BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF MECHANICAL ENGINEERING

FAKULTA STROJNÍHO INŽENÝRSTVÍ

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY

PREDICTIVE MAINTENANCE OF PNEUMATIC PISTONS

MOŽNOSTI PREDIKTIVNÍ ÚDRŽBY PNEUMATICKÝCH PÍSTŮ

MASTER'S THESIS

DIPLOMOVÁ PRÁCE

AUTHOR | Bc. Artyom Voronin

AUTOR PRÁCE

SUPERVISOR | Ing. Martin Brablc

VEDOUCÍ PRÁCE

BRNO 2021

Assignment Master's Thesis

Institut:	Institute of Solid Mechanics, Mechatronics and Biomechanics
Student:	**Bc. Artyom Voronin**
Degree programm:	Applied Sciences in Engineering
Branch:	Mechatronics
Supervisor:	**Ing. Martin Brablc**
Academic year:	2020/21

As provided for by the Act No. 111/98 Coll. on higher education institutions and the BUT Study and Examination Regulations, the director of the Institute hereby assigns the following topic of Master's Thesis:

Predictive maintenance of pneumatic pistons

Brief Description:

With the ever–increasing degree of automation in the industry, a widespread effort to measure, record, and exploit information and signals related to the state of a given machine and its production quality, is becoming more relevant. Predictive Maintenance (PM) is a relatively new method, which builds on and further expands the ideas of the already established Fault Detection and Analysis (FDA). The purpose of this work is to demonstrate various approaches to Predictive Maintenance (e.g., signal–based and model–based) using the Matlab/Simulink software tools on a double–acting pneumatic piston as a case–study.

Master's Thesis goals:

1. Conduct research in the area of Predictive Maintenance, Fault Detection and Analysis, and related approaches and try to define their similarities and differences. Provide a practical demonstration for each of the approaches.
2. Create a simulation model of the demonstration device, including models of the sensors. Test different methods to create the model (e.g., software simulation, physical properties, black–box identification, etc.) and identify the models with real data.
3. Apply Predictive Maintenance techniques to a test dataset without using a simulation model.
4. Apply Predictive Maintenance techniques to a test dataset using a simulation model.
5. Evaluate the suitability of each approach for the application of PM and FDA.

Recommended bibliography:

PRITCHARD, Philip J. Introduction to Fluid Mechanics, 9th edition, Wiley, ISBN 978-1118921876.

NELLES, Oliver. Nonlinear system identification: from classical approaches to neural networks and fuzzy models. Berlin: Springer, 2011. ISBN 978-364-2086-748.

LJUNG, Lennart. System identification: theory for the user. 2nd ed. Upper Saddle River, NJ: Prentice Hall PTR, 1999. ISBN 978-0136566953.

VALÁŠEK, Michael. Mechatronika. Dot. 1. vyd. Praha: České vysoké učení technické, 1996. ISBN 80-010-1276-X.

NOSKIEVIČ, Petr. Modelování a identifikace systémů. Ostrava: Montanex, 1999. ISBN 80-722-50-0-2.

Deadline for submission Master's Thesis is given by the Schedule of the Academic year 2020/21

In Brno,

L. S.

prof. Ing. Jindřich Petruška, CSc. Director of the Institute	doc. Ing. Jaroslav Katolický, Ph.D. FME dean

Abstrakt

...Abstrakt...

Summary

...anglicky...

Klíčová slova

...klicova slova...

Keywords

...anglicky...

Bibliographic citation

VORONIN, A. *Predictive maintenance of pneumatic pistons* . Brno: Brno University of Technology, Faculty of Mechanical Engineering, 2021. **??** pages, Master's thesis supervisor: Martin Brablc.

Prohlášení...

Artyom Voronin

Brno . . . .

Poděkování..

Artyom Voronin

Contents

1 Introduction

The production process always included elements of fault control and online monitoring. From the first methods of fault detection, such as visual inspection, today's factories move to automated systems consisting of sensors and computing units to evaluate the faults. Sometimes it is critical to monitor processing equipment in real-time to prevent damage caused by fault or anomaly. Every single fault can cause a slowing down of the production process and thus reducing the profit.

Device real-time monitoring algorithms have formed the Fault Detection and Analysis (FDA) field. FDA methods, in most cases, do not require machine learning techniques and can detect failures, using fundamental algorithms from Fourier analysis and trend checking algorithms to more complex techniques such as Gaussian Mixture Models.

Due to the amount of data collected in recent years and the expansion of data storage technology as cloud services and computation efficiency, it has become possible to use more advanced algorithms for fault detection and analysis. Using classification machine learning techniques, it is possible to isolate where does the fault occur. Another option that becomes available with a large amount of data is to estimate the remaining useful life (RUL) of the entire system. These techniques have led to predictive maintenance as an effort for optimal maintenance solutions. The current technical condition of the equipment is always available by information extracted from measured signals. It is possible to use current system conditions to estimate remaining useful life in time or distance measurements such as days, kilometers, or cycles. Estimated residual lifetime gives an option to plan maintenance concerning actual system conditions.

These remaining useful life estimation algorithms, the fault detection methods and system modeling and identification techniques form a new predictive maintenance field.

System modeling allows providing experiments and developing solutions offline before physical hardware implementations. Unavailable or challenging to implement measurements can be replaced by generated data from the simulation model and finally helps to deploy a robust algorithm.

This thesis provides a brief introduction to fault detection and predictive maintenance methodologies and a basic terminology. The 2 chapter describes the main goal and problems in these areas and focuses on similarities and differences between these two approaches.

Developing the simulation model of the double-acting pneumatic actuator and comparing it with the real-life equipment using different approaches is described in chapter 3, 4, and 5.

The following chapter 6 illustrates signal-based predictive maintenance methods using different sensors available in a demonstration device. Appling preprocessing, feature extraction, and classification model, sensors were evaluated in terms of functionality, accuracy, and price.

The model-based predictive maintenance techniques and simulation model exploitation are demonstrated in chapter 7. The simulation model is used to determine the residual signals between the measured data and the simulation model's output. Also, using a simulation model, degradation data are generated and used in the remaining useful life estimation.

2 Theoretical Survey

This chapter contains a short introduction to the main goals and problems presented in fault detection and analysis and predictive maintenance techniques. A brief review of methodologies used in these fields and general approaches.

Section digital twin presents scenarios where a simulation model is used in predictive maintenance.

2.1 Problem Definition

In practice many types of machinery require some calibration and monitoring for adequate working. An anomaly or fault detection in time can prevent machinery from damage that causes loss of money due to non-working or destroyed equipment. Predicting where the fault appears reduces the cost of diagnosis and replacement operations. The possibility of estimating the remaining useful life allows to optimize a maintenance process and reduce maintenance costs.

Smart manufacturing, the combination of sensors, the possibility of preprocessing and extracting useful information from measurements and decision algorithms based on this information, allows increasing production efficiency and significantly reducing maintenance operations.

Types of Maintenance There are three main types of maintenances 2.1. Each following type of maintenance requires increasing complexity of monitoring and decision algorithms:

- Reactive maintenance, where maintenance coming after the life of the system is excess.

- Preventive maintenance is driven item by schedules that may keep the system safe but not optimal from an efficiency/cost perspective.

- Predictive maintenance is an effort to optimize a maintenance strategy.

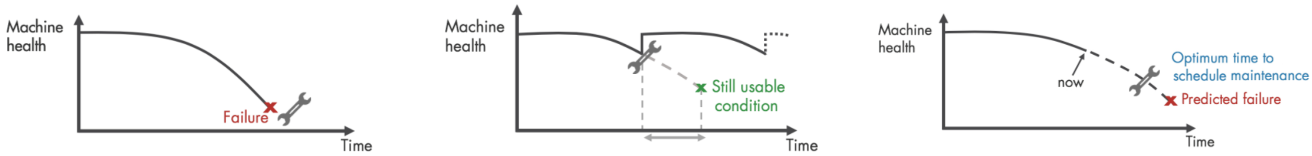

Figure 2.1: Reactive, preventive and predictive types of maintenance

Fault Types A fault is not an acceptable deviation of at least one characteristic or parameter of the system from the standard condition. There are different faults by their sources.

- Plant faults appear in system behavior and cause manufacturing performance.
- Component fault
- Sensor faults occurred in the sensor during measurements.
- Combination of faults

In many cases, faults lead to a system failure and the system is no longer able to perform required functions. There may also be a malfunction after which the system returns to normal operation.

Faults can be classified by the location where they appear, by a fault form, or based on the form in which the fault is added to the system.

2.2 Fault Detection and Analysis (FDA)

Fault Detection and Analysis, FDA (Fault Detection and Isolation, FDI) is a subfield of control engineering focused on detecting the fault and identifying where this fault is located. The main goals of FDI are

- Fault detection, detect anomalies in real-time
- Fault isolation, find the root cause
- Fault identification, estimation of the magnitude, type, or nature of the fault

Several methods are partly overlapped but divided into two main categories.

Signal-Based methods Signal-Based methods (SB), explore measured data and extract useful information in the form of features 2.2. The following methods belong to the SB approach:

- Limit and trend checking
- Spectral analysis
- Data analysis (PCA)
- Pattern recognition

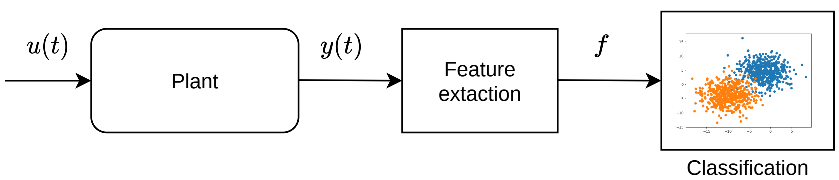

Figure 2.2: Signal-Based Method

Model-Based methods Model-Based methods exploit models identified from real-life systems 2.3. The model-based approach is suitable when it is difficult to gain useful information using only measured signals. If the system structure is known, it is possible to extract features such as state variables or some system parameters. Another option is to compare real system behavior with nominal healthy model and use residuals as inputs to decision algorithms. Typical model-based techniques include

- Residual estimation (compare measurements with "healthy" model)
- Polynomial coefficients
- State variables estimated using state observers
- Parameter estimation

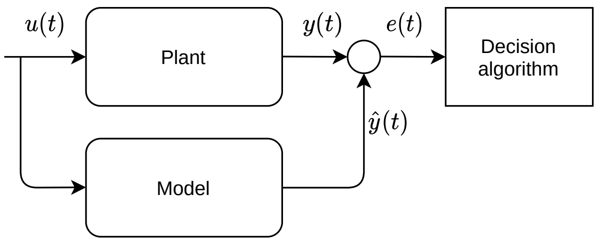

Figure 2.3: Model-Based Method

Automated fault detection depends on input from sensors and postprocessing algorithms. In many manufacturing applications, sensor failures are the most common equipment failure.

The result of FDI is the detection and identification of faults that occur during the operation of the device. Subsequently, predicted faults are processed using fault tolerance and predictive maintenance algorithms.

Fault Tolerance: Provide the system with the hardware architecture and software mechanisms that will allow, if possible, to achieve a given objective in normal operation and given fault situations.

2.3 Predictive maintenance (PdM)

Predictive maintenance (PdM) is cost-effective maintenance strategy that predicts time to failure and warns of an anticipated location where this could occur.

2.3.1 Goals

The are two main goals of Predictive maintenance, RUL (remaining useful life) estimation and identification where the future failure can appear, or what is the reason of decreasing RUL. As a result of PdM is RUL representing of number cycles, days, or some time period before fault occurred. And probability where this fault can appear.

2.3.2 Overview of the PdM development sequence

Figure 2.4 represents the recommended PdM development workflow. The development of predictive maintenance algorithms starts with raw measured signals from sensors. For further working with data, it's an excellent manner to combine measurements to a dataset

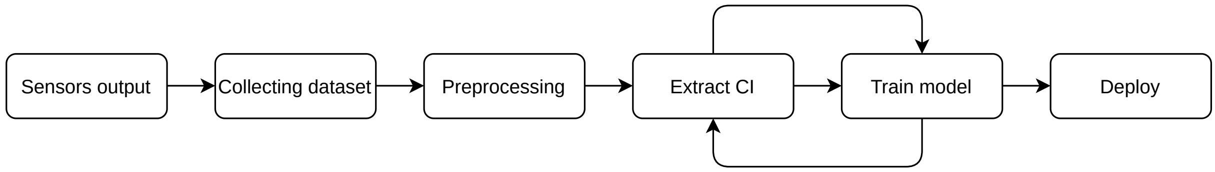

Figure 2.4: Predictive maintenance development sequence

with a logical structure of elements. In this thesis, a common data ensemble structure was used. Each measurement has its data file with all measured signals at a particular time.

If collected data require some preprocessing techniques as data cleaning, smoothing or filter the signal, detrend, normalizing, etc., it can be done at this step.

The next step is to extract condition indicators using predictive maintenance methods described in 2.3.3. Figure out the best combination of CI to train the classification model. As long as the optimal solution not find, try to figure out the best combination of CI described in 2.3.4 and train different classification models iteratively. After the efficient solution is found, deploy the algorithm to work recursively with the study-case system.

2.3.3 Condition Indicators

In the prediction maintenance field, features extracted from measured signals are called **Condition Indicators, CI**.

Condition Indicators represent some system behavior and hide information about system operation conditions. Generally, CI is represented by three main domains. There is a time domain, frequency domain, time-frequency domain. But in fact, CI can be any system parameter or value corresponding with the system's current condition.

The methods of extracting condition indicators from the signal are defined in the same way as in FDI. The **signal-based approach** is suitable when we have measurements from the system in different operating conditions. But there is a problem that signal-based approach enables classifying and learning patterns observed in the training dataset. On the other hand, the **model-based approach** uses physical failure models and does not require a large dataset of failure data. And they may work in situations never observed in data before. Also, the model-based method is helpful in case the measured signal has a more complex relationship with the input signal.

Between common signal-based CI belongs:

- Time-domain: mean, standard deviation, RMS, skewness, etc.
- Frequency-domain: mean frequency, peak values/frequencies, power bandwidth, etc.
- Time-frequency domain: Spectral entropy/kurtosis, moments, etc.

Model-based approach use model properties such as:

- poles and zeros location
- damping coefficient
- state variables values
- modal analysis
- residual values

2.3.4 Condition Indicators Ranking

From each sensor signal, multiple condition indicators can be extracted. A good practice to reduce the number of CI and keep only those who provide essential information.

One of the possibilities is applying Principle Component Analysis (PCA) to transform features from one coordinate system to a new orthogonal basis. Data reduced by using first n principal components that optimally describe the variance of the dataset. Applying the PCA algorithm still requires the extraction of all condition indicators from the signal.

Another option is to rank the futures using the Analysis of Variance (ANOVA) algorithm. This algorithm describes relations among CI in the form of their mean values. The result gives information about how much particular CI represents data. Using first n CI, we reduce the number of CI and reduce the number of extracted features from measured signals. This fact means that using ANOVA reduced the time and complexity of the calculations.

2.3.5 Fault Classification

Classification models are used to recognize faults from a set of CI. The CI The set of CI must contain labels that determine the current condition of the device in the form of fault code, string, etc. The correlation between different CI can be explored using a 2d or 3d scatter plot. The model performance is usually represented by total accuracy and confusion matrix, where on one axis true labels on another predicted from the model. The common types of classification models are:

- Decision Trees
- Supported Vector Machines (SVM)
- Neigherest Neighbors (KNN)
- Ensemble Classifiers
- Neural Networks (ANN)

A good practice is to divide an original dataset of CI to train and test sub-datasets to prevent model overfitting. Choosing the best classification model depends on training data and requires experiments with different models.

2.3.6 Remaining useful life

The remaining useful life is the expected time remaining before the machine requires repair or replacement, and it's a central goal of PdM.

The problem of estimating the remaining useful life is connected with evaluating condition indicators associated with the system's degradation process. These condition indicators must satisfy the requirements for monotonicity, trendability, and prognosability.

The models used to estimate the remaining useful life depend on the historical data we have available. There are three types of possible models.

Survival model The survival model is considered when we have only failure data available, but the whole degradation history is not recorded. The probability density function can be obtained from failure data and used to estimate RUL.

Degradation model The degradation model gives an option to estimate RUL on data without failure moment captured but only recorded degradation process. In this situation, it's necessary to determine a safety threshold that CI shouldn't cross.

Similarity model In case we have a whole history of the degradation process of similar systems, including failure, the similarity model can be used. The upcoming CI is compared with historical degradation paths obtained from the training dataset and evaluated best similarity trend as RUL value.

2.4 Digital twin

A digital twin is a digital representation of the real-life system. Can be represented as a component, a system of components, or as a system of system.

A digital twin can be updated with incoming data from sensors. Fitting model to new data, digital twin represents the current condition state of the real-world object. There are many advantages to using models in PdM. A digital twin can hold historical data about system behavior and can be used for simulation system operation in different conditions, designing control, and simulating future behavior (RUL, "What-if"). Dataset extended by data from the simulation model represent synthetic dataset. This dataset type can contain different measured fault or healthy data of the system and hard to realizable in real-world fault situations.

A mathematical model of the real-world system can be created using different approaches.

- First-principles modeling requires an understanding of the fundamental process of the system.
- Physical modeling (Simscape).
- Data-driven modeling where the system is represented as a Blackbox.
- Combination multiply approaches.

2.5 Comparison PdM and FDA approaches

Figure 2.5 presents a relative arrangement of Predictive Maintenance (PdM) and Fault Detection and Identification (FDI or FDA) algorithms. From the figure, it's clear that Predictive Maintenance it's an extension of the FDI approach, with recommended workflow techniques suitable for optimizing system maintenance.

Both methods are closely overlapped and use quite similar techniques. However, Predictive maintenance over the FDA is extended by RUL estimation. And it leads not only to fault detection at a given moment but to the possible prediction of a fault in the future. Open the possibility of monitoring the system's state not only in the current time but predict near-future behavior too.

2.6 Applications

The most significant interest in PdM is the manufacturing sector that requires efficiency maintenance strategies to increase productivity and reduce money-lost. Another field for

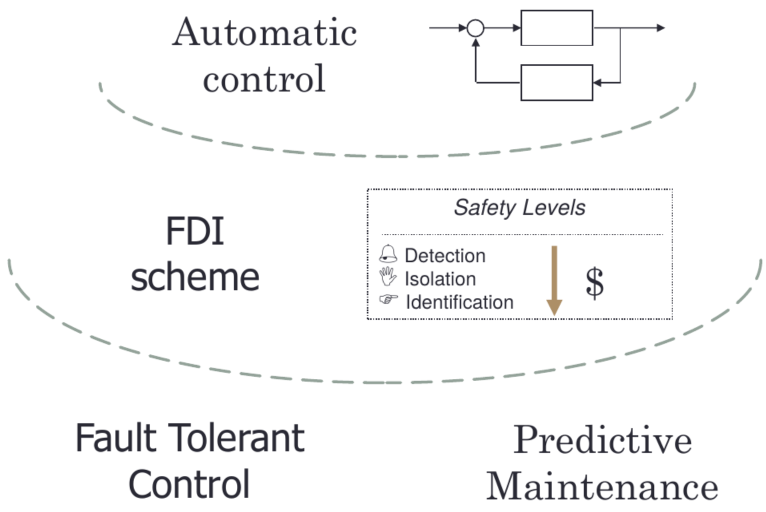

Figure 2.5: Relative arrangement of PdM and FDI algorithm **??**

PdM applications is highly dependent on safety types of machinery such as aircraft or rail industry. Using the PdM condition monitoring, it's possible to prevent unexpected fails. The oil and gas industry supports the PdM field; due to the amount of data collected in these industries, the PdM techniques are beneficial.

3 Demonstration Device Overview

The case study of this thesis is the double-acting pneumatic piston, with a pneumatic circuit and mechanical assembly driven by a piston. Figure **??** is a schematical representation of the system. Figure **??** is a 3D render of the system.

There are seven types of sensors located on the system. Table 1 describes a sensor purpose, signal name in the datastore, and the signal unit.

Sensor	**Unit**	**Description**	**Name**
Encoder	m	displacement measurement	LeverPosition
Encoder	m/s	velocity calculated from displacement	LeverVelocity
Accelerometer	g	accelerometer on moving part	AccelerometerMovin_axisZ/Y
Accelerometer	g	accelerometer on static part	AccelerometerStatic_axisZ/Y
Flow Sensor	l/min	air flow extrusion to A chamber	FlowExtrusion
Flow Sensor	l/min	air flow contraction from A chamber	FlowContraction

Table 3.1: Sensors overview

The dataset measured on the system contains almost five thousand thousand measurements in different operating conditions. Each measurement includes a 10-second recording of moving the pistol up and down. This data was given in the format of massive files with the ".mat" extension, which was divided into files contains only one measurement. The divided dataset is easier to maintain, and Matlab recommends this type of datastores called Data Ensemble **??**.

The measured examples are shown in figures 2,3, and 4.

4 First Principle Modeling (15 pages)

First Principles (White-Box)
Simplification, Liniarization, Reduction, Parameter Estimation.
SimScape (Physical modeling), Simulink (Differential equations).

Data-Driven modeling (Black-Box)
Measurements, Identification.

4.1 Pneumatic piston system overview

4.2 General physical principles

Equation of state Generally $pV = nR_mT$ but for air, using ideal gas constant $R = 287.1[Jkg^{-1}K^{-1}]$ state equation can be rewrite as 4.1.

$$pV = mRT \tag{4.1}$$

Isothermal process For isothermal process 4.2:

$$p_1V_1 = p_2V_2 = const \tag{4.2}$$

Adiabatic process Adiabatic process 4.3:

$$p_1V_1^{\kappa} = p_2V_2^{\kappa} = const \tag{4.3}$$

where $\kappa = c_p/c_v$ is a heat capacity ratio. Another important equation is Mayer's relation $c_p = c_v + R$.

Bernoulli's principle Bernoulli's principle 4.4:

$$H_1 + \frac{mw_1^2}{2} + mgz_1 + Q = H_2 + \frac{mw_2^2}{2} + mgz_w + W_T \tag{4.4}$$

$$H_1 - H_2 = -\int_1^2 V dp = c_p(T_1 - T_2) = c_pT_1(1 - \frac{T_2}{T_1}) \tag{4.5}$$

Differential form:

$$\nu dp + wdw + gdz + dw_T = 0 \tag{4.6}$$

Fluid mechanics Continuity equation 4.7:

$$\dot{m} = S_1w_1\rho_1 = S_2w_2\rho_2 = const \tag{4.7}$$

Air expansion from tank Assuming $W_T = 0, z_1 = z_2, Q = 0$ conditions and combine with 4.4 we will get 4.8 equation:

$$w_2 = \sqrt{2(H_1 - H_2)} \tag{4.8}$$

$$w_2 = \sqrt{2RT_1(\frac{\kappa}{\kappa - 1})(1 - (\frac{p_2}{p_1})^{\frac{\kappa-1}{\kappa}})} \tag{4.9}$$

$$\rho_2 = \frac{p_1}{RT_1}(\frac{p_2}{p_1})^{\frac{1}{\kappa}} \tag{4.10}$$

Together 4.7 4.9 4.10:

$$\dot{m} = Sp_1\sqrt{\frac{2}{RT_1}} \cdot \sqrt{\frac{\kappa}{\kappa - 1}\left(\left(\frac{p_2}{p_1}\right)^{\frac{2}{\kappa}} - \left(\frac{p_2}{p_1}\right)^{\frac{\kappa+1}{\kappa}}\right)} \tag{4.11}$$

where:

$$\psi\left(\frac{p_2}{p_1}\right) = \sqrt{\frac{\kappa}{\kappa - 1}\left(\left(\frac{p_2}{p_1}\right)^{\frac{2}{\kappa}} - \left(\frac{p_2}{p_1}\right)^{\frac{\kappa+1}{\kappa}}\right)} \tag{4.12}$$

Finally 4.13:

$$\dot{m} = Sp_1\sqrt{\frac{2}{RT_1}}\psi\left(\frac{p_2}{p_1}\right) \tag{4.13}$$

Critical flow velocity Speed of sound:

$$c = \sqrt{\frac{dp}{d\rho}} = \sqrt{\frac{\kappa p}{\rho}} = \sqrt{\kappa RT} \tag{4.14}$$

Assume $c = w_2$ (4.9, 4.14) we will get the critical flow velocity:

$$c_2 = w_k = \sqrt{\kappa RT} = \sqrt{2RT_1\frac{\kappa}{\kappa - 1} - 2w_k^2\frac{1}{\kappa - 1}} \tag{4.15}$$

$$w_k^2 = 2RT_1\frac{\kappa}{\kappa - 1} - 2w_k^2\frac{1}{\kappa - 1} \tag{4.16}$$

$$w_k = \sqrt{2RT_1\frac{\kappa}{\kappa - 1}} = \sqrt{2p_1\nu_1\frac{\kappa}{\kappa + 1}} \tag{4.17}$$

For calculating critical pressure ratio assume $w_k = w_2$ 4.17 4.9:

$$\sqrt{2RT_1\frac{\kappa}{\kappa-1}} = \sqrt{2RT_1\frac{\kappa}{\kappa-1}\left(1-\left(\frac{p_2}{p1}\right)^{\frac{\kappa+1}{\kappa}}\right)} \tag{4.18}$$

$$\left(\frac{p_2}{p1}\right)^{\frac{\kappa-1}{\kappa}} = \frac{2}{\kappa+1} \tag{4.19}$$

$$\tag{4.20}$$

$$\left(\frac{p_2}{p1}\right)_k = \left(\frac{p_k}{p1}\right) = \left(\frac{2}{\kappa+1}\right)^{\frac{\kappa}{\kappa-1}} = \beta_k \tag{4.21}$$

Critical pressure condition is $p_k = p_1\beta_k$.
Applying 4.21 to 4.12:

$$\psi_{max}(\beta_k) = \left(\frac{2}{\kappa+1}\right)^{\frac{\kappa}{\kappa-1}}\sqrt{\frac{\kappa}{\kappa+1}} \tag{4.22}$$

For air $\beta_k = 0.528, \psi_{max} = 0.484$
Final equation for ψ:

$$\psi\left(\frac{p_2}{p_1}\right) = \begin{cases} \sqrt{\frac{\kappa}{\kappa-1}\left(\left(\frac{p_2}{p_1}\right)^{\frac{2}{\kappa}} - \left(\frac{p_2}{p_1}\right)^{\frac{\kappa+1}{\kappa}}\right)} & 0.528 < \frac{p_2}{p_1} \leq 1 \\ \left(\frac{2}{\kappa+1}\right)^{\frac{1}{\kappa+1}}\sqrt{\frac{\kappa}{\kappa+1}} & 0 \geq \frac{p2}{p1} \leq 0.528 \end{cases} \tag{4.23}$$

4.3 Pressure model

p_A, p_B	Pa	pressure in chamber A, B
$\dot{m}_A, \dot{m}_B$	$kg \cdot s^{-1}$	mass flow on way to chamber A, B
S_A, S_B	m^2	piston area
V_A, V_B	m^3	volume of chamber A,B
V_{0A}, V_{0B}	m^3	"dead" volume of chamber A,B
m	kg	piston mass
F_{load}	N	load
x	m	piston position
l	m	maximum piston position

There are different approaches how to model thermal processes in pneumatic system. Isothermal, adiabatic, polytropic models are suitable in different technical applications.

Isothermal model of pressure in cylinder

$$m = \rho V \tag{4.24}$$

$$\dot{m} = \dot{\rho}V + \rho\dot{V} \tag{4.25}$$

Applying 4.1:

$$\rho = \frac{p}{RT} \tag{4.26}$$

$$\dot{\rho} = \frac{\dot{p}}{RT} \tag{4.27}$$

Finally get 4.28:

$$\dot{p} = -\frac{p}{V}\dot{V} + \frac{RT}{V}\dot{m} \tag{4.28}$$

Adiabatic model of pressure in cylinder Assume adiabatic process. For simple adiabatic model following equation can be used 4.29:

$$\dot{p} = -\frac{\kappa p}{V}\dot{V} + \frac{\kappa RT}{V}\dot{m} \tag{4.29}$$

$$\dot{p_A} = \frac{\kappa}{S_A x + V_{0A}} \left(-p_A S_A \dot{x} + RT_A \dot{m_A}\right) \tag{4.30}$$

$$\dot{p_B} = \frac{\kappa}{S_B(l-x) + V_{0B}} \left(p_B S_B \dot{x} + RT_B \dot{m_B}\right) \tag{4.31}$$

Volumes of chambers:

$$V_A = S_A x + V_{0A} \tag{4.32}$$

$$V_B = S_B(l-x) + V_{0B} \tag{4.33}$$

$$\dot{V}_A = S_A \dot{x} \tag{4.34}$$

$$\dot{V}_B = -S_B \dot{x} \tag{4.35}$$

4.4 Mass flow model

4.4.1 Input/Output mass flows

$$\dot{m}T = \dot{m_{in}} T_s - \dot{m_{out}} T_{A/B} \tag{4.36}$$

4.4.2 Valve model

S_{eq}	m^2	Equivalent cross section
S_{max}	m^2	Maximum cross section
Cd	$-$	Coefficient of contraction
u	$-$	Regulation variable

Valve flow model with simply input control signal For regulation flow this model used input control signal directly without spool mechanics.

Coefficient of contraction 4.37:

$$C_d = \frac{S_{eq}}{S_{max}} \tag{4.37}$$

For flow control regulation $u \in \langle -1, 1 \rangle$ can be used.

$$u = \begin{cases} u \in \langle -1, 0) & \text{discharge the chamber} \\ u = 0 & \text{valve closed} \\ u \in (0, 1\rangle & \text{filling the chamber} \end{cases} \tag{4.38}$$

$$\dot{m} = u S_{max} C_d p_1 \sqrt{\frac{2}{RT_1}} \cdot \psi\left(\frac{p_2}{p_1}\right) \tag{4.39}$$

For filling the chamber:

- $p_1 = p_s$
- $p_2 = p_A$ or p_B
- $T_1 = T_s$

For discharge the chamber:

- $p_1 = p_A$ or p_B
- $p_2 = p_0$
- $T_1 = T_A, T_B$

where p_s is supply pressure. p_0 atmospheric pressure. As T_i - atmospheric temperature using according to isothermal process.

$$\dot{m}_A = \begin{cases} u S_v C_d p_s \sqrt{\frac{2}{RT_s}} \cdot \psi\left(\frac{p_A}{p_s}\right) & , u \in (0, 1\rangle \\ 0 & , u = 0 \\ u S_v C_d p_A \sqrt{\frac{2}{RT_A}} \cdot \psi\left(\frac{p_0}{p_A}\right) & , u \in \langle -1, 0) \end{cases} \tag{4.40}$$

$$\dot{m}_B = \begin{cases} u S_v C_d p_s \sqrt{\frac{2}{RT_s}} \cdot \psi\left(\frac{p_B}{p_s}\right) & , u \in (0, 1\rangle \\ 0 & , u = 0 \\ u S_v C_d p_A \sqrt{\frac{2}{RT_B}} \cdot \psi\left(\frac{p_0}{p_B}\right) & , u \in \langle -1, 0) \end{cases} \tag{4.41}$$

Valve flow with spool mechanic included With respect to valve spool modeled as 1DOF system 4.48 and mechanical and geometrical properties following equation were used.

Valve flow with spool In this model we accept a spool displacement x_s, controlled by input voltage u.

$$\dot{m}(P_u,P_d)=\begin{cases} C_f A_v \left(\frac{\kappa}{R}\left(\frac{2}{\kappa-1}\right)\right)^{\frac{1}{2}}\cdot\frac{P_u}{\sqrt{T}}\left(\frac{P_d}{P_u}\right)^{\frac{1}{\kappa}}\cdot\sqrt{1-\left(\frac{P_d}{P_u}\right)^{\frac{\kappa-1}{\kappa}}} & \text{, if } \frac{P_d}{P_u}>P_{cr} \text{ (subsonic)} \\ C_f A_v \frac{P_u}{\sqrt{T}}\cdot\sqrt{\frac{\kappa}{R}\left(\frac{2}{\kappa+1}\right)^{\frac{\kappa+1}{\kappa-1}}} & \text{, if } \frac{P_d}{P_u}\le P_{cr} \text{ (sonic)} \end{cases} \tag{4.42}$$

where C_f is discharge coefficient, A_v is the effective are of valve orifice.

$$A_v=\frac{\pi x_s^2}{4} \tag{4.43}$$

$$x_s=C_v u \tag{4.44}$$

where C_v is the valve constant.

Valve model by Endler Require fitting constants and generally system identification. Mass flow rates are given by following equations:

$$\begin{aligned} \dot{m}_A(u,p_A) &= g_1(p_A,sign(u))arctg(2u) \\ \dot{m}_B(u,p_B) &= g_2(p_B,sign(u))arctg(2u) \end{aligned} \tag{4.45}$$

where g_1, g_2 are signal functions given:

$$\begin{aligned} g_1(p_A,sign(u)) = \beta\Delta p_A &= \begin{cases} (p_s-p_A)\beta^{ench} & \text{, if u} \ge 0 \\ (p_A-p_0)\beta^{esv} & \text{, if u} < 0 \end{cases} \\ g_2(p_B,sign(u)) = \beta\Delta p_B &= \begin{cases} (p_s-p_B)\beta^{ench} & \text{, if u} < 0 \\ (p_B-p_0)\beta^{esv} & \text{, if u} \ge 0 \end{cases} \end{aligned} \tag{4.46}$$

where $\beta^{ench}, \beta^{evs}$ are constant coefficients. For fitting model stop piston (speed of piston is null). This mean that volume is constant. We can measure flow rate $\dot{m}$ versus input voltage u with given pressure difference.

Valve dead-zone For more precision control and modeling of the valve system, valve dead-zone can be used 4.47.

$$u_z=\begin{cases} g_z(u)<0 & \text{, if } u\le u_n \\ 0 & \text{, if } u_n<u<u_p \\ h_z(u)>0 & \text{, if } u\ge u_p \end{cases} \tag{4.47}$$

4.5 Mechanical assembly

4.5.1 Equation of motion

The motion of the pneumatic piston mechanism describes in terms of the general 1dof dynamical equation 4.48.

$$m\ddot{x} + b\dot{x} + kx = u \tag{4.48}$$

In the case of the pneumatic piston, the equation 4.48 transforms into an equation 4.49.

$$(M + M_L)\ddot{x} + F_{damp} + F_g + F_{hs} = F_p \tag{4.49}$$

Where M represents a mass of the all moveable part of the piston, M_L is load mass, F_g gravity force acting to mechanical moving assembly, F_{hs} - models endpoints (hard stop), F_{damp} represents shock absorbers acted at endpoints, F_p is a force produced by the pneumatic piston 4.50.

$$F_p = P_A S_A - P_B S_B - P_0 S_0 \tag{4.50}$$

4.5.2 Hard stop

Hard stop can be represented as spring and dumps:

$$F_{HS} = \begin{cases} K_p(x - g_p) + D_p v & \text{for } x \geq g_p \\ 0 & \text{for } g_n < x < g_p \\ K_n(x - g_n) + D_n v & \text{for } x \leq g_n \end{cases} \tag{4.51}$$

4.5.3 Shock Absorbers

4.5.4 Friction

Friction force can be modeled in the different ways.

TO MUCH 4.52.

$$F_f = \begin{cases} C\dot{x} + \left(f_c + (f_s - f_c)e^{-\left(\frac{\dot{x}}{v_s}\right)^{\delta}} \right) sign(\dot{x}) & , \text{ if } \dot{x} \leq v_e \\ \mu\dot{x} & , \text{ if } \dot{x} > v_e \end{cases} \tag{4.52}$$

where C - viscous friction coefficient, f_c - Coulomb friction, f_s - maximum static friction, μ - dynamic friction factor, v_s - Stribeck velocity, δ - arbitrary index, v_e critical velocity.

4.6 Sensors Modeling

- Sensors models

4.7 Parameter identification

4.7.1 Mechanical assembly

In mechanical system there is F_f force represented by frictions accruing in the system. This force can be modeled by different friction models with respect to **??**. Friction force parameters can be estimated using "gray-box" method. Using $\dot{m}$ mass flow data versus x position measured on real assembly and use these data as an input and output, we can fit F_f. Simplify model can contain TODO:

- F_C static friction
- C_v viscous
- C_p Pressure difference

4.7.2 Cylinder

Dead volume: $p_1 V_1^n = p_2 V_2^n$ or datasheet.

4.7.3 Valve

For valve system there are two parameters that need to be estimated. According to equation 4.53 with constant p_1 (pressure supply) and p_2 (atmospheric pressure), we can estimate $\boldsymbol{C}$ if we neglect Valve Spool dynamic. If in experiment we determine that spool dynamic necessary to include. We provide same experiment with spool model including to "Gray-box" fitting model.

$$\dot{m} = \boldsymbol{u}(x_s)\boldsymbol{C} p_1 \sqrt{\frac{2}{RT_1}} \cdot \psi\left(\frac{p_2}{p_1}\right) \tag{4.53}$$

5 Models comparison (2-3 pages)

5.1 First Principle Model

This model 5.1 was developed with respect to equations represented before.

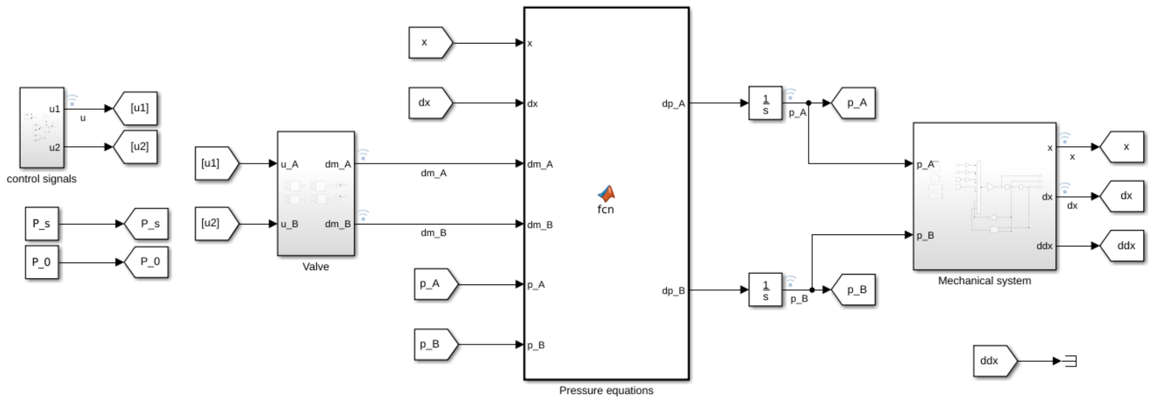

Figure 5.1: Simulink model based on equations

5.2 Alternative Modeling Techniques (3 pages)

Generally with dataset of input-output signals approximation model can be fit. Using System Identification Toolbox and modeled as Black-Box or Gray-Box models. This section attempted to fit some models using data from SimScape and Equation model presented before.

Fit approximation model make sense only if we know what to fit. Using signal process techniques and identify dominant signals that providing best classification features we will train models with respect to this signals.

5.2.1 Physical Model (SimScape)

Working, very slow. Equations are faster for estimation parameters. Model 5.2 was developed using SimScape toolbox.

5.2.2 State-space/ARX Models

Not working, Nonlinearities.

5.2.3 Hammerstein-Wiener Model

Working only for Position.

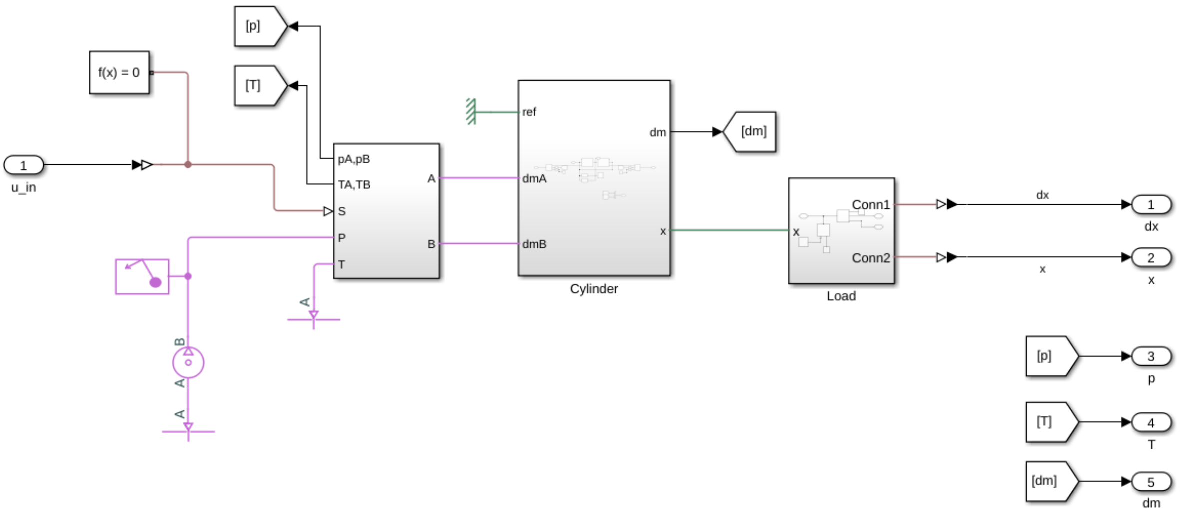

Figure 5.2: Simulink model using SimScape Toolbox

5.2.4 Nonparametric model (ANN)

Working. Can be used as "Normal operation" model.

5.3 Comparison

Following figure 5.3 represent comparison of 2 models (Simscape and based on equations) using same parameters for simulation: There is slight difference between models causing Valve dynamics simplifications in model based on equations.

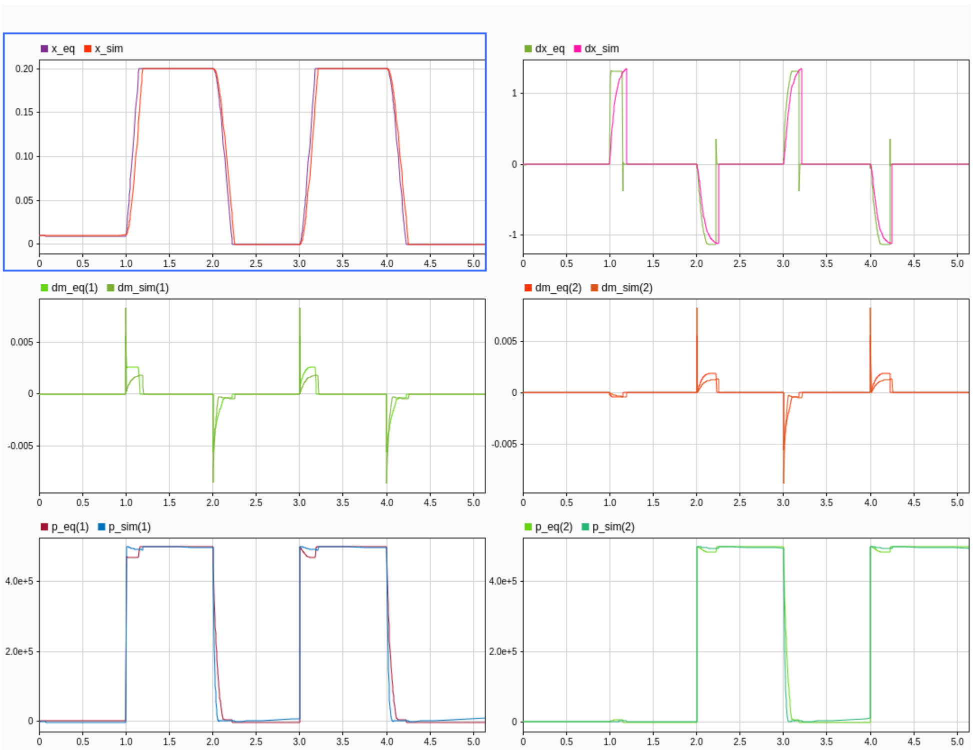

Figure 5.3: Comparison of simscape and model based on equations

6 Signal-Based PdM (15 pages)

Signal-Based Predictive Maintenance.

General

Workflow

6.1 Sensors

Sensors comparison, cost.

6.2 Data exploring

Data has been collect from 8 types of sensors corresponding table 6.1:

Signal name	Description
FlowExtrusin	Flow sensor
FlowContraction	Flow sensor
AirPressure	Pressure sensor
AccelerometerMoving_axisY	Accelerometer
AccelerometerMoving_axisZ	Accelerometer
AccelerometerStat_axisY	Accelerometer
AccelerometerStat_axisZ	Accelerometer
	Temperature sensor
	Proximity sensor
	Strain gauge
	Microphones

Table 6.1: Measured signals

There are 660 measurements with different parameters system parameters 6.2.

Adjusting valve 1	
Adjusting valve 2	

Table 6.2: Device parameters

Dataset was divided to 5 main categories.

Data has been accumulated to ".mat" files. Each file contains signals from sensors during 10 seconds measurements with different pneumatic actuator configuration. Example results from one experiment are represented in figures ??, ??.

6.3 Data management

Data Ensembles Data files have been reshaped to Data Ensembles format used for Condition monitoring purposes. This format allows processing data without copying the whole dataset to memory at once but processes them one by one. In large datasets it gives an option to manipulate with data without problems with allocated memory.

Divided to 3 datasets:

- Train data
- Validation data
- Test data

6.4 Preprocessing

Measured signals require preprocessing concerning the preservation of the information base. For smoothing data Moving Average function were used. As an example, the figure **??** is shown the "raw" and filtered signals. The whole dataset of preprocessed data is relatively big. For time-saving, parallel computing was used for all computationally demanding parts of the code.

6.5 FDI methods

6.5.1 Line checking

We can use Proximity sensor time delay between input signal and upper proximity sensor signal to evaluate if there is some fault.

Same with Position, if not reach some end position, there is a fault.

Flow sensor, check if the float mean value is under some threshold, there is fault.

6.6 Condition Indicators extraction

For classification task purpose from the signals have been extracted statistical features such as mean, median, peak to peak value, etc. As a condition "FaultCode" variable were used. This variable represent configuration of pneumatic actuator during the measurement.

All calculated features were added to the dataset and were ranked by Kruskal-Wallis ANOVA algorithm. Following table **??** contain 5 first best features ranked for classification purpose.

Kruskal-Wallis is very suitable to ranking features before using PCA or SVD.

Selecting Condition Indicators There is a problem if we will deploy classification task with large features dataset. There are different possibilities to reduce data before train classification model or do a prediction. On of them is to rank a features by Analysis of Variation algorithm to evaluate a good representation features.

6.6.1 Microphones

Cheap, good results, but maybe problems with real life integration (noise from another machines). Another problem cannot be modeled in simulation system. For predictive purposes require data from real model.

6.6.2 Encoder

Good results, useful in simulations and compare results with Digital Twin. Can be used in Model-Based CI. Digital twin can generate fault data, that will be applicable with encoder sensor.

6.6.3 Acceleration sensors

Not good, not bad. Can be used for classification task. But encoder has more accuracy information.

6.6.4 Proximity Sensors

Cheap. Very correlated features.Can not be used for classification. But suitable to detect binary classification (Health, Failed). Only statistical features, no Frequency domain.

6.6.5 Flow Sensors

Very expensive sensors. Not so good results.

6.6.6 Air Pressure

This sensor always used, to control pressure valve. But not good results. Maybe in combination with another sensor.

6.6.7 Strain Gauge

Expensive, Normal results of classification. But not suitable for Simulation Model.

6.6.8 Temperature

Good results on data. But only because Ambient temperature was changed between measurements. In one day it was warm, another colder :)

6.7 Classification Task

The main goal of the classification task is to train a model that can predict the "Fault-Code", or "Label" signalized about pneumatic actuator behavior by calculated features.

Using Kuskal-Wallis one way analysis of variance, features were ranked by importance with respect to correlation. This gives opportunity to reduce number of features before PCA analysis.

Principal component analysis (PCA) has been used to reduce the number of features and chose the best representants.

The trained model has been exported to **models/** directory.

7 PdM using a Simulation Model (10-15 pages)

7.1 Differences between Model-Based PdM and PdM using Digital Twin

There is a difference between using Model-Based PdM and using Simulation Model as a Digital Twin.

7.2 Using Digital Twin to Generate Fault Data

We can use Digital Twin to model situations that were not captured in the original dataset or if it is hard to model some cases with real-world hardware. As an example, we can model sensors fault such as sensor drift or complete signal loss.

7.3 Model-Based Condition Indicators

Model-Based approach is suitable when it's difficult to identify condition indicators using only signals. In some cases it's useful to fit some model from data and extract condition indicators as some system parameter.

7.3.1 Static and Dynamic Models

If the system behavior can be fit from the data as a static model, than we can extract condition variables from this model. For example, if model was fitting to a polynomial model, than polynomial coefficients can be use as condition indicators.

Signals showing dynamic behavior can be fitted to dynamic models such as State-Space or AR, ARX, NLARX (Nonlinear auto recursive model) and so on. Then condition indicators can be extracted as poles, zeros damping coefficients from estimated model.

7.3.2 Using Hammerstain-Weiner Model

Demo using Hammerstain-Wiener Model. Fit model to position signal and extract coefficients from model as Condition indicators. Classification.

7.4 Using Simulation Model for Residuals Estimation

Another option is using the Simulink model with **prediction error minimization function** to compute difference between Simulink model and measured data. From this difference we can separate fault condition and healthy operation.

7.4.1 Comparison with Nominal System Model

Same thing as section 7.4

Compare actual system behavior with system model. This will generate some error $e(t) = y(t) - \hat{y}(t)$. From this error residual can be generated in form $r(t) = \Phi(u_t, y_t, \varepsilon_t, v_t, d)$ and after some decision.

7.5 Using Digital Twin to Generate Prognostic Data

Another option is to use Digital Twin to generate a system degradation process. We can evaluate CI from sensor signal by changing a system's mechanical properties as friction or mass flow leakage. Another advantage is that we can design experiments on the model to evaluate what type of data we require from a real-world system to develop a robust algorithm.

7.6 RUL

Demo RUL using generated from model degradation dataset.

8 Conclusion

8.1 Simulation Model

One of the outcomes from the thesis is a simulated model built based on differential equations from the pneumatic-mechanical domain, modeled and developed using Matlab/Simulink software. The simulation model was estimated with parameters of healthy system behavior. However, there is an option to reestimate parameters to fault state and simulate the system in a fault condition.

Due to the available measured data and significantly nonlinear dynamics of the system, the simulation model shows good agreement with the measured data. In contrast to the model built using SimScape library, it is distinctly less computationally expensive while maintaining numerical stability. These facts are fundamental when parameter estimation is in progress.

The simulation model was used to experiment with the system's behavior in different conditions, model fault situations and generate data for the design and development of the robust predictive maintenance algorithms.

8.2 Signal-Based PdM

Another outcome is verifying the possibility of classification and detection of a fault condition applying predictive maintenance techniques, using signal-based and model-based methods.

The experiments were performed on a dataset measured on a demonstration device using seven types of sensors.

A signal-based method is based on the extraction of useful information directly from the signal in time-frequency domains. Each sensor required an individual approach for preprocessing, extracting features, ranking features, and building the classification models. But generally, there is minimal preprocess needed to keep the possible helpful information.

The table 8.1 contains the comparison of sensors in 2 categories, accuracy performed in the test dataset and sensor cost. The graph 8.1 visualizes these data.

Surprisingly, all sensors showed an accuracy of more than 75%. Microphones offer excellent performance from a cost/accuracy perspective, and they are suitable for installation and maintenance.

Sensor	Acc	Encoder	Flow	Mics	Pressure	Proximity	Strain
Accuracy [%]	91.6	96.1	97.2	95.8	76.6	80.5	95.0
Cost [czk]	2x 3500	25000	6000	3x 500	1000	2x 1000	15000

Table 8.1: Comparison of sensors from accuracy/cost perspective

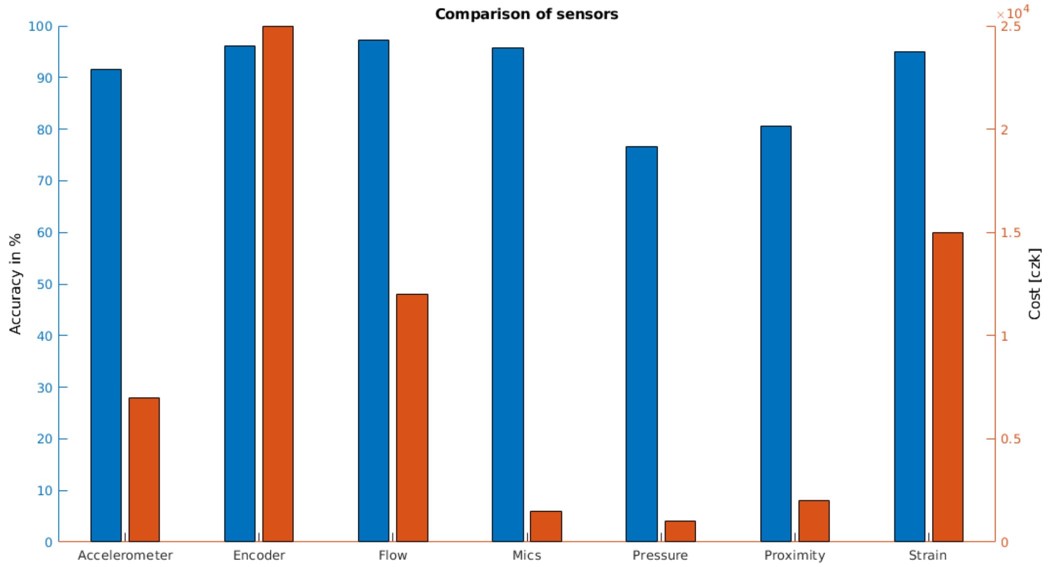

Figure 8.1: Comparison of sensors from accuracy/cost perspective

8.3 Model-Based PdM

The next part of this thesis was to apply model-based methods and using a simulation model for predictive maintenance algorithms. These algorithms are practical when it's hard to extract useful information using a signal-based method. Or it's suitable in some cases where we understand the system dynamics and know how to exploit some system variables as condition indicators.

The use of the method of extraction features in the form of a Nonlinear system identification model coefficient, specifically with the Hammerstein-Wiener model, did not give reliable results. Extracted features have no statistical addiction, and it's impossible to predict fault type using this method on the measured data from the pneumatic piston as a case study.

On the other hand, the residual estimation using the simulation model showed excellent results. The measured position signal was compared with the signal from the simulation model. This residual signal was used to classify the fault condition and achieve 99 % on a smaller dataset. But given the results obtained using the signal-based method, the residual estimation method may seem unnecessary. In this particular case, from a practical point of view, the improvement of the result by a few percent does not bring fundamental changes, but the calculation time increases significantly.

The possibility of modeling and simulation sensor faults was also verified using the simulation model. Although it is difficult to collect data from the sensor fault in real-life conditions, fault data can be generated from the simulation model and even combined with the primary dataset to create a synthetic dataset.

8.3.1 RUL

One of the main goals of predictive maintenance is to estimate the remaining useful life. The original dataset does not contain a record of historical data that shows degradation behavior.

A common problem in the maintenance of pneumatic actuators is the leakage of air from the chamber by the piston. This situation was modeled on the simulation model and generated data used for RUL estimation.

In the demonstration example, a flow signal was measured. From the measurements, the shape factor parameter was calculated and used as a condition indicator. The generated dataset contains 25 simulations with different failure dynamics. Each measurement includes various 10 seconds cycles, depending on the failure dynamic, before the system failure occurs. The outcome is that it is possible to estimate the remaining useful life on generated degradation dataset by using the residual similarity model, pairwise similarity model, and linear degradation model. The prediction results are satisfying.

8.4 Further Development

As a further development, it would be appropriate to estimate the modeled system parameters piecewise to improve the results, emphasizing the characteristics of throttle valves and dampers with adjustments.

Perform air leak fault condition measurements and collect historical degradation data from a real pneumatic piston. Subsequently, evaluate the dynamics of the failure caused by the air leak. Verify the possibility of estimating the remaining useful life using a flow sensor. An interesting case study could be to verify if it is possible to estimate RUL using microphones.

If the performance of the available sensors will be deficient, the pressure measurements in the chamber can be performed. The pressure in the chamber is directly dependent on the air leakage from the chamber, as presented in equation **??**. An example of pressure changes from the simulation model is shown in figure **??**.

List of Abbreviations

LWL Locally Weighted Learning

LS Least Squares Method

RLS Recursive Least Squares Method

RFWR Receptive Field Weighted Regression

LOLIMOT Local Linear Model Tree

EGR Exhaust Gas Recirculation

PID Proportional-Integrational-Derivative controller

Bibliography

[1] NELLES, Oliver. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models.* New York: Springer, c2001. ISBN 35-406-7369-5.

[2] ENGLERT, Peter. *Locally Weighted Learning* [online]. Darmstadt, Germany, 2012, , 9 [cit. 2016-04-28]. Dostupné z: http://www.ausy.informatik.tu-darmstadt.de/uploads/Teaching/AutonomousLearningSystems/Englert_ALS_2012.pdf

[3] BIRATTARI, Mauro a Gianluca BONTEMPI. *The Lazy Learning Toolbox* [online]. 1999, , 30 [cit. 2016-04-28]. DOI: 10.1.1.45.3853. Dostupné z: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.45.3853

[4] GREPL, Robert. Adaptive composite control of electronic throttle using local learning method. In: *2010 IEEE International Symposium on Industrial Electronics* [online]. Brno, the Czech Republic: IEEE, 2010, s. 58-61 [cit. 2016-04-28]. DOI: 10.1109/ISIE.2010.5637899. ISBN 9781424463909. Dostupné z: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5637899

[5] ZHAO, Y. a J.A. FARRELL. A Locally Weighted Learning Method for Online Approximation Based Control. In: *Proceedings of the 44th IEEE Conference on Decision and Control* [online]. Seville, Spain: IEEE, 2005, s. 2694-2701 [cit. 2016-04-28]. DOI: 10.1109/CDC.2005.1582570. ISBN 0780395670. Dostupné z: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1582570

[6] SU, J., J. WANG a Y. XI. Incremental Learning With Balanced Update on Receptive Fields for Multi-Sensor Data Fusion. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)* [online]. 2004, **34**(1), 659-665 [cit. 2016-04-28]. DOI: 10.1109/TSMCB.2002.806485. ISSN 10834419. Dostupné z: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1262536

[7] NAKANISHI, Jun, Jay A. FARRELL a Stefan SCHAAL. Composite adaptive control with locally weighted statistical learning. *Neural Networks* [online]. 2005, **18**(1), 71-90 [cit. 2016-04-28]. DOI: 10.1016/j.neunet.2004.08.009. ISSN 08936080. Dostupné z: http://linkinghub.elsevier.com/retrieve/pii/S0893608004001728

[8] SCHAAL, Stefan, Christopher G. ATKESON a Sethu VIJAYAKUMAR. Scalable techniques from nonparametric statistics for real time robot learning. *Applied Intelligence* [online]. 2002, **17**(1), 49-60 [cit. 2016-04-28]. DOI: 10.1023/A:1015727715131. ISSN 0924669x. Dostupné z: http://link.springer.com/10.1023/A:1015727715131

[9] VIJAYAKUMAR, Sethu, Aaron D'SOUZA a Stefan SCHAAL. Incremental Online Learning in High Dimensions. *Neural Computation* [online]. 2005, **17**(12), 2602-2634 [cit. 2016-04-28]. DOI: 10.1162/089976605774320557. ISSN 08997667. Dostupné z: http://www.mitpressjournals.org/doi/abs/10.1162/089976605774320557

[10] Local dimensionality reduction. SCHAAL, S., S. VIJAYAKUMAR a C.G. ATKESON. *Advances in neural information processing systems 10: proceedings of the 1997 conference ; [presented at the Eleventh Annual Conference on Neural Information Processing (NIPS), held in Denver, Colorado from December 1 to December 6, 1997]* [online]. Electronic version. Cambridge, Mass. [u.a.]: MIT Press, 1998, s. 633-639 [cit. 2016-04-28]. ISBN 0262100762. Dostupné z: http://papers.nips.cc/paper/1387-local-dimensionality-reduction.pdf

[11] ATKESON, Christopher G., Andrew W. MOORE a Stefan SCHAAL. Locally weighted learning for control. *Artificial Intelligence Review* [online]. 1997, **11**(1/5), 75-113 [cit. 2016-04-28]. DOI: 10.1023/A:1006511328852. ISSN 02692821. Dostupné z: http://link.springer.com/10.1023/A:1006511328852

[12] SCHAAL, Stefan a Christopher G. ATKESON. Constructive Incremental Learning from Only Local Information. *Neural Computation* [online]. 1998, **10**(8), 2047-2084 [cit. 2016-04-28]. DOI: 10.1162/089976698300016963. ISSN 08997667. Dostupné z:https://pdfs.semanticscholar.org/0d1d/0b6e29e3b7d6b357537b9e1908b852a37a0e.pdf

[13] ATKESON, Christopher G., Andrew W. MOORE a Stefan SCHAAL. Locally weighted learning. *Artificial Intelligence Review* [online]. 1996, **11**(1/5), 11-73 [cit. 2016-04-28]. DOI: 10.1023/A:1006559212014. ISSN 02692821. Dostupné z: http://link.springer.com/10.1023/A:1006559212014

[14] NAKANISHI, Jun, Jay A. FARRELL a Stefan SCHAAL. Composite adaptive control with locally weighted statistical learning. *Neural Networks* [online]. 2005, **18**(1), 71-90 [cit. 2016-05-11]. DOI: 10.1016/j.neunet.2004.08.009. ISSN 08936080. Dostupné z: http://linkinghub.elsevier.com/retrieve/pii/S0893608004001728