remote Control* 7 (1964), pp. 905–917.

[118] S.H. Suh. *Theory and Design of CNC Systems*. Advanced Manufacturing Series. Springer, 2008.

[119] S. Kalpakjian and S.R. Schmid. *Manufacturing Engineering and Technology*. Prentice Hall PTR, 2010.

[120] Mustafa Kurt and Eyup Bagci. "Feedrate optimisation/scheduling on sculptured surface machining: a comprehensive review, applications and future directions." In: *The International Journal of Advanced Manufacturing Technology* 55.9-12 (2011), pp. 1037–1067.

[121] N. J M Van Dijk, N. Van de Wouw, and H. Nijmeijer. "Low-order control design for chatter suppression in high-speed milling." In: *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. 2011, pp. 663–668.

[122] M. Lamraoui, M. Thomas, and M. El Badaoui. "Cyclostationarity approach for monitoring chatter and tool wear in high speed milling." In: *Mechanical Systems and Signal Processing* 44.1-2 (2014). Special Issue on Instantaneous Angular Speed (IAS) Processing and Angular Applications, pp. 177 –198.

[123] Jérémie Monnin, Fredy Kuster, and Konrad Wegener. "Optimal control for chatter mitigation in milling-Part 1: Modeling and control design." In: *Control Engineering Practice* 0 (2013), pp. –.

[124] Jérémie Monnin, Fredy Kuster, and Konrad Wegener. "Optimal control for chatter mitigation in milling-Part 2: Experimental validation." In: *Control Engineering Practice* 0 (2013), pp. –.

[125] G. Quintana, M.L. Garcia-Romeu, and J. Ciurana. "Surface roughness monitoring application based on artificial neural networks for ball-end milling operations." In: *Journal of Intelligent Manufacturing* 22.4 (2011), pp. 607–617.

[126] S. Das et al. "Essential steps in prognostic health management." In: *Prognostics and Health Management (PHM), 2011 IEEE Conference on*. 2011, pp. 1–9.

[127] Xavier Desforges, Abdallah Habbadi, and Bernard Archimède. "Design methodology for smart actuator services for machine tool and machining control and monitoring." In: *Robotics and Computer-Integrated Manufacturing* 27.6 (2011), pp. 963 –976.

[128] Petre Stoica and Randolph L Moses. *Introduction to spectral analysis*. Vol. 1. Prentice hall New Jersey: 1997.

[129] G. Witvoet D.J.H. Bruijnen. *ShapeIt: Loopshaping Tool for Motion System*. `http://http://cstwiki.wtb.tue.nl/index.php?title=Home_of_ShapeIt`. [Online; accessed 21-October-2013]. 2006.