



Conference proceedings
Sborník odborného semináře

**Nové metody a postupy v oblasti přístrojové
techniky, automatického řízení a informatiky 2021**
**New Methods and Practices in the Instrumentation,
Automatic Control and Informatics 2021**

15. 9. – 17. 9. 2021, Žatec

ISBN 978-80-01-06889-2

OBSAH

<i>Izabela Wawrzonkiewicz, Daniel W. Andrukonis, Josef Kokeš</i> BIOMECHANICS OF HEALTHY HUMAN'S GAIT	3
<i>Sami Jradi</i> INDUSTRIAL IOT WEB APPLICATION FOR MOTOR CHARACTERISTIC MONITORING	27
<i>Ondřej Bimka, Josef Kokeš</i> TIME SERIES FORECASTING WITH RECURRENT NEURAL NETWORKS	38
<i>Youssef Alsoufi</i> ADVANCED SUSPENSION SYSTEM WITH TWO-DEGREE-OF-FREEDOM HYBRID MASS DAMPERS	43
<i>Ahmad Atiyah</i> ANALYSIS OF LOCOMOTIVE MOTOR TORQUE IMPACT AT CHANGEABLE DYNAMIC OPERATION CONDITIONS	52
<i>Adrian Saldanha, Milan Hofreiter</i> IDENTIFICATION AND CONTROL USING GEA	60
<i>Jakub Znamenáček</i> APLIKACE PRO INDUSTRIAL EDGE	70
<i>Jan Janovský</i> CONTROLLING MOTION APPS OF MOTION TERMINAL FESTO VTEM BY NODE-RED	74
<i>Vladimír Hlaváč</i> ŘEŠENÍ INVERZNÍ KINEMATIKY PLANÁRNÍHO MANIPULÁTORU MLP NEURONOVOU SÍTÍ	78
<i>Ladislav Červinka</i> FESTO MOTION TERMINAL VTEM CONTROLLING UNIVERSAL PRESSING FACILITY	85
<i>Pavel Blažek</i> MOŽNOSTI REALIZACE INTERFERENČNÍHO SPEKTROMETRU	91
<i>Petr Siblík</i> POUŽITÍ NAVÁDĚNÉHO EVOLUČNÍHO ALGORITMU K OPTIMALIZACI PARAMETRŮ PSD REGULÁTORŮ	102
<i>Jakub Jura, Pavel Trnka, Martin Cahyna, Martin Novák</i> DIGITÁLNÍ DVOJČE PRO DISTANČNÍ VÝUKU PROGRAMOVATELNÝCH AUTOMATŮ	110
<i>Jan Mohelník1, Pavel Trnka</i> MODULAR STATION CP LAB	115
<i>Ivana Oswaldová, Jaromír Fišer, Tomáš Vyhlídal</i> PROCRAFT A SPOLUPRÁCE S LETECKÝM MUZEEM KBELY	124
<i>Kryštof Bystřický</i> DATA-DRIVEN MODEL DISCOVERY USING SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS	130
<i>Michal Pála</i> NÁVRH NELINEÁRNÍHO MODELU 3D PORTÁLOVÉHO JEŘÁBU	145
<i>Tomáš Pelcl</i> MATEMATICKÝ MODEL VĚŽOVÉHO JEŘÁBU	149

BIOMECHANICS OF HEALTHY HUMAN'S GAIT

I. Wawrzonkiewicz¹, D.W. Andrukonis², J.Kokeš³

¹ *Faculty of Mechanical Engineering Bialystok University of Technology, wawiza77@wp.pl/wawriza@cvut.cz*

² *Faculty of Mechanical Engineering Bialystok University of Technology, 117.daniel@gmail.com/andrudan@cvut.cz*

³ *Faculty of Mechanical Engineering CTU in Prague, Josef.Kokes@fs.cvut.cz*

Abstract: Most people are extremely skilled in many everyday movements like standing, walking, or climbing stairs. Unfortunately, modern living does not require enough movement to prevent several chronic diseases associated with low physical activity. Fortunately, many human movement professions like physical educators, coaches, athletic trainers, strength & conditioning coaches, personal trainers, and physical therapists help people reap the benefits of physical activity.

Kinesiology is the term referring to the whole scholarly area of human movement study, while biomechanics is the study of motion and its causes in living things. Biomechanics provides key information on the most effective and safest movement patterns, equipment, and relevant exercises to improve human movement. In a sense, kinesiology professionals solve human movement problems every day, and one of their most important tools is biomechanics. This work outlines the field of biomechanics, why biomechanics is such an important area and where biomechanics information can be found. The aim of this study was to present some possible quantitative and qualitative measurements of human's gait using simple Software Kinovea and Scilab and to determine kind of gait pattern for examined persons.

This paper describes gait as a one of human's motion pattern. It also details two "regions" of biomechanics analysis which are qualitative and quantitative analysis and their influence on human's everyday life.

Keywords: Biomechanical study, Quantitative, Qualitative, Human movement, Gait

1 Introduction

Biomechanics has been defined as the study of the movement of living things using the science of mechanics (H.Hatze, 1974). Mechanics is a branch of physics that is concerned with the description of motion and how forces create motion. Forces acting on living things can create motion, be a healthy stimulus for growth and development, or overload tissues, causing injury. Biomechanics provides conceptual and mathematical tools that are necessary for understanding how living things move and how kinesiology professionals might improve movement or make movement safer. Kinesiology is the academic area for the study of human movement (Corbin & Eckert, 1990). [7]

Biomechanics provides information for a variety of kinesiology professions to analyze human movement to improve effectiveness or decrease the risk of injury. How the movement is analyzed falls on a continuum between a qualitative analysis and a quantitative analysis. [7]

Quantitative analysis involves the measurement of biomechanical variables and usually requires a computer to do the voluminous numerical calculations performed. Even short movements will have thousands of samples of data to be collected, scaled, and numerically processed. In contrast, qualitative analysis has been defined as the "systematic observation and introspective judgment of the quality of human movement for the purpose of providing the most appropriate intervention to improve performance" (Knudson & Morrison, 2002, p. 4). [7]

Analysis in both quantitative and qualitative contexts mean identification of the factors that affect human movement performance, which is then interpreted using other higher levels of thinking (synthesis, evaluation) in applying the information to the movement of interest. Solving problems in human movement involves high levels of

critical thinking and an interdisciplinary approach, integrating the many kinesiology sciences. [7]

2 Theme development

2.1 Gait as a pattern of motion

The gait cycle is defined as the time interval between two successive occurrences of one of the repetitive events of walking. Although any event could be chosen to define the gait cycle, it is generally convenient to use the instant at which one foot contacts the ground ('initial contact'). If it is decided to start with initial contact of the right foot, then the cycle will continue until the right foot contacts the ground again. [4]

Two phases of gait can be distinguished: stance phase, which is also called the 'support phase' or 'contact phase' when the foot is on the ground; lasts from initial contact to toe off. It is subdivided into: loading response, mid-stance, terminal stance and pre-swing. The other one is swing phase, when the foot is moving forward through the air. The swing phase lasts from toe off to the next initial contact. It is subdivided into: initial swing, mid-swing and terminal swing. The representation of each phase of gait is presented below (Fig. 1). [4]

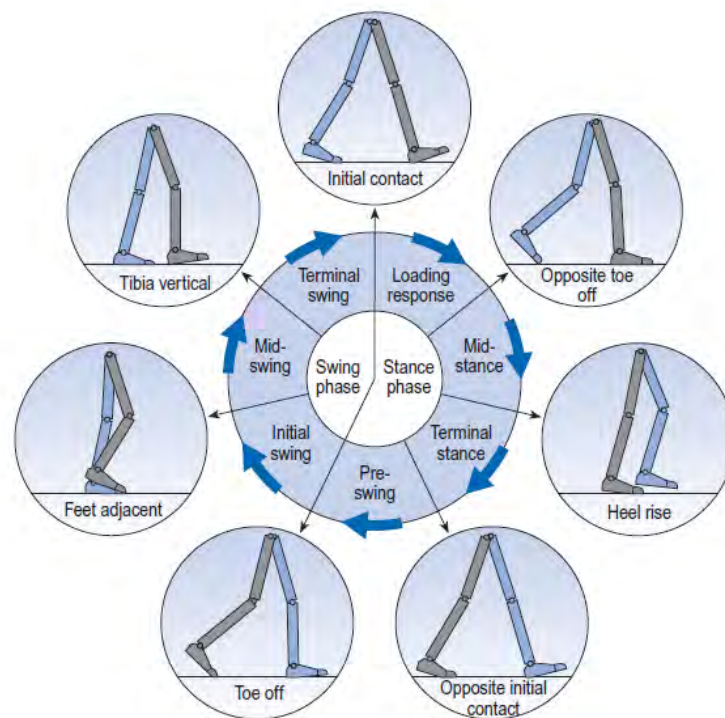


Fig. 1.: Positions of the legs during a single gait cycle by the right leg (gray) [4]

Double support phase is when one foot is forward, having just landed on the ground, and the other one is backward, being just about to leave the ground. In each gait cycle, there are thus two periods of double support and two periods of single support. The stance phase usually lasts about 60% of the cycle, the swing phase about 40% and each period of double support about 10%. The swing phase becomes proportionately longer and the stance phase as well as double support phases becomes shorter, as the speed increases. The final disappearance of the double support phase marks the transition from walking to running. Between successive steps in running there is a flight phase, also known as the 'float', 'double-float' or 'non-support' phase, when neither foot is on the ground. [4]

During gait, important movements occur in all three planes – sagittal, frontal and transverse. However, the largest movements occur in the sagittal plane. That is the reason that all this research has been carried out in a sagittal plane. All of the planes are presented below in Fig. 2. [4]

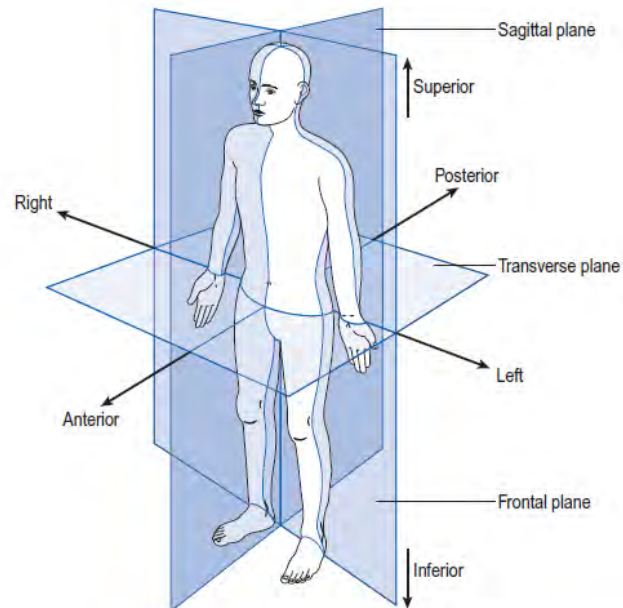


Fig. 2.: The anatomical position, with three reference planes and six fundamental directions [4]

2.2 Gait cycle

Each of the following sections consider some general remarks about the events surrounding a particular event in the gait cycle.

Initial contact - the beginning of the loading response, which is the first period of the stance phase. Initial contact is frequently called 'heelstrike', since in normal individuals there is often a distinct impact between the heel and the ground, known as the 'heelstrike transient'. Other names for this event are 'heel contact', 'footstrike' or 'foot contact'. The direction of the ground reaction force changes from generally upwards turning the heelstrike transient to upwards and backward in the loading response, immediately afterwards.

Loading response - the double support period between initial contact and opposite toe off. During this period, the foot is lowered to the ground by plantarflexion of the ankle. The ground reaction force increases rapidly in magnitude, its direction being upwards and backwards. Loading response typically occupying the first 10–12% of the cycle.

Opposite toe off - the end of the double support period known as loading response and the beginning of midstance, the first period of single support. The forefoot, which was being lowered by plantarflexion of the ankle, contacts the ground at 'foot flat', also known as 'forefoot contact', which generally occurs around the time of opposite toe off. On the opposite (left) side, it marks the end of the stance phase and the beginning of the swing phase.

Mid-stance - the period of the gait cycle between opposite toe off and heel rise, although the term has been used in the past to describe an event of the gait cycle- the time at which the swing phase leg passes the stance phase leg, corresponding to the swing phase event of 'feet adjacent'.

Heel rise - marks the transition from midstance to terminal stance. It is the time, at which the heel begins to lift from the walking surface. Its timing varies considerably, both from one individual to another and with the speed of walking.

Opposite initial contact -as might be expected, opposite initial contact in symmetrical gait occurs at close to 50% of the cycle. It marks the end of the period of single support and the beginning of pre-swing, which is the second period of double support. At the time of opposite initial contact, also known as 'opposite foot contact', the hip begins to flex, the knee is already flexing and the ankle is plantarflexing. The period between heel rise and toe off (terminal stance followed by pre-swing) is sometimes called the 'terminal rocker'. This is appropriate, since the leg is now rotating forwards about the forefoot, rather than about the ankle joint.

Toe off generally occurs at about 60% percent of the gait cycle. It separates pre-swing from initial swing and is

the point at which the stance phase ends and the swing phase begins. The name 'terminal contact' has been proposed for this event, since in pathological gait the toe may not be the last part of the foot to leave the ground.

Feet adjacent separates initial swing from mid-swing. It is the time when the swinging leg passes the stance phase leg and the two feet are side by side. The swing phase occupies about 40% of the gait cycle and the feet become adjacent around the center of this time. Alternative names for feet adjacent are 'foot clearance' and 'mid-swing'; the latter term is now applied to a period of the gait cycle, rather than to a particular event. Initial swing is also known as 'lift off'.

Tibia vertical - the division between the periods of mid-swing and terminal swing is marked by the tibia of the swinging leg becoming vertical. Terminal swing is also known as 'reach'.

Terminal foot contact - the gait cycle ends at the next initial contact of the same foot (in this case, the right foot). Because it is confusing to refer to the end of the cycle as 'initial contact', it is sometimes known as 'terminal foot contact'. [4]

2.3 Most important gait parameters

The cyclic nature of human gait is a very useful feature for reporting different parameters. There are literally hundreds of parameters that can be expressed in terms of the percent cycle. For the aim of this article it was chosen just a few examples to illustrate this point. There can be distinguished distance (spatial) and time (temporal) variables. Distance variables are: step length, stride length, width of walking base and foot angle (degree of toe out or angle of gait). Time variables are: step time, stride time, stance time, single limb time, double limb time, swing time, cadence and speed. [9]

Step length- distance between corresponding successive points of heel contact of the opposite feet. For right leg step length is equal with left leg (in a normal gait).

Stride Length- distance between successive points of heel contact of the same foot. Its is double the step length (in normal gait).

Walking Base- side-to-side distance between the line of the two feet. It is also known as 'stride width'.

Degree of toe out- it represents the angle of foot placement and may be found by measuring the angle formed by each foot's line of progression and a line intersecting the center of the heel and the second toe. The angle for men is about 7°. The degree of toe out decreases as the speed of walking increases in normal men.

Step time- it is referred to the amount of time spent during single step. It is the time between heel strike of one leg and heel strike of the contra-lateral leg.

Stride time- it is referred to the amount of time it takes to complete one stride. Stride duration and gait cycle duration are the same.

Stance time- it is the amount of time that passes during the stance phase of one extremity in a gait cycle. It includes single support and double support.

Swing time- it is the amount of time that passes during the swing phase of one extremity in a gait cycle. If the stride time of the gait cycle is one second, the stance time is 0.6 second and swing time is 0.4 second.

Single limb time- it is the amount of time that passes during the period when only one extremity is on the supporting surface in a gait cycle.

Double limb time- it is the amount of time that a person spends with both feet on the ground during one gait cycle. The percentage of time spent in double support decreases as the speed of walking increases.

Cadence- number of steps per unit time. Normally it is 100 – 115 steps/min .

Speed (Velocity)- distance covered by the body in unit time. Usually measured in m/s. Instantaneous velocity varies during the gait cycle. Average velocity (m/min) = step length (m) x cadence (steps/min). Average walking speed = 80m/minute. [9]

2.4 Purposes of the study human's gait

From the clinical point of view, the importance of human gait analysis lies in the fact that gait disorders affect a high percentage of the world's population and are key problems in neurodegenerative diseases such as multiple sclerosis, amyotrophic lateral sclerosis or Parkinson's disease, as well as in many others such as myelopathies, spinal amyotrophy, cerebellar ataxia, brain tumors, cranioccephalic trauma, neuromuscular diseases (myopathies), cerebrovascular pathologies, certain types of dementia, heart disease or physiological ageing. Study of human gait characteristics may be useful for clinical applications and may benefit the various groups suffering from gait-related disorders. In the elderly, physical exercise has a major impact on osteoporosis, because it significantly helps to

prevent falls, which are the biggest risk factor for this age group. Therefore, evaluation of gait quality may be valuable for early diagnosis of disease such as osteoporosis. Many neurodegenerative and age-related diseases such as Parkinson's are linked to other parameters which make it possible to diagnose and know the patient's evolution. New methods have great impact in various fields such as human recognition, sports, and especially in the clinical field, where objective gait analysis plays an important role in diagnosis, prevention and monitoring of neurological, cardiopathic and age-related disorders. [1]

2.5 Qualitative analysis

The purpose of a qualitative anatomical analysis is to determine the predominant muscular activity during specific phases of a performance and to identify instants when large stresses may occur due to large muscle forces or extremes in joint ranges of motion. [6] Qualitative analysis of gait (walking) also helps the therapist decide whether sufficient muscular strength and control have been regained in order to permit safe or cosmetically normal walking. [7] The results included measurements of gait in slow motion for 5 health persons and motion analysis in the ankle, knee, hip and tibia. Analysis was carried out only in sagittal plane, because of using only one camera. The range of movement in these body parts are extension, flexion and neutral position.

2.6 Quantitative analysis

A quantitative gait analysis is generally considered to be any objective means that can be used to measure walking performance. The procedure can be as simple as measuring step length with a ruler or determining cadence with a stopwatch, or it can be as sophisticated as full-body motion capture with state-of-the-art instrumentation. Regardless of the methods, the measurements that are collected, are used to assess the quality of the gait and to characterize the motion. Gait analyses are typically performed before and after the intervention to determine efficacy of treatment. Treatment is considered beneficial if improvement in the gait pattern is observed, evident by a reduction in abnormal movements with an evolution toward patterns that are more like those of able-bodied individuals. [2] In this work there were made three parts connected with quantitative analysis of gait. First part was calibration of video of human's gait. This calibration was made in two ways: in Kinovea and Scilab Software to compare results from these two programs. Second part of this research was analyzing human's gait by its parameters like velocity and acceleration. The aim was to put values from Kinovea to Excel and based on it, to make some plots, which showed how velocity and acceleration are changing in the time of walking. Adding trendline made plots more clear. Third part of quantitative analysis was to find centers of mass of human body. It was conducted using a code in Scilab and allowed to determine a center of mass each segment of body and then of all the body. The results included plots which presents how the center of body is changing in each phase of walking.

3 Systematic observation

Systematic observation means, that it is not possible get credible results, if many tries won't be conducted. It is important to make at least 3 trials to get credible results. Another thing which is essential in carrying out experiment is to plan the whole process of research. It is significant to determine place, right equipment (camera, tripod, bars, markers) and people who we will be 'investigating'. The research was conducted on 5 people (3 woman and 2 man), with no significant gait disorders. Another essential thing while carrying out the experiment is making sure that investigated people have proper clothes. It is really important to have shorts and t-shirt what enables to observe knee, shoulder, elbow, wrist and another important parts of the body. Investigated persons should also have proper shoes, for good ankle visibility.

4 Result

4.1 Qualitative analysis of human's gait

Results contain motion analysis of 5 investigated persons in the ankle, tibia, knee and hip for left and right leg carried out in sagittal plane (Fig. 7, 8). The range of movement in these body parts are extension, flexion and neutral position. There is also presented personal data of each examined persons (Fig. 3). The analysis was conducted on the basis on slow motion videos of natural walking of 5 examined persons, which were recorded during working on this article. Range of movement for different parts of body were presented in Fig. 4, 5, 6.

number	sex(M/F)	height(cm)	weight(kg)
1	F	181	65
2	M	189	67
3	F	170	80
4	F	178	65
5	M	175	75

Fig. 3.: Personal data for examined persons

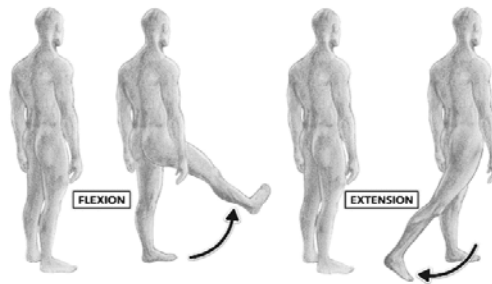


Fig. 4.: Flexion and extension at the hip [8]

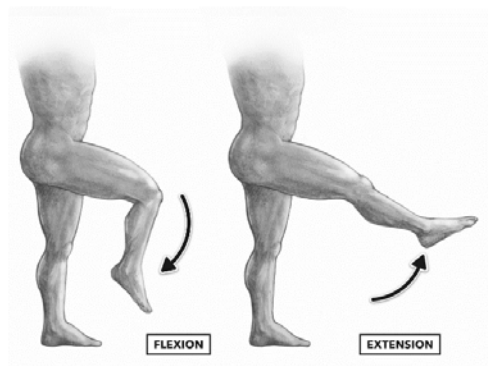


Fig. 5.: Flexion and extension at the knee [8]

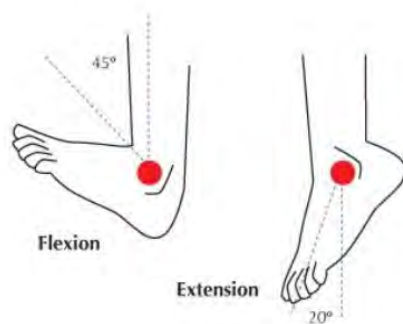


Fig. 6.: Flexion and extension at the ankle [8]

Left				Person 1	Person 2	Person 3	Person 4	Person 5	
Sagittal	Stance Phase	Loading response	Initial Contact	Ankle	extension	extension	extension	extension	extension
				Tibia	extension	extension	extension	extension	extension
				Knee	extension	flexion	extension	flexion	flexion
				Hip	flexion	flexion	flexion	flexion	flexion
		Mid-stance	opposite toe off	Ankle	extension	neutral	neutral	neutral	neutral
				Tibia	extension	neutral	neutral	extension	neutral
				Knee	extension	flexion	neutral	flexion	flexion
				Hip	flexion	flexion	neutral	flexion	flexion
		Terminal - stance	heel rise	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	extension	extension	flexion	extension	flexion
				Hip	extension	extension	extension	extension	extension
		Pre-swing	opposite initial contact	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	extension	flexion	flexion	extension	flexion
				Hip	extension	extension	extension	extension	extension
	Swing Phase	Initial swing	Toe off	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	flexion	flexion	flexion	flexion	flexion
				Hip	neutral	extension	extension	extension	extension
		Mid swing	feet adjacent	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	flexion	flexion	flexion	flexion	flexion
				Hip	flexion	flexion	flexion	flexion	flexion
		Terminal- swing	tibia vertical	Ankle	flexion	neutral	extension	neutral	extension
				Tibia	neutral	neutral	neutral	neutral	neutral
				Knee	flexion	flexion	flexion	flexion	flexion
				Hip	flexion	flexion	flexion	flexion	flexion

Fig. 7.: Results for 5 examined persons (left leg) in sagittal plane- quantitative analysis of gait

Right									
Sagittal	Stance Phase	Loading response	Initial Contact	Ankle	extension	extension	extension	extension	extension
				Tibia	extension	extension	extension	extension	extension
				Knee	extension	flexion	extension	flexion	flexion
				Hip	flexion	flexion	flexion	flexion	flexion
		Mid-stance	opposite toe off	Ankle	extension	neutral	neutral	neutral	neutral
				Tibia	extension	neutral	neutral	extension	neutral
				Knee	extension	flexion	neutral	flexion	flexion
				Hip	flexion	flexion	neutral	flexion	flexion
		Terminal - stance	heel rise	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	extension	extension	flexion	extension	flexion
				Hip	extension	extension	extension	extension	extension
		Pre-swing	opposite initial contact	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	extension	flexion	flexion	extension	flexion
				Hip	extension	extension	extension	extension	extension
	Swing Phase	Initial swing	Toe off	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	flexion	flexion	flexion	flexion	flexion
				Hip	neutral	extension	extension	extension	extension
		Mid swing	feet adjacent	Ankle	flexion	flexion	flexion	flexion	flexion
				Tibia	flexion	flexion	flexion	flexion	flexion
				Knee	flexion	flexion	flexion	flexion	flexion
				Hip	flexion	flexion	flexion	flexion	flexion
		Terminal- swing	tibia vertical	Ankle	flexion	neutral	extension	neutral	extension
				Tibia	neutral	neutral	neutral	neutral	neutral
				Knee	flexion	flexion	flexion	flexion	flexion
				Hip	flexion	flexion	flexion	flexion	flexion

Fig. 8.: Results for 5 examined persons (right leg) in sagittal plane- quantitative analysis of gait

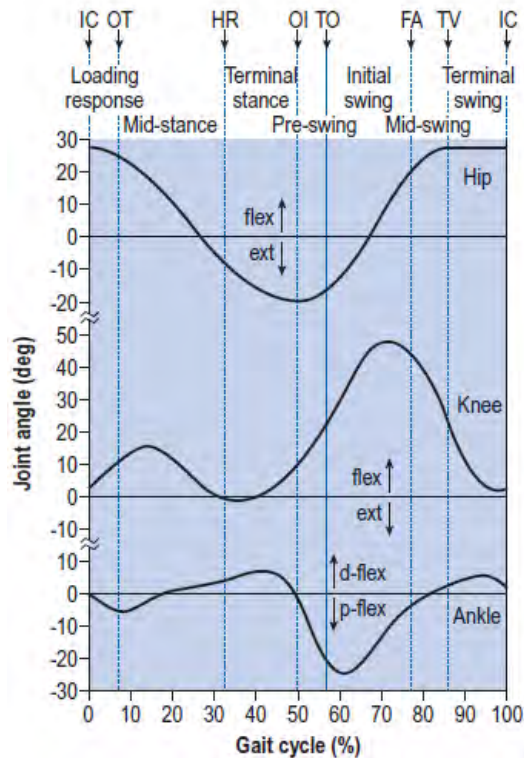


Fig. 9.: Sagittal plane joint angles (degrees) during a single gait cycle of right hip (flexion positive), knee (flexion positive) and ankle (dorsiflexion positive) [4]

Most of the results for each phase are overlapping with those showed in the reference (Fig. 9). However, pattern of gait for each human is different, considering weight of body, individuals character of gait and some unknown disorders of gait, as well as for example spine diseases. This part was conducted to show how complicated are moves of human body and its individual parts during conducting one full cycle of gait.

4.2 Quantitative analysis- Calibration 2DDL

The direct linear transformation (DLT) method has been one of most widely used camera digitalization and reconstruction algorithms. The relationship between the object space coordinates and the image plane coordinates are described by a set of cameras and markers placed in certain places. Strength of this method is much more accurate than Kinovea Software mathematical method, and it is not as complicated as 3DDL one. Only one plane is needed (sagittal one in this case). Camera which was used for recordings was HTC U11 mobile phone camera. In slow motion it was able to record video in 120 fps in 1080p resolution. The experiment consisted of 3 parts.

1. Calibration is the setting or correcting of a measuring device or base level, usually by adjusting it to match or conform to a dependably known and unvarying measure. [10] To be confident in the results being measured there is an ongoing need to maintain the calibration of equipment throughout its lifetime for reliable, accurate and repeatable measurements. In this work it was necessary to make a calibration, because the experiment was carried out by using a camera. Calibration was done both in Kinovea Software and by using Scilab Software to compare both methods. It was necessary to create at least 4 points, which then were reference. Experiment was conducted by using 12 points, which gave more accurate measurement. These 12 points were used as reference ones. The goal was to determine the calibration coefficients and digital coordinates using Kinovea and Scilab Software. On the picture below (Fig. 10) it is presented how the experiment was runned. It consisted of 3 attempts.

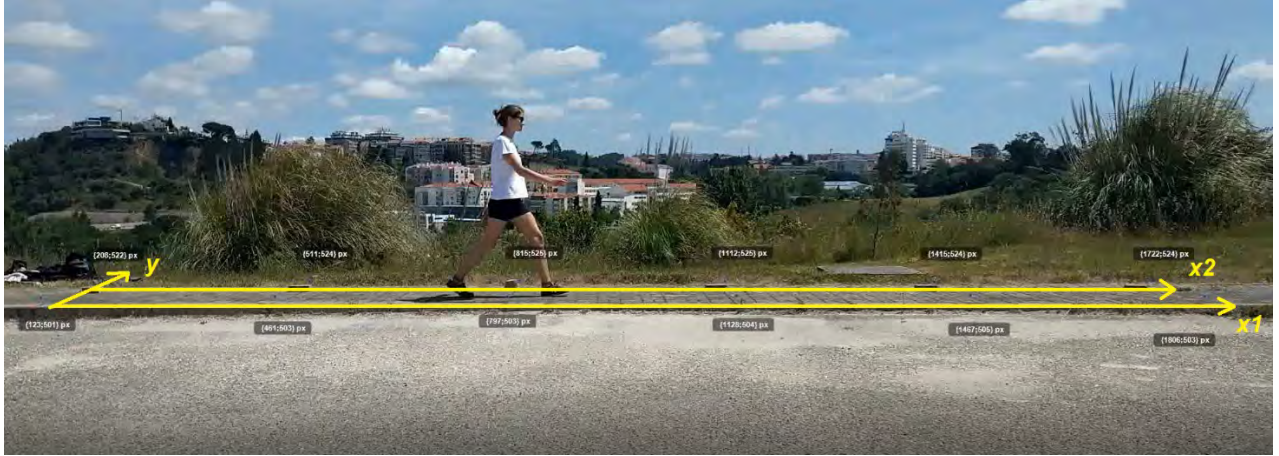


Fig. 10.: Digital coordinates pointed in Kinovea on our video of walking

Real coordinates		Digital coordinates	
x1	y	x	y
0	0	123	501
2	0	461	503
4	0	797	503
6	0	1128	504
8	0	1467	505
10	0	1806	503
x2	y	x	y
0	1	208	522
2	1	511	524
4	1	815	525
6	1	1112	525
8	1	1415	524
10	1	1722	524

Fig. 11.: Real and digital coordinates of 12 reference points from Kinovea putted in Excel (first attempt)

2.Measuring the bars- Second part of this work was to measure lengths of bars. In this case there were used stones, on which were putted some well-visible tape. Configuration of bars was 2m- first bar, 1m- second bar, 1m- third bar. They were placed parallel to camera to avoid errors caused by angle between camera and object.

It was necessary to make calibrations of bars in Kinovea Software what we can see below in Fig. 12. Also it was important to put points of localization of bars (7 points) which then were used in Scilab Software (Fig. 13).



Fig. 12.: Calibration of bars in Kinovea (third attempt)

Bar coordinates	
x	y
386	505
543	505
707	506
764	504
924	505
1009	510
1165	509

Fig. 13.: Localization of bars (7 points) from Kinovea (first attempt)

3. Authors of this article prepared 2 codes in Scilab Software presented in Fig. 14 and 15. First step in code 1 (Fig.14) was to open real and digital coordinates of reference points known from Kinovea. We obtained value of coefficient "A". Then by using code 2 (Fig. 15) and knowing value "A" it was necessary to open file with digital coordinates of bars. We obtained coefficient "H". Based on those codes, results of bar lengths were obtained (Fig. 16, 17, 18).

```
ZDDLT_calcula_coef_new.sce  ZDDLT_calcula_dlt_new.sce
1  clc;clear;
2  //F -- REAL COORDINATES
3  //L -- CAMERA COORDINATES
4
5  //SELECT THE EXCEL FILE WITH REAL COORDINATES
6  path=uigetfile(['*.xls']);
7  //OPEN XLS FILES WITH REAL COORDINATES
8  [total_sheet,txt_vol,nome_sheet,pos_sheet]=xls_open(path);
9  //OPEN SHEET WITH REAL COORDINATES
10 [num_coo,txt_coo] = xls_read(total_sheet,pos_sheet(1));
11 F=num_coo;
12 Cut=[];
13
14 //SELECT THE EXCEL FILE WITH DIGITAL COORDINATES
15 path1=uigetfile(['*.xls']);
16 //OPEN XLS FILES WITH DIGITAL COORDINATES
17 [total_sheet1,txt_vol1,nome_sheet1,pos_sheet1]=xls_open(path1);
18 //OPEN SHEET WITH DIGITAL COORDINATES
19 [num_cool,txt_cool] = xls_read(total_sheet1,pos_sheet1(1));
20 L=num_cool;
21
22 if size(F) ~= size(L)
23 disp('# of calibration points entered and seen in camera do not agree')
24 end
25
26 m=size(F,1); Lt=L'; C=Lt(:);
27
28 for i=1:m
29     B(2*i-1,1) = F(i,1);
30     B(2*i-1,2) = F(i,2);
31     B(2*i-1,3) = 1;
32     B(2*i-1,7) = -F(i,1)*L(i,1);
33     B(2*i-1,8) = -F(i,2)*L(i,1);
34     B(2*i,4) = F(i,1);
35     B(2*i,5) = F(i,2);
36     B(2*i,6) = 1;
37     B(2*i,7) = -F(i,1)*L(i,2);
38     B(2*i,8) = -F(i,2)*L(i,2);
39 end
40
41 //Cut the lines out of B and C including the control points to be discarded
42 //Cutlines=[Cut.*2-1, Cut.*2];
43 //B([Cutlines],:)=[];
44 //C([Cutlines],:)=[];
45
46 //Solution for the coefficients
47 A=B\C;
48 D=B*A;
49 R=C-D;
50 res=norm(R); avgres=res/size(R,1)^0.5;
```

Fig. 14.: Code 1 in Scilab Software

```
2DDLT_calcula_coef_new.sce 2DDLT_calcula_dit_new.sce
1 //A - Coefficients
2 //L Coordinates
3
4 //READ DIGITAL COORDINATES
5 //SELECT THE EXCEL FILE WITH DIGITAL COORDINATES - (in this case of the bar)
6 path2=uigetfile(['*.xls*']);
7 //OPEN EXCEL FILE
8 [total_sheet2,txt_vol2,nome_sheet2,pos_sheet2]=xls_open(path2);
9 //OPEN SHEET WITH DIGITAL COORDINATES
10 [num_coo2,txt_coo2] = xls_read(total_sheet2,pos_sheet2(1));
11 L=num_coo2;
12
13 n=size(A,2);
14 // check whether the numbers of cameras agree for A and L
15 if size(A,2)~=1 | size(L,2)~=2; disp('there is more then one camera given in A or L')
16     disp('hit any key and then try again'); pause;
17 end
18
19
20 H(size(L,1),2)=[0]; // initialize H
21
22 // _____ Building L1, L2: L1 * G (X,Y) = L2 _____
23
24 for k=1:size(L,1) //number of time points
25     L1=[]; L2=[]; // initialize L1,L2
26     x=L(k,1); y=L(k,2);
27     if ~(isnan(x) | isnan(y)) //do not construct l1,l2 if camx,y=NaN
28         L1=[A(1)-x*A(7), A(2)-x*A(8) : ...
29             A(4)-y*A(7), A(5)-y*A(8) ];
30         L2=[x-A(3);y-A(6)];
31     end
32
33     if (size(L2,1))==1 // check whether data available
34         g=L1\L2;
35     else
36         g=[NaN;NaN];
37     end
38
39     H(k,:)=g';
40 end
41
42 bar1=sqrt((H(3,1)-H(1,1))^2+(H(3,2)-H(1,2))^2);
43 bar2=sqrt((H(5,1)-H(4,1))^2+(H(5,2)-H(4,2))^2);
44 bar3=sqrt((H(7,1)-H(6,1))^2+(H(7,2)-H(6,2))^2);
45
```

Fig. 15.: Code 2 in Scilab Software

bar1	1.93
bar2	0.957
bar3	0.953

Fig. 16.: Bars lengths for the first attempt

bar1	2.07
bar2	1.03
bar3	1.02

Fig. 17.: Bars lengths for the second attempt

bar1	2
bar2	0.989
bar3	0.99

Fig. 18.: Bars lengths for the third attempt

All the results are presented in Fig. 19. It contains the results both from Kinovea and Scilab Software. It is clearly visible that Scilab method is much more precise than Kinovea one. Results from both Kinovea and Scilab methods are similar or almost the same with the real values, so the calibration was made in a proper way.

Real coordinates	Observations	Mathematical method (Kinovea)	Scilab method 2DDL
Bar 1	1	1,97m	1,93m
2m	2	1,95m	2,07m
	3	1,95m	2m
Bar 2	1	0,98m	0,96m
1m	2	0,96m	1,03m
	3	0,97m	0,99m
Bar 3	1	0,96m	0,95m
1m	2	0,95m	1,02m
	3	0,95m	0,99m

Fig. 19.: Results from calibration using Kinovea and Scilab Software

After performing calibration, it was possible to proceed to the next task.

4.3 Quantitative analysis- Obtaining and processing linear kinematic parameters

There are several methods to determine quantitatively the human motion. One of the methods that allow to quantify the human motion is through the kinematics methods (study the human motion without explain the forces action). It can be obtained through several tools, as for example, the cameras, but it is necessary to use mathematical methods to calculate the obtained data.

In this part first it was recorded video in slow motion of human's gait. Then using Kinovea, virtual marker on the ankle was putted and using this, the path of one step for right leg was obtained. Obtaining this, data from Kinovea was putted to Excel. Having this data, it was possible to calculate desired equations and plots. The goal was to determine **position, displacement, velocity and acceleration** of the step and using this, make several plots. It was possible by using first and second derivatives. The obtained signal was very rapid changing, and it demanded to put trendline to make it calmer. However, by using polynomial method in case of acceleration it still wasn't enough to clear the signal. The signal was changing very fast. Reasonable idea would be using in this case Butterworth filter

as a code in Scilab, but we only be focused on Polynomial method of smoothing processing data. The results and some equations are presented below. The measurements were conducted only for person 1. The experiment is presented below (Fig. 20).

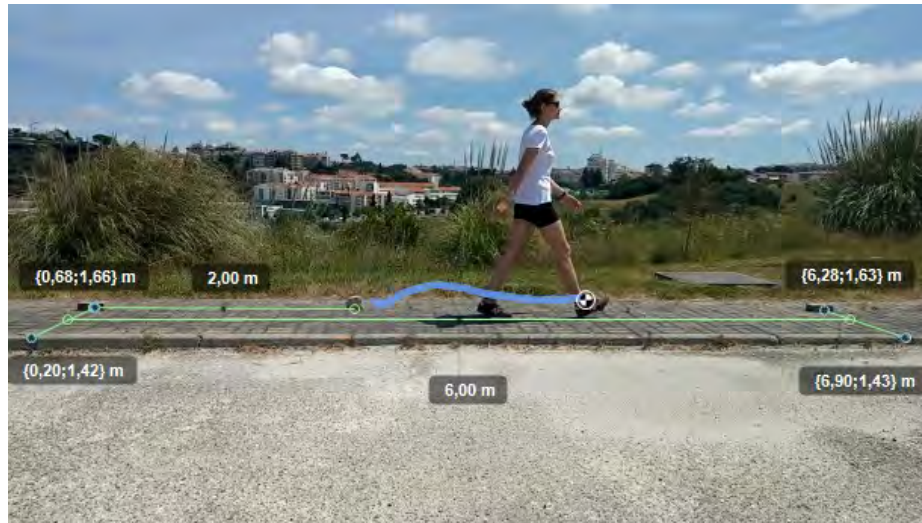


Fig. 20.: Obtaining and processing linear kinematic parameters by using Kinovea Software

D2				fx		=(A4-A2)/(2*(C3-C2))	
	A	B	C	D	E	F	G
1	x[m]	y[m]	t[s]	Vx	Vy	Ax	Ay
2	2,83	1,7	0	0,714286	0	38,26531	-44,6429

Fig. 21.: Equation for obtaining velocity on position x (person 1)

E2				fy		=(B4-B2)/(2*(C3-C2))	
	A	B	C	D	E	F	G
1	x[m]	y[m]	t[s]	Vx	Vy	Ax	Ay
2	2,83	1,7	0	0,714286	0	38,26531	-44,6429

Fig. 22.: Equation for obtaining velocity on position y (person 1)

F2				fx		=(D4-D2)/(2*(C3-C2))	
	A	B	C	D	E	F	G
1	x[m]	y[m]	t[s]	Vx	Vy	Ax	Ay
2	2,83	1,7	0	0,714286	0	38,26531	-44,6429

Fig. 23.: Equation for obtaining acceleration on position x (person 1)

G2				fy		=(E4-E2)/(2*(C3-C2))	
	A	B	C	D	E	F	G
1	x[m]	y[m]	t[s]	Vx	Vy	Ax	Ay
2	2,83	1,7	0	0,714286	0	38,26531	-44,6429

Fig. 24.: Equation for obtaining acceleration on position y (person 1)

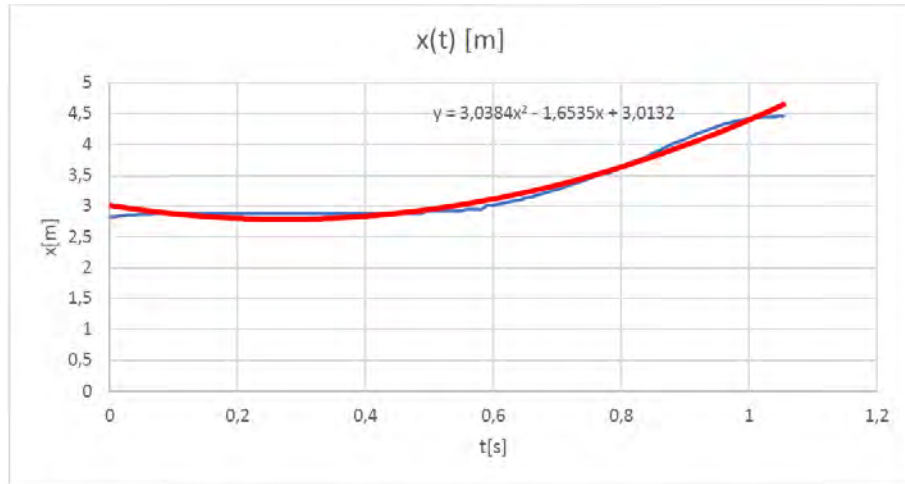


Fig. 25.: Linear position $x(t)$ (person 1)

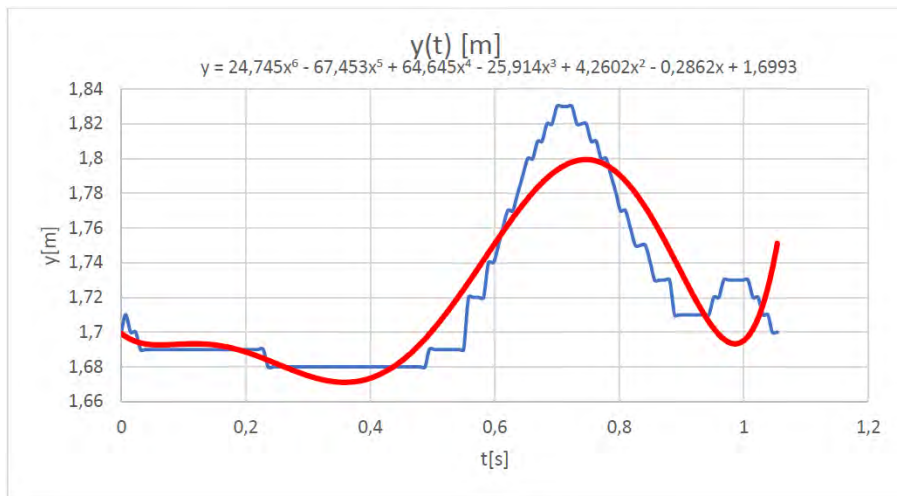


Fig. 26.: Linear position $y(t)$ (person 1)

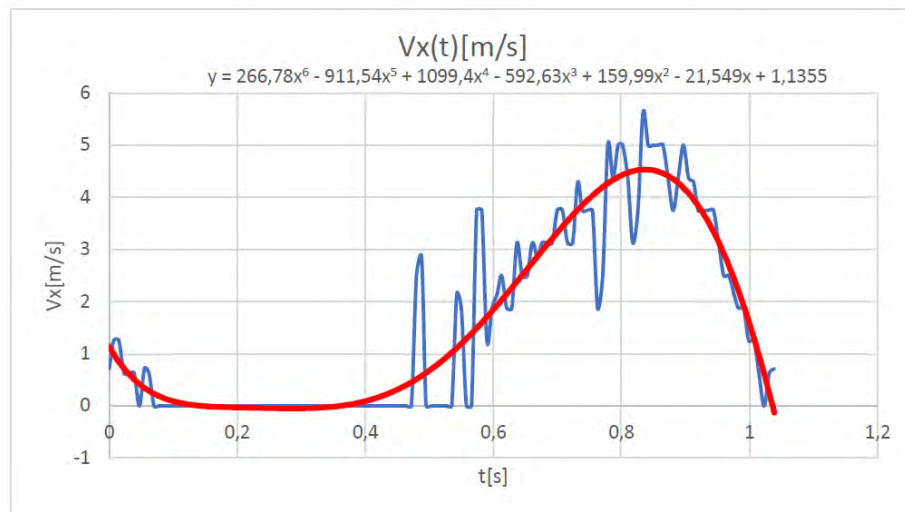


Fig. 27.: Linear velocity $V_x(t)$ (person 1)

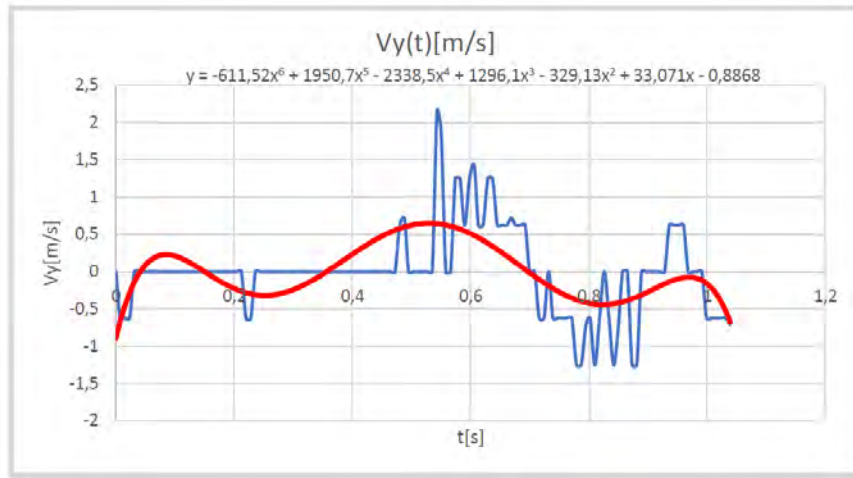


Fig. 28.: Linear velocity $V_y(t)$ (person 1)

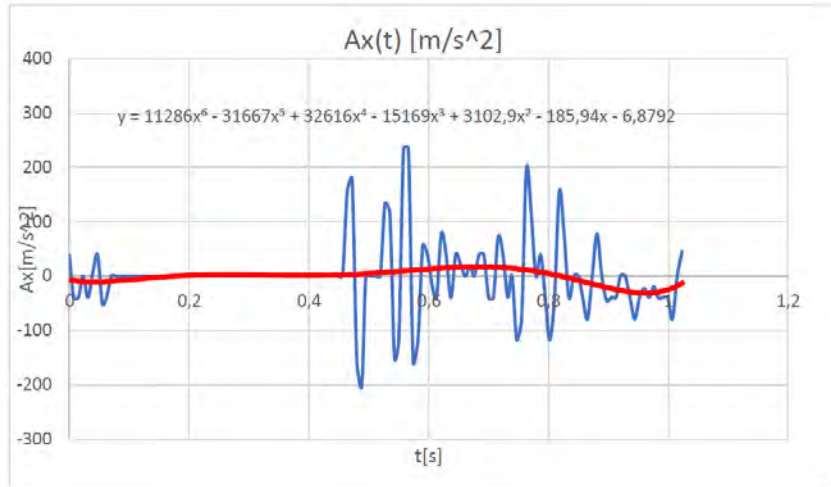


Fig. 29.: Linear acceleration $A_x(t)$ (person 1)

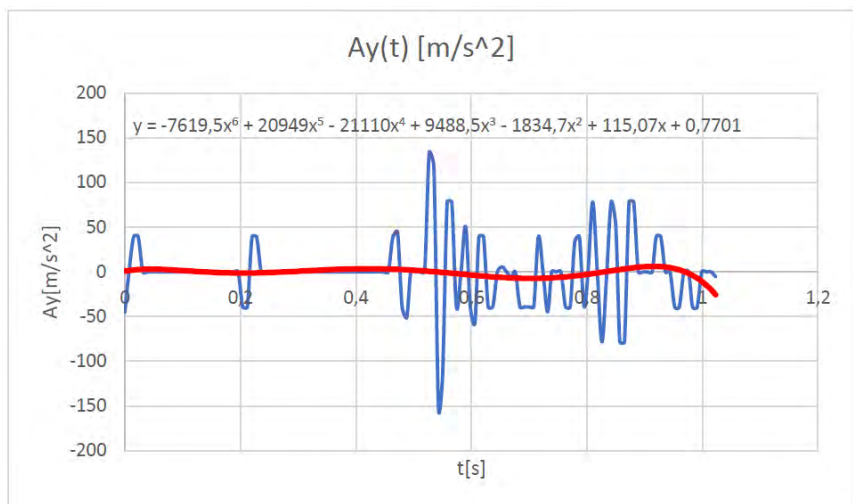


Fig. 30.: Linear acceleration $A_y(t)$ (person 1)

Achieved plots gives view for kinematic movement parameters. On the beginning we obtained „x” and „y” coordinates of ankle. Video record framerate describes time in which displacement had place. By simple math we obtained velocity and acceleration of ankle. All data have been presented on plots, where „x” axis describes time, and „y”: position, velocity and acceleration. Because of rough signal discretion, obtained velocity values are not changing smoothly (their values changes impulsively). Therefore, obtained acceleration values are not credible.

4.4 Quantitative analysis- Obtaining center of mass

The last stage of this work was to obtain information about mass centers of every examined person's body from record gait cycle. To this aim Kinovea and Scilab Software were used.

First part of the task was to develop code in Scilab. This code was able to use the Kinovea digital coordinates and on this basis calculate localization of certain body mass segment. The Scilab code for this purpose is presented below.

```
clc;clear;

//SELECT THE EXCEL FILE WITH DIGITAL COORDINATES
path1=uigetfile(["*.xls"]);
//OPEN XLS FILES WITH DIGITAL COORDINATES
[total_sheet1,txt_vol1,nome_sheet1,pos_sheet1]=xls_open(path1);
//OPEN SHEET WITH DIGITAL COORDINATES
[num_coo1,txt_coo1] = xls_read(total_sheet1,pos_sheet1(1));
L=num_coo1;

//1 right leg mid-foot
//2 right ankle
//3 right knee
//4 right hip
//5 right trunk
//6 right middle finger
//7 right wrist
//8 right elbow
//9 right arm
//11 the top of the head
//10 chin
//12 left mid-foot
//13 left ankle
//14 left knee
//15 left hip
//16 left middle finger
//17 left wrist
//18 left elbow
//19 left arm

RSr_X=L(1,1)
RSr_Y=L(1,2)
Rkostka_X=L(2,1)
Rkostka_Y=L(2,2)
Rkolano_X=L(3,1)
Rkolano_Y=L(3,2)
RHip_X=L(4,1)
RHip_Y=L(4,2)
//Trunk_X=L(5,1)
//Trunk_Y=L(5,2)
RPs_X=L(6,1)
RPs_Y=L(6,2)
RNadg_X=L(7,1)
RNadg_Y=L(7,2)
RLok_X=L(8,1)
RLok_Y=L(8,2)
```

RRamie_X=L(9,1)
RRamie_Y=L(9,2)
Broda_X=L(10,1)
Broda_Y=L(10,2)
CzubekG_X=L(11,1)
CzubekG_Y=L(11,2)
LSr_X=L(12,1)
LSr_Y=L(12,2)
Lkostka_X=L(13,1)
Lkostka_Y=L(13,2)
Lkolano_X=L(14,1)
Lkolano_Y=L(14,2)
LHip_X=L(15,1)
LHip_Y=L(15,2)
LPs_X=L(16,1)
LPs_Y=L(16,2)
LNadg_X=L(17,1)
LNadg_Y=L(17,2)
LLok_X=L(18,1)
LLok_Y=L(18,2)
LRamie_X=L(19,1)
LRamie_Y=L(19,2)

SEG_XFootR=L(2,1)-(L(2,1)-L(1,1))*0.50
SEG_YFootR=L(2,2)-(L(2,2)-L(1,2))*0.50

SEG_XLegR=L(3,1)-(L(3,1)-L(2,1))*0.433
SEG_YLegR=L(3,2)-(L(3,2)-L(2,2))*0.433

SEG_XThighR=L(4,1)-(L(4,1)-L(3,1))*0.433
SEG_YThighR=L(4,2)-(L(4,2)-L(3,2))*0.433

SEG_XForearmR=L(8,1)-(L(8,1)-L(7,1))*0.430
SEG_YForearmR=L(8,2)-(L(8,2)-L(7,2))*0.430

SEG_XHandR=L(7,1)-(L(7,1)-L(6,1))*0.506
SEG_YHandR=L(7,2)-(L(7,2)-L(6,2))*0.506

SEG_XArmR=L(9,1)-(L(9,1)-L(8,1))*0.436
SEG_YArmR=L(9,2)-(L(9,2)-L(8,2))*0.436

SEG_XTrunkSH=(L(9,1)+L(19,1))*0.5
SEG_YTrunkSH=(L(9,2)+L(19,2))*0.5
SEG_XTrunkHip=(L(4,1)+L(15,1))*0.5
SEG_YTrunkHip=(L(4,2)+L(15,2))*0.5
SEG_XTrunk=((SEG_XTrunkSH)+(SEG_XTrunkHip))*0.5
SEG_YTrunk=((SEG_YTrunkSH)+(SEG_YTrunkHip))*0.5

SEG_XHead=L(10,1)-(L(10,1)-L(11,1))*1
SEG_YHead=L(10,2)-(L(10,2)-L(11,2))*1

SEG_XFootL=L(13,1)-(L(13,1)-L(12,1))*0.50
SEG_YFootL=L(13,2)-(L(13,2)-L(12,2))*0.50

SEG_XLegL=L(14,1)-(L(14,1)-L(13,1))*0.433
SEG_YLegL=L(14,2)-(L(14,2)-L(13,2))*0.433

SEG_XThighL=L(15,1)-(L(15,1)-L(14,1))*0.433
SEG_YThighL=L(15,2)-(L(15,2)-L(14,2))*0.433

SEG_XHandL=L(17,1)-(L(17,1)-L(16,1))*0.506
SEG_YHandL=L(17,2)-(L(17,2)-L(16,2))*0.506

$$\text{SEG_XForearmL}=\text{L}(18,1)-(\text{L}(18,1)-\text{L}(17,1))*0.430$$
$$\text{SEG_YForearmL}=\text{L}(18,2)-(\text{L}(18,2)-\text{L}(17,2))*0.430$$

$$\text{SEG_XArmL}=\text{L}(19,1)-(\text{L}(19,1)-\text{L}(18,1))*0.436$$
$$\text{SEG_YArmL}=\text{L}(19,2)-(\text{L}(19,2)-\text{L}(18,2))*0.436$$

$$\text{CM_XLegR}=(\text{SEG_XFootR}*0.015+\text{SEG_XLegR}*0.047+\text{SEG_XThighR}*0.1)/(\text{0.015}+\text{0.047}+\text{0.1})$$
$$\text{CM_YLegR}=(\text{SEG_YFootR}*0.015+\text{SEG_YLegR}*0.047+\text{SEG_YThighR}*0.1)/(\text{0.015}+\text{0.047}+\text{0.1})$$

$$\text{CM_XLegL}=(\text{SEG_XFootL}*0.015+\text{SEG_XLegL}*0.047+\text{SEG_XThighL}*0.1)/(\text{0.162})$$
$$\text{CM_YLegL}=(\text{SEG_YFootL}*0.015+\text{SEG_YLegL}*0.047+\text{SEG_YThighL}*0.1)/(\text{0.162})$$

$$\text{CM_XLegRL}=(\text{SEG_XFootR}*0.015+\text{SEG_XLegR}*0.047+\text{SEG_XThighR}*0.1)+(\text{SEG_XFootL}*0.015+\text{SEG_XLegL}*0.047+\text{SEG_XThighL}*0.1)/(\text{2}*(\text{0.015}+\text{0.047}+\text{0.1}))$$
$$\text{CM_YLegRL}=(\text{SEG_YFootR}*0.015+\text{SEG_YLegR}*0.047+\text{SEG_YThighR}*0.1)+(\text{SEG_YFootL}*0.015+\text{SEG_YLegL}*0.047+\text{SEG_YThighL}*0.1)/(\text{2}*(\text{0.015}+\text{0.047}+\text{0.1}))$$

$$\text{CM_XHandR}=(\text{SEG_XHandR}*0.006+\text{SEG_XArmR}*0.028+\text{SEG_XForearmR}*0.016)/(\text{0.006}+\text{0.028}+\text{0.016})$$
$$\text{CM_YHandR}=(\text{SEG_YHandR}*0.006)+(\text{SEG_YArmR}*0.028)+(\text{SEG_YForearmR}*0.016)/(\text{0.006}+\text{0.028}+\text{0.016})$$

$$\text{CM_XHandL}=(\text{SEG_XHandL}*0.006)+(\text{SEG_XArmL}*0.028)+(\text{SEG_XForearmL}*0.016)/(\text{0.006}+\text{0.028}+\text{0.016})$$
$$\text{CM_YHandL}=(\text{SEG_YHandL}*0.006)+(\text{SEG_YArmL}*0.028)+(\text{SEG_YForearmL}*0.016)/(\text{0.006}+\text{0.028}+\text{0.016})$$

$$\text{CM_XHandRL}=(\text{SEG_XHandR}*0.006+\text{SEG_XArmR}*0.028+\text{SEG_XForearmR}*0.016)+(\text{SEG_XHandL}*0.006)+(\text{SEG_XArmL}*0.028)+(\text{SEG_XForearmL}*0.016)/(\text{2}*(\text{0.006}+\text{0.028}+\text{0.016}))$$
$$\text{CM_YHandRL}=(\text{SEG_YHandR}*0.006)+(\text{SEG_YArmR}*0.028)+(\text{SEG_YForearmR}*0.016)+(\text{SEG_YHandL}*0.006)+(\text{SEG_YArmL}*0.028)+(\text{SEG_YForearmL}*0.016)/(\text{2}*(\text{0.006}+\text{0.028}+\text{0.016}))$$

$$\text{CM_UpperBodyX}=(\text{SEG_XHead}*0.081)+(\text{SEG_XTrunk}*0.497)+(\text{SEG_XHandR}*0.006)+(\text{SEG_XArmR}*0.028)+(\text{SEG_XForearmR}*0.016)+(\text{SEG_XHandL}*0.006)+(\text{SEG_XArmL}*0.028)+(\text{SEG_XForearmL}*0.016)/(\text{0.081}+\text{0.497}+\text{0.006}+\text{0.028}+\text{0.016}+\text{0.06}+\text{0.028}+\text{0.016})$$

$$\text{CM_UpperBodyY}=(\text{SEG_YHead}*0.081)+(\text{SEG_YTrunk}*0.497)+(\text{SEG_YHandR}*0.006)+(\text{SEG_YArmR}*0.028)+(\text{SEG_YForearmR}*0.016)+(\text{SEG_YHandL}*0.006)+(\text{SEG_YArmL}*0.028)+(\text{SEG_YForearmL}*0.016)/(\text{0.081}+\text{0.497}+\text{0.006}+\text{0.028}+\text{0.016}+\text{0.06}+\text{0.028}+\text{0.016})$$

$$\text{CM_AllBodyX}=(\text{SEG_XFootR}*0.015)+(\text{SEG_XLegR}*0.047)+(\text{SEG_XThighR}*0.1)+(\text{SEG_XFootL}*0.015)+(\text{SEG_XLegL}*0.047)+(\text{SEG_XThighL}*0.1)+(\text{SEG_XHead}*0.081)+(\text{SEG_XTrunk}*0.497)+(\text{SEG_XHandR}*0.006)+(\text{SEG_XArmR}*0.028)+(\text{SEG_XForearmR}*0.016)+(\text{SEG_XHandL}*0.006)+(\text{SEG_XArmL}*0.028)+(\text{SEG_XForearmL}*0.016)/(\text{0.015}+\text{0.047}+\text{0.1}+\text{0.015}+\text{0.047}+\text{0.1}+\text{0.081}+\text{0.497}+\text{0.006}+\text{0.028}+\text{0.016}+\text{0.006}+\text{0.028}+\text{0.016})$$

$$\text{CM_AllBodyY}=(\text{SEG_YFootR}*0.015)+(\text{SEG_YLegR}*0.047)+(\text{SEG_YThighR}*0.1)+(\text{SEG_YFootL}*0.015)+(\text{SEG_YLegL}*0.047)+(\text{SEG_YThighL}*0.1)+(\text{SEG_YHead}*0.081)+(\text{SEG_YTrunk}*0.497)+(\text{SEG_YHandR}*0.006)+(\text{SEG_YArmR}*0.028)+(\text{SEG_YForearmR}*0.016)+(\text{SEG_YHandL}*0.006)+(\text{SEG_YArmL}*0.028)+(\text{SEG_YForearmL}*0.016)/(\text{0.015}+\text{0.047}+\text{0.1}+\text{0.015}+\text{0.047}+\text{0.1}+\text{0.081}+\text{0.497}+\text{0.006}+\text{0.028}+\text{0.016}+\text{0.006}+\text{0.028}+\text{0.016})$$

//Right Leg

plot([RSr_X Rkostka_X],[RSr_Y Rkostka_Y])
plot([Rkostka_X Rkolano_X],[Rkostka_Y Rkolano_Y])
plot([Rkolano_X RHip_X],[Rkolano_Y RHip_Y])
plot(SEG_XFootR,SEG_YFootR,'ro')
plot(SEG_XLegR,SEG_YLegR,'ro')
plot(SEG_XThighR,SEG_YThighR,'ro')

//Left Leg

plot([LSr_X Lkostka_X],[LSr_Y Lkostka_Y])
plot([Lkostka_X Lkolano_X],[Lkostka_Y Lkolano_Y])
plot([Lkolano_X LHip_X],[Lkolano_Y LHip_Y])
plot(SEG_XFootL,SEG_YFootL,'ro')
plot(SEG_XLegL,SEG_YLegL,'ro')
plot(SEG_XThighL,SEG_YThighL,'ro')

//Right hand

plot([RPs_X RNadg_X],[RPs_Y RNadg_Y])

```
plot([RNadg_X RLok_X],[RNadg_Y RLok_Y])
plot([RLok_X RRamie_X],[RLok_Y RRamie_Y])
plot(SEG_XHandR,SEG_YHandR, 'ro')
plot(SEG_XForearmR,SEG_YForearmR, 'ro')
plot(SEG_XArmR,SEG_YArmR, 'ro')
//Left hand
plot([LPs_X LNadg_X],[LPs_Y LNadg_Y])
plot([LNadg_X LLok_X],[LNadg_Y LLok_Y])
plot([LLok_X LRamie_X],[LLok_Y LRamie_Y])
plot(SEG_XHandL,SEG_YHandL, 'ro')
plot(SEG_XForearmL,SEG_YForearmL, 'ro')
plot(SEG_XArmL,SEG_YArmL, 'ro')
//Trunk
plot([RRamie_X LHip_X],[RRamie_Y LHip_Y])
plot([LRamie_X RHip_X],[LRamie_Y RHip_Y])
plot(SEG_XTrunk,SEG_YTrunk, 'ro')
//Head
plot([Broda_X CzubekG_X],[Broda_Y CzubekG_Y])
plot(SEG_XHead,SEG_YHead, 'ro')

plot(CM_XLegR,CM_YLegR, 'go')
plot(CM_XLegL,CM_YLegL, 'go')
plot(CM_XLegRL,CM_YLegRL, 'go')

plot(CM_XHandR,CM_YHandR, 'go')
plot(CM_XHandL,CM_YHandL, 'go')
plot(CM_XHandRL,CM_YHandRL, 'go')
plot(CM_AllBodyX,CM_AllBodyY, 'ko')
```

Next stage was to make digitalization of every phase (IC, OT, HS, OIC, TO, FA, TV). For every phase there were putted 19 points on body, every time selecting them in Kinovea in the same order. Having knowledge about distal and proximal points it was possible to obtain localization of center of mass of specific part of body.

Location in X:

$$\text{Seg}_x = \text{seg_proximal} - (\text{seg_proximal} - \text{seg_distal}) * \text{CM_seg} \quad (1)$$

Location in Y:

$$\text{Seg}_y = \text{seg_proximal} - (\text{seg_proximal} - \text{seg_distal}) * \text{CM_seg} \quad (2)$$

Values CM_seg we took from table (Fig.31).

TABLE 3.1 Anthropometric Data

Segment	Definition	Segment Weight/ Total Body Weight	Center of Mass/ Segment Length		Radius of Gyration/ Segment Length			Density
			Proximal	Distal	C of G	Proximal	Distal	
Hand	Wrist axis/knuckle II middle finger	0.006 M	0.506	0.494 P	0.297	0.587	0.577 M	1.16
Forearm	Elbow axis/ulnar styloid	0.016 M	0.430	0.570 P	0.303	0.526	0.647 M	1.13
Upper arm	Glenohumeral axis/elbow axis	0.028 M	0.436	0.564 P	0.322	0.542	0.645 M	1.07
Forearm and hand	Elbow axis/ulnar styloid	0.022 M	0.682	0.318 P	0.468	0.827	0.565 P	1.14
Total arm	Glenohumeral joint/ulnar styloid	0.050 M	0.530	0.470 P	0.368	0.645	0.596 P	1.11
Foot	Lateral malleolus/head metatarsal II	0.0145 M	0.50	0.50 P	0.475	0.690	0.690 P	1.10
Leg	Femoral condyles/medial malleolus	0.0465 M	0.433	0.567 P	0.302	0.528	0.643 M	1.09
Thigh	Greater trochanter/femoral condyles	0.100 M	0.433	0.567 P	0.323	0.540	0.653 M	1.05
Foot and leg	Femoral condyles/medial malleolus	0.061 M	0.606	0.394 P	0.416	0.735	0.572 P	1.09
Total leg	Greater trochanter/medial malleolus	0.161 M	0.447	0.553 P	0.326	0.560	0.650 P	1.06

Fig. 31.: Table with anthropometric data (marked black is our CM_seg coefficient) [5]

Knowing Seg_x and Seg_y we can apply values, which corresponds to weight of certain parts of body:

CM_X:

$$CM_x = \frac{\sum(seg_x * mass_seg)}{\sum(mass)} \quad (3)$$

CM_Y:

$$CM_y = \frac{\sum(seg_y * mass_seg)}{\sum(mass)} \quad (4)$$

Values mass_seg were taken from Fig. 32.

Head and neck	C7-T1 and 1st rib/ear canal	0.081 M	1.000	—	PC	0.495	1.116	—	PC	1.11
Shoulder mass	Sternoclavicular joint/ glenohumeral axis	—	0.712	0.288	—	—	—	—	—	1.04
Thorax	C7-T1/T12-L1 and diaphragm*	0.216 PC	0.82	0.18	—	—	—	—	—	0.92
Abdomen	T12-L1/L4-L5*	0.139 LC	0.44	0.56	—	—	—	—	—	—
Pelvis	L4-L5/greater trochanter*	0.142 LC	0.105	0.895	—	—	—	—	—	—
Thorax and abdomen	C7-T1/L4-L5*	0.355 LC	0.63	0.37	—	—	—	—	—	—
Abdomen and pelvis	T12-L1/greater trochanter*	0.281 PC	0.27	0.73	—	—	—	—	—	1.01
Trunk	Greater trochanter/ glenohumeral joint*	0.497 M	0.50	0.50	—	—	—	—	—	1.03
Trunk head neck	Greater trochanter/ glenohumeral joint*	0.578 MC	0.66	0.34 P	0.503	0.830	0.607 M	—	—	—
HAT	Greater trochanter/ glenohumeral joint*	0.678 MC	0.626	0.374 PC	0.496	0.798	0.621 PC	—	—	—
HAT	Greater trochanter/mid rib	0.678	1.142	—	0.903	1.456	—	—	—	—

Fig. 32.: Table with mass of each body segment [5]

```
CM_XLegR=(SEG_XFootR*0.015+SEG_XLegR*0.047+SEG_XThighR*0.1)/(0.015+0.047+0.1)
CM_YLegR=(SEG_YFootR*0.015+SEG_YLegR*0.047+SEG_YThighR*0.1)/(0.015+0.047+0.1)
```

Fig. 33.: Part of code calculating centre of mass for right leg

When all data was collected, it was possible to make a plot, which shows how centers of masses of every body segment looks in every phase. Results are presented below.

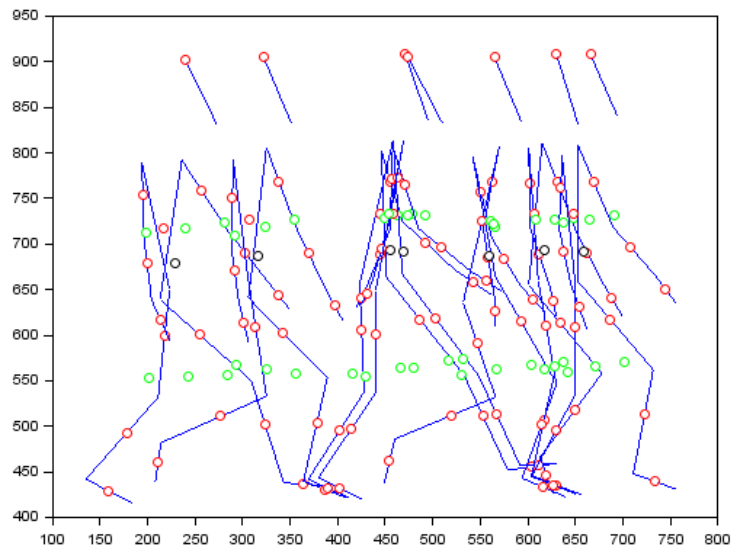


Fig. 34.: Result plot of program for every phase (from left IC, OT, HS, OIC, TO, FA, TV) for man of height 175cm and weight 75kg. Circle green signs mass center for upper and down body part, and black circle is all body mass center

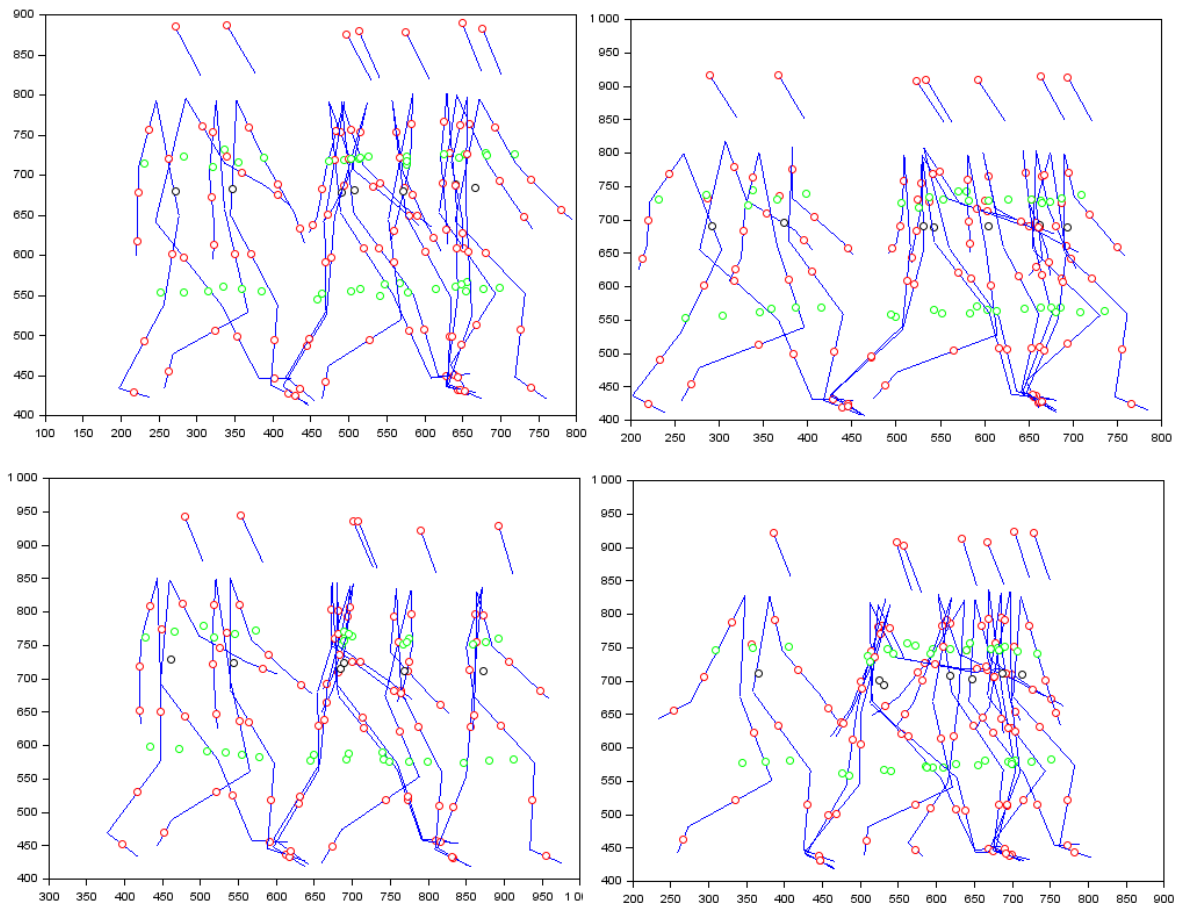


Fig. 35.: Comparison plots for rest of examined persons

Every person has point of mass center in a different place. Every of us has different body weight, anatomy, natural gait speed, and because of this every plot is different from another. To compare our results we put below the representation of center mass point while walking (Fig. 36).

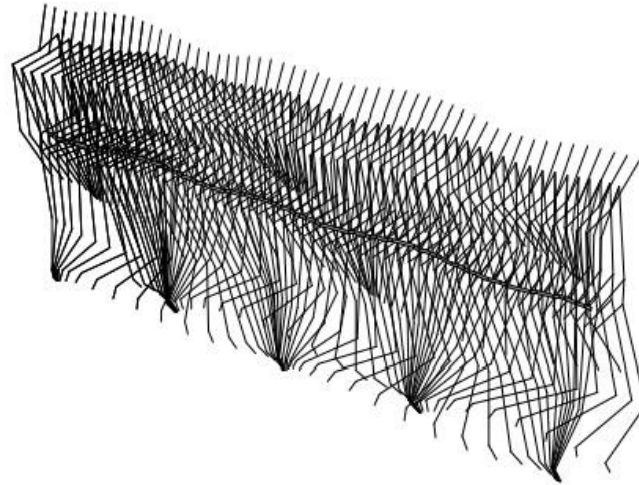


Fig. 36.: Model of subject during natural walking with drawn calculated trajectory of body center of mass [3]

5 Conclusions

Normal Gait is series of rhythmical , alternating movements of the trunk & limbs which result in the forward progression of the center of gravity and the body. Biomechanics of movement can be classified into two main areas: the improvement of performance and the reduction or treatment of injury. [7] Biomechanical research is a powerful ally in the sports medicine quest to prevent and treat injury. Biomechanical studies help prevent injuries by providing information on the mechanical properties of tissues, mechanical loadings during movement, and preventative or rehabilitative therapies. Biomechanical studies provide important data to confirm potential injury mechanisms hypothesized by sports medicine physicians. [7] People need help in improving human movement and this help requires knowledge of "why" and "how" the human body moves. Since biomechanics gives the kinesiology professional much of the knowledge and many of the skills necessary to answer these "what works?" and "why?" questions, biomechanics is an important science for solving human movement problems. [7]

The advantages of numerical measurements of quantitative over those of qualitative analysis are greater accuracy, consistency, and precision. Most quantitative biomechanical analysis is performed in research settings; however, more and more devices are commercially available that inexpensively measure some biomechanical variables (e.g., radar, timing lights, timing mats, quantitative videography systems). Unfortunately, the greater accuracy of quantitative measures comes at the cost of technical skills, calibration, computational and processing time, as well as dangers of increasing errors with the additional computations involved. Even with very fast modern computers, quantitative biomechanics is a labor-intensive task requiring considerable graduate training and experience. For these reasons and others, qualitative analysis of human movement remains the main approach kinesiology professionals use in solving most human movement problems. [7]

Qualitative and quantitative analysis of human gait makes possible to convert real body movement into digital path, what creates much more possibilities of analysis. Two methods which have been used in this article: Kinovea and 2DDLTL are great tools for visualization of human gait movement. Comprising these two approaches, Kinovea is easier and fastest way to get the results. Accuracy of results mostly depends on camera resolution and precision in obtaining points of body parts.

References

- [1] GARCIA-ZAPRAIN B., HERRA A.M., MENDEZ- ZORRILLA A.: Gait Analysis Methods: An Overview of Wearable and Non-Wearable Systems Highlighting Clinical Applications, „ Sensors (Basel)”, 2014, volume 14(2), p. 3362–3394.
- [2] GARD S.: Use of Quantitative Gait Analysis for the Evaluation of Prosthetic Walking Performance, „ JPO Journal of Prosthetics and Orthotics”, 2006, volume 18(6), p. 93-104.
- [3] JURCEVIC LULIC T., MUFTIC O.: Trajectory of the human body mass centre during walking at different speed , International Design Conference, Dubrovnik, May 14 - 17, 2002.
- [4] WHITTLE M.W.: Gait Analysis An Introduction, fourth edition, Elsevier Butterworth-Heinemann, 2007, ISBN: 9 780 7506 8883 3.
- [5] WINTER D.A.: Biomechanics And Motor Control Of Human Movement, fourth edition, John Wiley & Sons, 2009, ISBN: 978-0-470-39818-0.
- [6] website: <https://us.humankinetics.com/blogs/excerpt/use-qualitative-anatomical-analysis-to-identify-potential-performance-issues> [access: 02.11.2020]
- [7] website: <https://www.barnardhealth.us/qualitative-analysis/what-is-biomechanics.html> [access: 02.11.2020]
- [8] website: <https://www.crossfit.com/essentials/movement-about-joints-part-7-the-ankle> [access: 02.11.2020]
- [9] website: <https://www.slideshare.net/shimaa2022/gait-analysis-15743497> [access: 02.11.2020]
- [10] website: <https://whatis.techtarget.com/definition/calibration> [access: 02.11.2020]

INDUSTRIAL IOT WEB APPLICATION FOR MOTOR CHARACTERISTIC MONITORING

Sami Jradi (Bc)¹

¹ *sami.jradi@fs.cvut.cz*

Abstract:

This paper is based on the technology of web development to build the back-end of an application for motor characteristic monitoring in the field of industrial IoT. The purpose of the web application is to receive data such as velocity, acceleration, position, and torque from an industrial servo motor connected to a Siemens SIMOTION D PLC using MQTT protocol, and then analyze this data in the form of charts. The resulting chart would be known as the load curve which will offer the chance to perform predictive maintenance on the hardware functioning inside the factory.

Keywords:

speed, torque, IoT, MQTT, PLC, characteristic, backend, java, SQL

1 Introduction

This paper discusses the development of a web application for motor sizing and characteristic monitoring in the field of industrial IoT. The application will connect to the PLC and read the drive data of the motor being controlled by the PLC. These data will include the speed, acceleration, position, and torque. This data will be represented in charts. The speed vs torque chart is called the load curve and will be compared to torque characteristic of the motor from the catalogue, additionally the operating point will be plotted as well. This comparison enables the user to identify whether the proper motor has been selected for the respective application.

2 Load Profile and Operating Point

The load profile is used for the purpose of motor sizing. Proper sizing is a crucial aspect of motor selection. If a motor is undersized, it will not be able to control the load, leading to overshoot and ringing. If the motor is oversized, it may control the load, but it will also be larger and heavier, as well as more expensive in terms of price and cost of operations. [1]

Every motor will have rated values of voltage, current, speed and power. Normally these are visible on the motor's nameplate and/or given in the documentation. In general, rated values represent the maximum values that the motor should be subjected to in normal conditions. However, the rated torque is often not given on the nameplate but is a very important parameter for appropriately sizing the motor. In both DC and AC induction motors, operating current is proportional to the torque, so exceeding the rated torque is likely to lead to overheating and burnout of the motor windings. Exceeding the rated torque also risks mechanical damage to couplings and the drive shaft. Simply, if the load is constant then sizing the motor consists of choosing a motor whose rated torque is slightly above the torque required by the load. The torque produced by a motor varies with speed and the torque produced by a load also varies with speed. If the motor torque is greater than the load torque, then the load will accelerate. If the load torque is greater than the motor torque, then the load will decelerate.

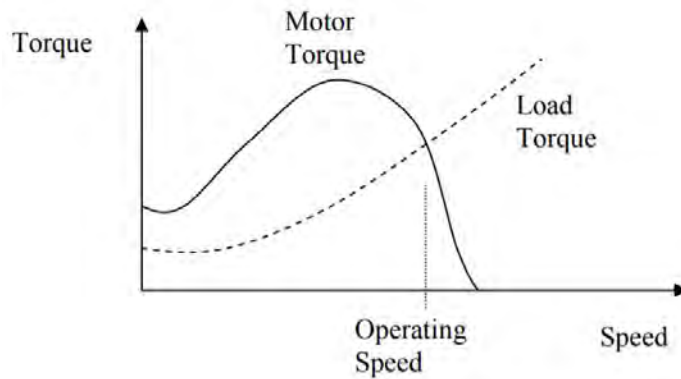


Figure 1 A motor that will start the load and get up to speed correctly

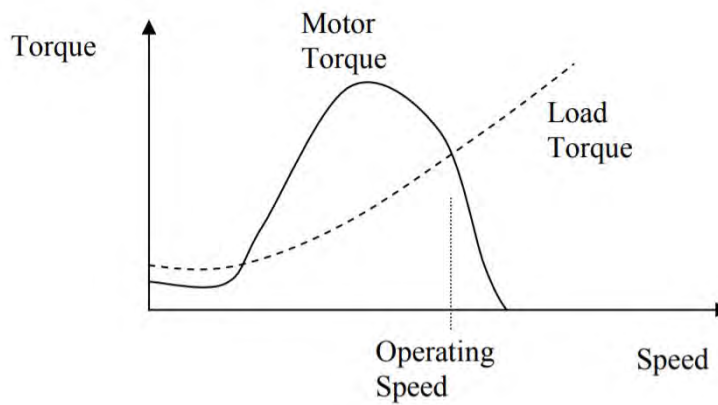


Figure 2 A motor that will never start

Some loads do not present a constant torque even after they have got up to full speed. This presents a variable power to the motor and complicates the sizing problem. In this case we should ensure that:

- Peak load torque < Rated motor torque
- The effective load torque (root mean square load torque) and effective load speed requirements must be less than 100% of the rated motor torque and speed and ideally greater than 75%.
- The motor can start the load and get it up to speed from the initial position.

The point at which the effective torque and effective speed meet is called the load point or operating point. The coordinates of this point can be calculated by the following formulas:

$$M_{eff} = \sqrt{\frac{1}{T} \cdot \sum_{i=1}^n M_i^2 \cdot t_i} \quad (1)$$

$$n_{eff} = \sqrt{\frac{1}{T} \cdot \sum_{i=1}^n n_i^2 \cdot t_i} \quad (2)$$

T ... cycle time of the measurement

t ... sample time

n ... number of samples

If the motor were to be operated at this operating point, the same temperature rise would occur as in the actual load cycle. This point is now entered in its torque-speed diagram for each motor that is considered in the sizing process. If the operating point is below the characteristic curve for S1 operation, the corresponding motor is able to meet the required load cycle from a thermal point of view. All motors that meet this condition are retained for further design steps while all others are eliminated. In figure 8, we can observe that motor 1 is suitable for the load cycle while motor 2 would be thermally overloaded and therefore cannot be used.

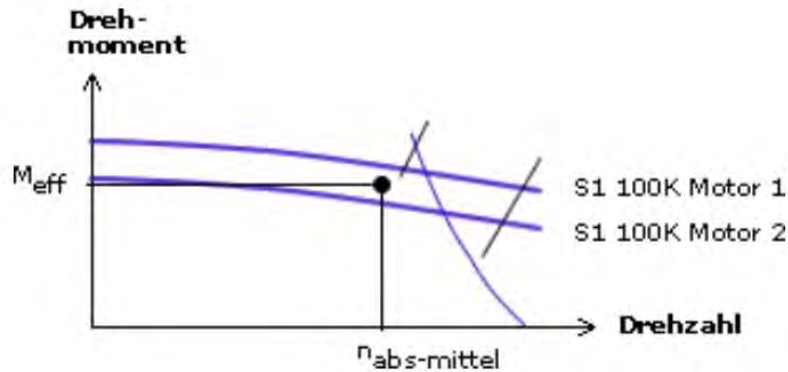


Figure 1 Characteristic of Motor 1 and Motor 2 [2]

To get a better idea of the required motor for a specific application, then the load torque and speed will be compared to the torque characteristic of the motor. This is done by measuring the torque and speed of the motor during operation and then converting these measurements to their absolute values. We call this the load curve, and it represents the possible effective torque vs the possible effective speed.

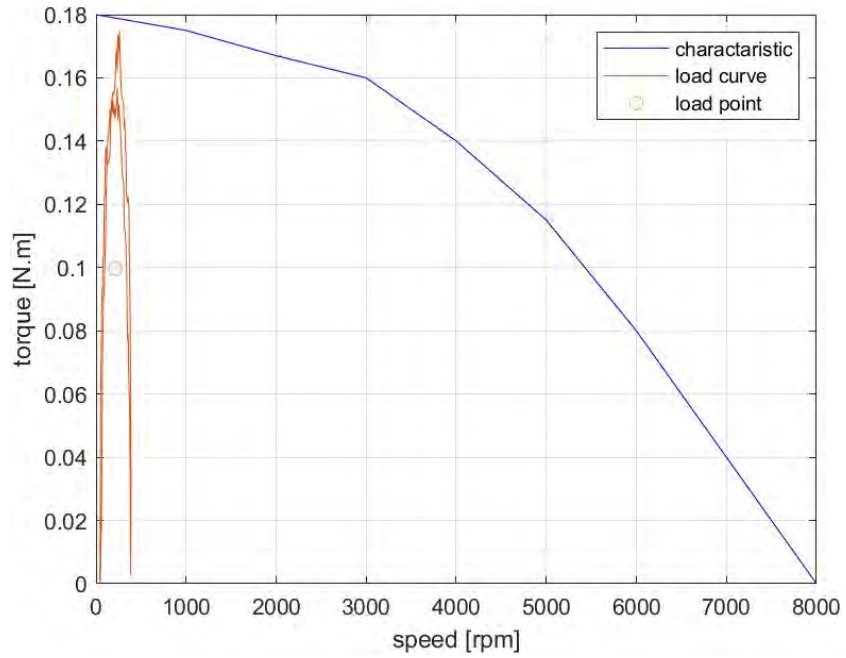


Figure 2 An example of a load curve below the characteristic

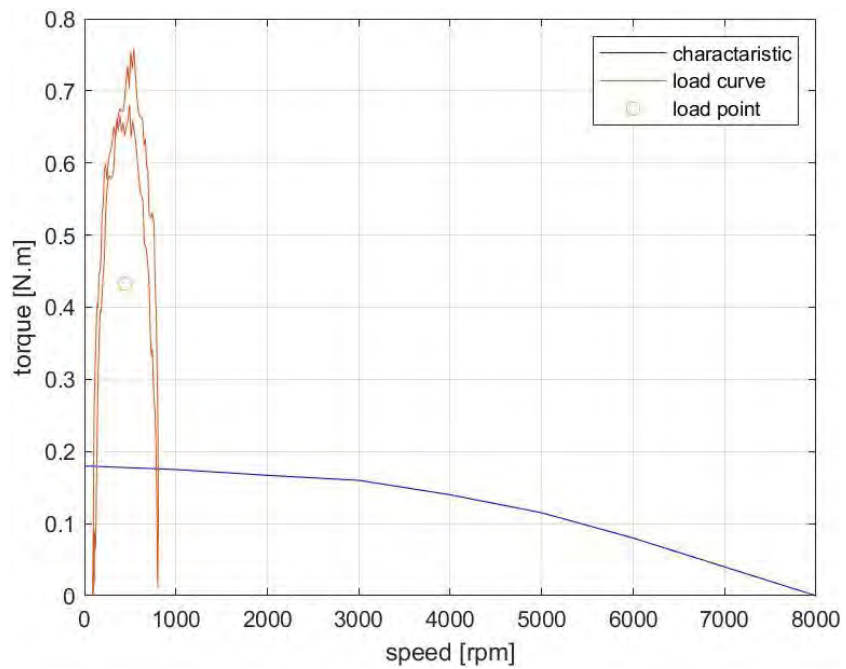


Figure 3 An example of a load curve above the characteristic

In figure 2, we can observe that the load curve or load profile is directly lower than the characteristic, this marks this motor as suitable for operation since the load/operating point falls below the characteristic as well. While in figure 3, the chosen motor should not be used at all and more sizing measurements have to be performed to find a better suited one for the required application.

3 Architectural Overview

This chapter describes an architectural overview of the application. Based on the requirements of the software, the development of the web application has been considered. The architecture consists of three main components: the back-end and front-end systems communicating with an API controller, and the database. The web application will be built as a “3-tier architecture” which comprises from a data tier, business tier and a presentation tier.

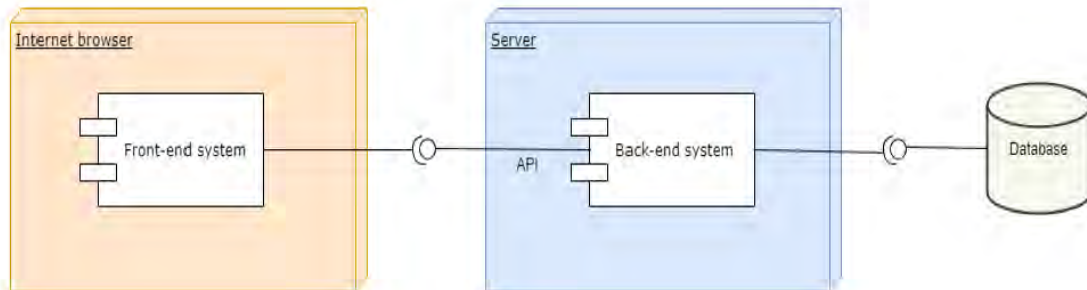


Figure 4 An overview of the proposed architecture

3.1 The Back-end

The server side of a website is referred to as the backend. It organizes and stores data, as well as ensuring that everything on the client side of the website functions properly. It's the section of the website you can't see or communicate with. It's the part of the app that doesn't interact with users directly. The parts and characteristics developed by backend designers are indirectly accessed by users through a front-end application. The backend includes activities such as writing APIs, developing libraries, and dealing with machine components without user interfaces or even scientific programming systems.

The back-end uses the Model-View-Controller (MVC) architectural pattern. The MVC pattern in Software Engineering Architecture is defined as an application being separated into three logical components: Model, View and Controller.

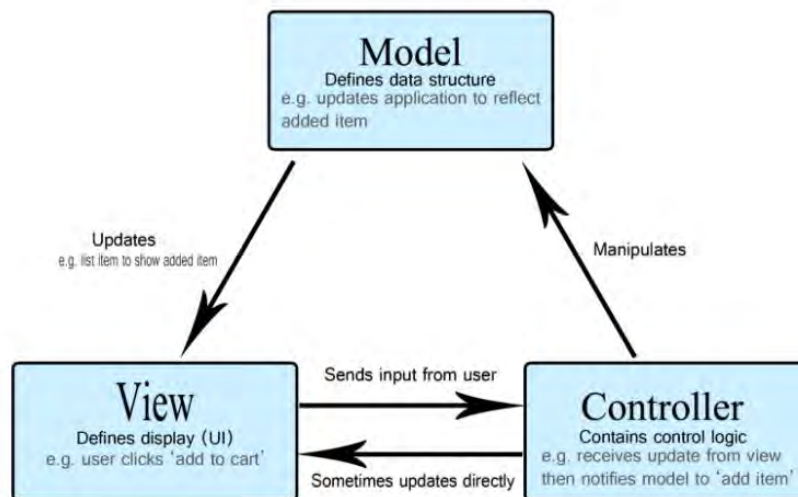


Figure 5 MVC Architecture [3]

3.2 The Front-end

The front end of a website is the aspect where the user communicates with directly. It is often referred to as the application's "client side." Anything that users see explicitly is included: text colors and styles, images, graphs and tables, buttons, colors, and the navigation menu. The languages used for Front End development are HTML, CSS, and JavaScript. Front End developers create the structure, design, behavior, and content of anything that appears on the browser screen when websites, web applications, or mobile apps are opened. The front end's main goals are responsiveness and results.

3.3 Database

SQL stands for Structured Query Language. A query language is a kind of programming language that's designed to facilitate retrieving specific information from databases. To put it in other words, SQL is the language of databases. A relational (SQL) database is a database that stores related information across multiple tables and allows you to query information in more than one table at the same time.

4 Implementation and Results

After building the web application, this section will focus on the functionalities of the app and the working principle behind them. The app will consist of four screens and one pop up window for configuration and setup. For the purpose of demonstration, some demo data has been imported into the app.

4.1 Configuration Screen

The configuration screen is used to connect to a PLC, browse the variables stored in its memory and create the setup of a motor axis for later measurements.

Figure 6 Connect new axis pop-up window

In figure 6 we can see the various fields which need to be filled before a setup can be saved. The first step is to choose a name for our machine and then enter its IP address and hit the connect button. This action will send a GET

request to the server and the server will respond by connecting to the machine and retrieving the required data. This data is then sent back to the front end as a response body in the form of JSON. An example of the response should be as follows:

```
{
  "type": "D445",
  "ipAddress": "192.168.1.7",
  "serialNumber": "ST-KN6031549",
  "firmware": "V 5.3.1.9",
  "status": "CONNECTED",
  "statusTime": "2021-04-12T20:43:33.663+02:00",
  "machineNodes": [
    {
      "node": "Red_Axis",
      "saved": false
    },
    {
      "node": "Blue_Axis",
      "saved": false
    }
  ]
}
```

The response includes a property called “machineNodes”. This property is a list of the available axes inside this machine and each axis contains a Boolean variable showing whether this axis has been already configured and saved or not. If the variable “saved” is false, then we will be able to select the axis from the list in the pop-up window. This action will produce another GET request to the server where the response is a JSON containing all information needed about the axis.

```
{
  "name": "Red_Axis",
  "motorType": "1FK7 synchronous motor",
  "motorMLFB": "1FK7022-5AK7x",
  "motorCode": "23726",
  "axisType": "LINEAR",
  "gearRatio": 10.0,
  "maxSpeed": 200000.0,
  "maxAcceleration": 10000.0,
  "maxJerk": 500.0,
  "dcLinkVoltage": 328.0,
  "possibleVoltages": "600.0,540.0,650.0,720.0"
}
```

4.2 Overview Screen

In the overview screen, we will display a list of all measurements of each axis connected. These measurements are known as “Traces” and each “Trace” object will contain information about the axis as well as a list of values of the measurements done. When loading the Overview page, the server will send a response of a list of all available “Trace” object. For this section a demo axis has been created with mock data and it’s would be as follows:

```

{
  "id": 92,
  "axisName": "Axis_Demo1",
  "axisType": "LINEAR",
  "motorMLFB": "1FK7011-5AK7x",
  "gearRatio": 200.0,
  "maxSpeed": 7200000.0,
  "maxAcceleration": 360000.0,
  "maxJerk": 18000.0,
  "dcLinkVoltage": 600.0,
  "machineName": "My Demo Machine 1",
  "measurements": {
    "id": 87,
    "measurementName": "Load Curve Edge App",
    "traceStartType": "IMMEDIATE",
    "duration": 166,
    "ipAddress": "192.168.1.8",
    "tracejobId": 101,
    "cycletime": 1,
    "timestamp": 702250513000,
    "machineSerialNumber": "ST-M123456789",
    "machineFirmwareVersion": "V 5.3.1.9"
  },
  "loadCurve": {
    "id": 93,
    "datetime": 702250513000,
    "cycleTime": 1
  },
  "timestamp": 702250513000,
  "duration": 166
}

```

This response contains an object “Measurements” which includes all information about the measurements received over MQTT. However, in this response the list of values of the measurements is being suppressed to avoid the congestion of too many data being sent on each request. To access this data an “Analyse” button will be available next to each measurement which sends two responses. One response will include the values for the position, speed and acceleration charts and the second response will include the values for the torque characteristic, load curve and operating point. The responses are as follows:

```

{
  "velocity": {
    "axisX": [...],
    "axisY": [...]
  },
  "acceleration": {
    "axisX": [...],
    "axisY": [...]
  },
}

```

```
"position": {  
  "axisX": [...],  
  "axisY": [...]  
}  
}  
  
{  
  "torqueCharacteristic": {  
    "axisX": [...],  
    "axisY": [...]  
  },  
  "loadProfile": {  
    "axisX": [...],  
    "axisY": [...]  
  },  
  "effectiveSpeed": 606.8984425107654,  
  "effectiveTorque": 0.17676690952747628  
}
```

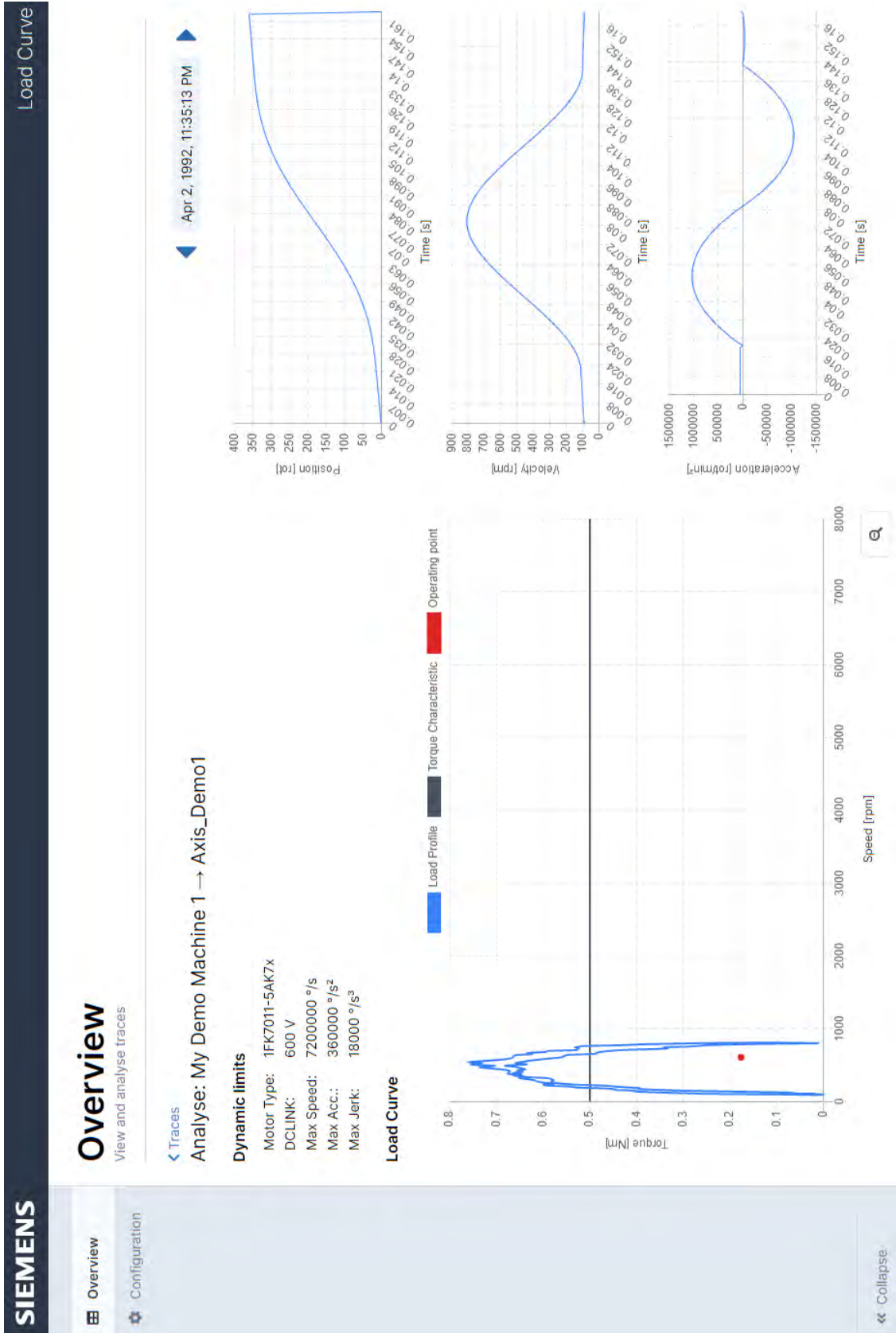


Figure 7 A full screenshot of the overview Analyse window

5 Conclusion

This paper has examined the possibility of developing an Industrial IoT web application used in the field of factory automation and production machines. The application is able to connect to a PLC using its OPC server and read variables stored inside its memory. By selecting an axis from the list, the application is then able to detect the properties of this axis and allow the user to save the configuration. After finalizing the setup, the user is then able to send a measurement over MQTT to the application and perform visualization of the data in charts. The application has proved to be functional, practical and user friendly when tested with a real-life factory setup. Future work can be done to enhance the usability of the application such as adding compatibility with other types of PLCs other than SIMOTION and allowing the user to trigger measurements directly from the application rather than sending the data from another application using MQTT.

REFERENCES

- [1] G. Kirckof, Servomotor Sizing and Application, International Society of Automation, 2012.
- [2] „Servo Technik,“ [Online]. Available: http://www.servotechnik.de/fachwissen/auslegung/f_beitr_00_708.htm.
- [3] „Medium,“ [Online]. Available: <https://medium.com/@eugeneteu/intro-to-software-engineering-architecture-model-view-controller-c29805284de6>.

TIME SERIES FORECASTING WITH RECURRENT NEURAL NETWORKS

Ondřej Bimka, Josef Kokeš

FS CVUT, ondrej.bimka@fs.cvut.cz

FS CVUT, josef.kokes@fs.cvut.cz

Abstract: The subject of the thesis “Time series forecasting with recurrent neural networks” is the description of the process of training models on typical time series data, the comparison of different models and their use in an application.

Keywords: Machine learning, Neural networks, Data science, Time-series data

1. Introduction

In this work I describe the process of training models on timeseries data predictions using recurrent neural networks (RNNs). I focus mainly on LSTM and GRU RNN cells. I selected the close prices of various cryptocurrencies as the dataset for the thesis. While the crypto market data is quite noisy and difficult to learn, I attempted to create models which would be trained to predict future prices with a degree of accuracy. [1] [2]

The text describes the entire process, from finding suitable data to implementing the best working model into an API. The final API creates predictions on real-time data.

2. Datasets

I used three main datasets. The first one was downloaded, and two others were created with two different python libraries which were wrapped into additional code. The Figure 1 shows the timeline of the three datasets.

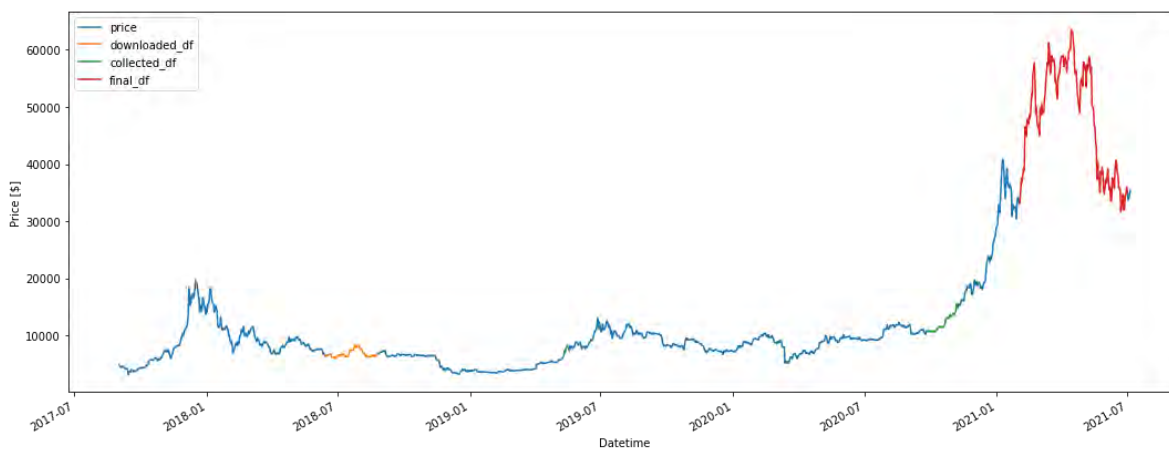


Fig. 1 BTC price with highlighted datasets.

The downloaded and collected datasets were smaller and contained 4 cryptocurrencies (BTC, LTC, ETH, BCH). Final dataset contained 8 highest correlated cryptocurrencies with BTC (BTC, OCEAN, ZRX, ATOM, BNT, ALGO, TWT, SUSHI) from a selected time period.

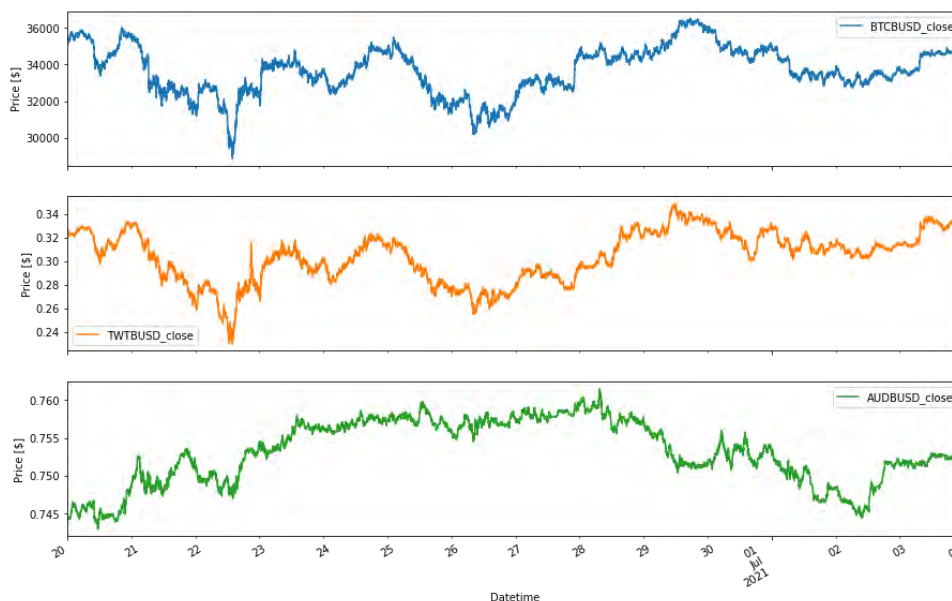


Fig. 2 Example of correlated and uncorrelated data.

3. Data preprocessing

One of the most challenging parts of the whole work was data scaling. The initial approach of data scaling, the z-score only turned out to be quite ineffective. Regardless, I still trained several models using this scaling method and got some interesting results.

Another option was using percentage change + z-score scaling, which worked better overall. The accuracy was higher, and it was easier to implement those models into an API, mostly because scalers worked for longer period on percentage change.

4. Results

The first set of models was trained on the older downloaded dataset using z-score. Predictions were made two minutes into the future using 24 minutes of past data (Fig. 2). The biggest obstacle in the first set was that input data predictions were sometimes shifted. Even in cases where the direction of the prediction was correct the scaling yielded results barely above 50%.

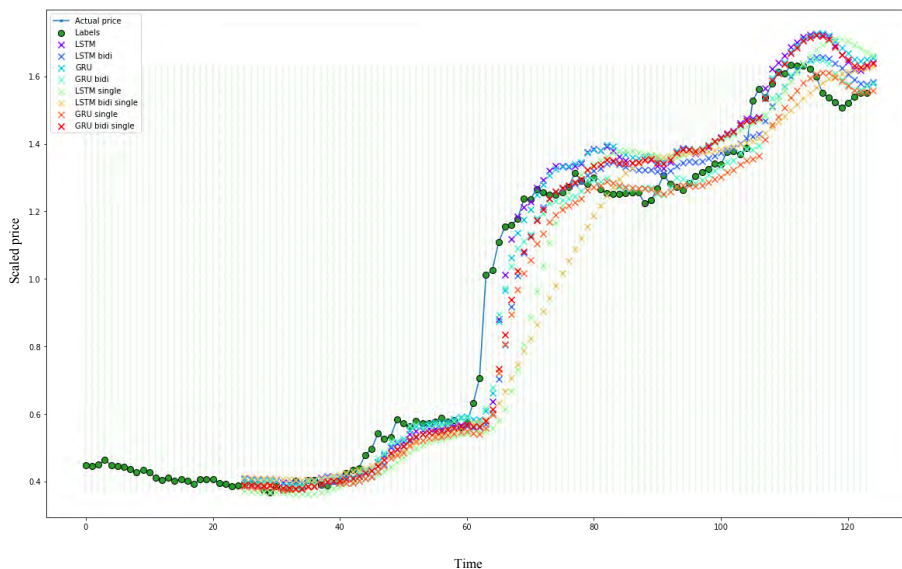


Fig. 3 Example of correlated and uncorrelated data.

In second step I used the downloaded dataset (oldest) with percentage change + z-score to predict 1 minute into the future. This approach worked quite well as on the test part of the downloaded dataset I was able to obtain models with more than 60% accuracy of direction prediction. On the more recent data (beginning of the collected dataset) the accuracy shifted to slightly below 60% accuracy.

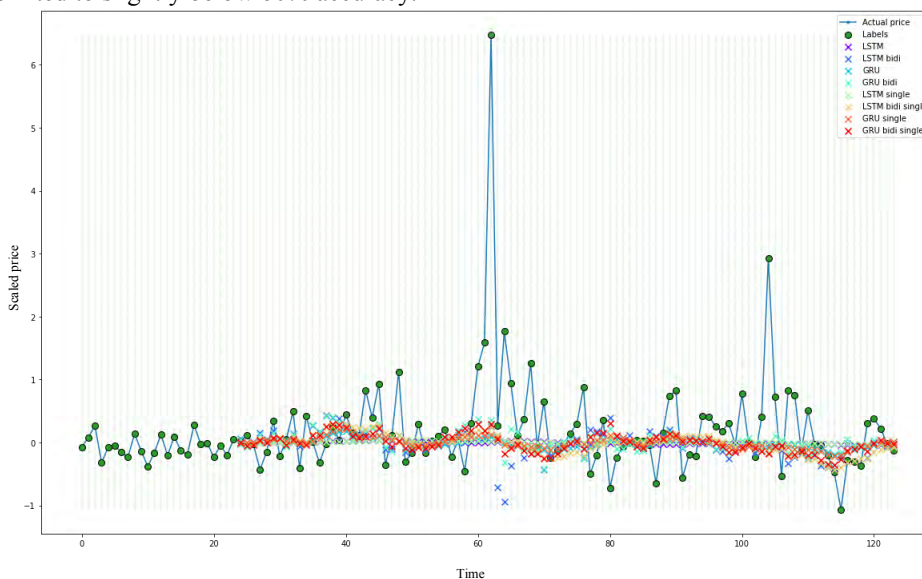


Fig. 4 Example of correlated and uncorrelated data.

The Figure 4 shows clearly that big sudden changes in values are quite unpredictable. That is not a very surprising fact, since models are very careful with more aggressive predictions. Table 1 lists the results of various architectures with 3 RNN layers, 256 cells and two dense layers first with 32 cells and last with 1 cell. The optimizer in this case was Adam with step 0.01. Adam is one of the most universal optimizers which combines the advantages of AdaGrad and RMSProp. [3] MAE is Mean Absolute Error score, and Direction tells if at least direction of prediction was correct.

Tab. 1. Performances of various models: Org refers to the test part of the training dataset. New refers to first week of collected dataset. Single means single shot models.

	MAE (org)	Direction (org) [%]	MAE (new)	Direction (new) [%]
LSTM	0.48208	60.19	0.36737	53.65
LSTM - bidirectional	0.45674	56.66	0.3564	52.83
GRU	0.47324	41.94	0.36148	44.35
GRU - bidirectional	0.46807	35.26	0.35103	40.85
LSTM (single)	0.44574	35.26	0.33903	40.7
LSTM - bidirectional (single)	0.44257	59.63	0.33837	53.12
GRU (single)	0.45174	64.74	0.33648	59.3
GRU - bidirectional (single)	0.44074	35.82	0.33681	40.85

The last set of models was trained on the final dataset which contained more data with more cryptocurrencies. When I checked the latest data (end of July 2021) markets became very dynamic, which lead to the fact that I was getting only slightly above 50% accuracy for longer time periods.

5. Predicting API

I created an API which loads selected model and makes predictions every minute based on new incoming data. Data with predictions are stored to MySQL database in predefined interval.

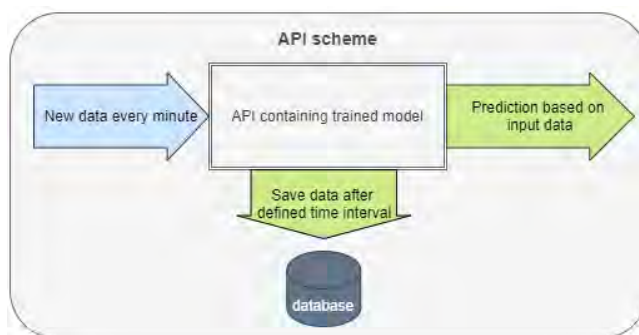


Fig. 5 API scheme.

This allows to load data later and eventually evaluate model performance, which is very useful when models predict in dynamic system such as cryptocurrency market.

6. Conclusions

It is possible to create RNN model which can predict price with limited accuracy. If we consider direction only, I was able to achieve around 60% correct percentage only from close prices on unseen data, which is higher than I expected. Unfortunately, prediction accuracy is closely related to market behavior and model performance can oscillate. For example, for the latest data I was able to get only slightly above 50% accuracy.

Acknowledgement

I wanted to thank everyone who supported me during the preparation and the work on this thesis. I also want to thank my supervisor doc. Ing. Josef Kokeš CSc. for his insides and guidance. Last but not the least, I want to thank my family for providing moral and material support during the course of my studies.

References

- [1] S. Hochreiter a J. Schmidhuber, „Long Short-Term Memory,“ *Neural Computation*, sv. 9, p. 1735–1780, 1997.
- [2] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk a Y. Bengio, *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*, 2014.
- [3] D. P. Kingma a J. Ba, *Adam: A Method for Stochastic Optimization*, 2017.

ADVANCED SUSPENSION SYSTEM WITH TWO-DEGREE-OF-FREEDOM HYBRID MASS DAMPERS

Youssef Alsoufi

*Department of Instrumentation and Control Engineering
Faculty of Mechanical Engineering
Czech Technical University in Prague , Czech Republic
Youssef.Alsoufi@fs.cvut.cz
<https://orcid.org/0000-0001-5420-9027>*

Abstrakt:

Klíčová slova:

Abstract: Historical review and Full explanation about types of dampers are presented. The benefits and drawbacks of using two degrees of freedom rather than one degree in vibration suspension are discussed in this study. Dual-Loop Controller (DLC) with two-degree-of-freedom (rotational and displacement) Hybrid Mass Damper (HMD) attached to a single degree of freedom primary system is studied. The proposed model aims to improve the performance of damping and reduce energy consumption with achieving the fail-safe behavior.

Keywords: Dynamic vibration absorber, active suspension system, hybrid vibration absorber, dual loop controller.

1 Introduction

In general, vibration problems are caused by structural resonances or significant harmonic loads. Vibration absorbers can be added to increase structural damping and thus reduce vibrations. Passive and active vibration dampers are used widely to reduce the effects of structure oscillations. As a traditional Passive device, the Tuned Mass Damper (TMD) is an auxiliary mass–spring–damper system that is correctly tuned to a target mode, or a well-known harmonic perturbation. The device is then referred to as a Dynamic Vibration Absorber (DVA) [1]. The main advantages of the passive approach are the overall set-inherent up's stability and the fact that no energy is required to dampen the oscillations. The design of a single-degree-of-freedom (SDOF) TMD to attenuate the vibration of a single mode of the main structure under various conditions is studied in [2]. Using multiple SDOF TMDs to absorb more than one mode of the main structure is discussed in [3]. The problem with DVA is that in practice, the resonant frequency of the primary structure or the disturbance frequency may change during the time. However, the traditional DVA has no ability to automatically adjust its passive parameters or absorption frequency. So, to overcome this problem, an adaptive DVA was developed, which has the ability to self-adjust its absorption frequency. The adaptive DVA can adjust its absorption frequency by modifying its passive parameters. However, the effect of adaptive DVA can only perform well on vibration control with slow time-varying disturbances frequency. Adding an actuator to control the force that the reaction mass applies to the structure allows for more efficient method of treating multiple resonances and improves the vibration absorber's efficiency in controlling structural vibration caused by random disturbance forces. The resulted proposed device is called Active Mass Damper (AMD) which can be efficient on all the controllable modes [4]. However, the drawback of traditional AMD design is that all counter forces are created by the active element and therefore, the power consumption and the size of the active actuator become very large in a case dealing with large structural vibration [5]. In [6], different types of controller designs are reported. The authors in the research proposed a new concept of active vibration absorption, "Delayed Resonator" where a controlled time delay has been used in a feedback loop. In fact, the main difference between this new generation damper and TMD is that this device is mainly developed for harmonic perturbation rejection and thus cannot be considered as a fail-safe. A robust

alternative of the delayed resonator using a spectral approach is proposed in [7] and [8].

Recently, a new definition appeared to refer to the new generation of dampers which combining the behavior of an optimal TMD and the active damping devices. These dampers are named Hybrid Mass Damper (HMD), or Hybrid Vibration Absorber (HVA). The HMD over the last decade is used widely and many control laws have been proposed. The researchers attempted to prove that using HMD will enhance the dynamic systems, decreasing the energy consumption by decreasing the control effort of the actuator, reducing the stroke of the moving mass. Optimal control is used to combine structural damping with a restricted stroke of the actuator to create a HMD in [9]. As a result, the goal of this research is to develop an advanced suspension system that will improve performance while also reducing energy consumption. The advanced suspension system is proposed to be built around a hybrid mass damper, as shown in the following paragraphs.

2 Review of passive tuned mass damper

The passive devices use the internal motion of the structure to enhance the absorption forces and dissipate the vibration energy of a specific resonance. They do not require an external power supply to operate. The control forces applied to the structure are only dependent on the structural motion in ideal passive devices. TMD absorbers are widely used in mechanical systems for vibration control. A mass is attached to a structure via a parallel spring-dashpot system in a TMD device. The mechanism is discussed in [10] where the added mass m of the TMD is usually about 10% of the main structure mass and mounted on a spring and a damper. The equations for stiffness and damping can be found in [11]. In [12], four optimum design methods for a dynamic absorber are compared when they are applied to a single-degree-of-freedom system with primary damping. The undamped natural frequency of the TMD is determined by its mass and stiffness, and it have to be tuned close to the natural frequency of the main structure. As a result, at the resonance frequency, reducing the amplitude of the main structure is achieved by properly adjusting TMD parameters damping. On the other side, because of the high sensitivity of the suppression amplitude to TMD stiffness and damping parameter variations, the challenge in designing TMD is hard tuning correctly the absorber's stiffness and damping parameters. In addition to another drawback in DVAs work which is that they are tuned to damp only one specific resonance. The problem was solved by adding an actuator that controls the force applied to the structure by the reaction mass. we mention it in section 3. In fact, parameter variations in the dynamics and natural frequency variations in the main structure have an impact on the overall performance of designed systems. Some methods designed to deal with this sensitivity. One of them is discussed in [13]. Comparison between passive and active dampers is presented in [14]. In [15], H_2 and H_∞ norms are used to optimize the system response under random and harmonic excitations with two-degree-of-freedom TMD.

3 Review of semi-active and active mass dampers

The controllers must be designed in such a way that the control effectiveness and energy consumption are balanced. Control strategies can be divided into two categories from this perspective: active and semi-active. Semi-active controllers require a small external power source to operate and rely on the structure's motion to generate control force. An external power source can be used to adjust the magnitude of the force. Since its introduction in [16], By bridging the gap between purely passive and purely active suspensions, these systems have attempted to combine the benefits of both passive and active devices. Semi-active devices, on the other hand, have fewer power requirements than active devices. Furthermore, while semi-active devices cannot inject mechanical energy into a controlled structural system, they do have properties that can be controlled to reduce the system's response to an optimal level. In [17], in order to design an optimal suspension, a Two-degree-of-freedom (TDOF) model of a semi-actively suspended vehicle is used as a starting point.

AMD, on the other hand, works by moving the auxiliary mass with an actuator connected between the structure and the auxiliary mass to generate reaction forces. The equations of AMD can be found in [18]. The AMD concept has been used in a variety of applications over the last few decades. To suppress the vibrations of structures, for example, moving a relatively small mass with a limited amplitude is required. This explains why AMDs have a limited ability to absorb large earthquake excitations. However, the goal of structural vibration energy absorption under small earthquake excitations or strong wind is met. As a result of active control, the auxiliary system's response motion is usually increased. Therefore, not only is it desirable to improve control efficiency, but also to limit auxiliary mass motion as much as possible. The allowable amplitude of the AMD is generally determined by the size of the AMD or the installation space, with strict limitations. The limitations of the auxiliary mass's amplitude are one of the main reasons for AMDs' poor performance. In [9], a control law for AMD that effectively suppress the vibrations of a SDOF structural system under the amplitude constraint of the auxiliary mass is studied.

The research [19] discussed the advantages of AMD compared with PSS. Pneumatic, linear motor, hydraulic actuators, and other force actuators can provide the required forces in [20]. A four-degree-of-freedom half-car model with active suspension is studied in [21]. Recently with self-driving vehicles, the suspension of the vertical vibration amplitude of the cameras which mounted on the vehicles is needed to reduce the unwanted motion effects.

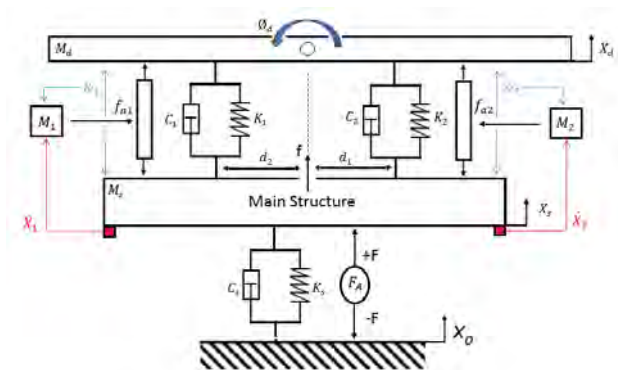


Fig. 1: Advanced suspension system with two degree of freedom hybrid mass damper

4 Review of hybrid mass damper

After decades of research, a new class of dampers has emerged that attempts to combine multiple objectives and features at the same time. These devices are gathered under the common name of Hybrid Mass Dampers (HMDs), or Hybrid Vibration Absorbers (HVAs) which combine passive and active vibration control. Combining passive and active elements the system is fail-safe that the damper will behave as a passive device even when the feedback control is turned off. The goal of using HMD may differ from one to the other. For example, in [22], H_∞ optimal design of HMD is used for the minimization of the resonant vibration amplitude of a SDOF vibrating structure. In [23], a pole placement technique is proposed to ensure performance and stability. And a special pole placement controller is designed such that all vibration modes of the flexible structures become critically damped in [24]. In [25], a dual loop approach is preferred to increase the stability margins. Improving the performance and stability of hybrid mass dampers by creating a hyperstable controller is studied in [26].

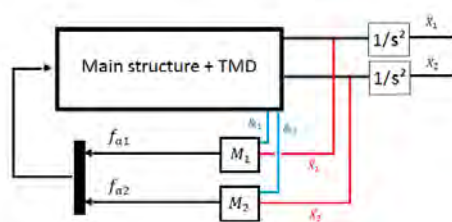


Fig. 2: Controller loops.

5 proposed model

More than one mode of vibration of an absorber body relative to a primary system can be tuned to suppress single-mode vibration of a primary system. Therefore, mitigation of the response under harmonic excitations is proposed in the present model. Fig. 1 represents a model with two-degree of freedom HMD attached to a single degree of freedom primary system. Vertical displacement movement and rotational movement around the center of the auxiliary mass as a basic design is proposed. The results show that an optimal two-degree of freedom TMD outperforms a traditional single-degree of freedom TMD with optimal mass distribution, even when the absorber's rotary inertia approaches zero. We will use this proof as a starting point for converting TMDs to HMDs later. It is well known that when we use an HMD to attenuate the vibration; the passive part has a positive impact on reducing the amplitudes of the vibrations i.e. damping the oscillations occurs without the need for any external power. The frequency band where the absorber effectively suppresses vibrations, on the other hand, is relatively

narrow, being centered at the absorber's natural frequency. Because the physical absorber is never perfect, i.e. it has nonzero damping, the damper cannot absorb the vibration completely even if the vibration frequency is identical to the absorber's natural frequency. Solving this problem is presented in section 2. The proposed model aims to accomplish three main goals: (1) improve the performance of the suspension system, (2) reduce energy consumption, and (3) ensure fail-safe behavior. The passive parts are TMDs optimally tuned using Den Hartog's law in [1], and the active control forces (f_{a1}) and (f_{a2}) are introduced between the two masses. The general concept of dual-loop controller is introduced in [4] and is used precisely as parallel dual loops and act on the same transducer in [27] which is the closest to our model with the exception that in the previous research a dual loop controller (DLC) with a single degree of freedom was used. The proposed controller employs two inputs: (i) the relative displacement between the inertial mass and the main structure, and (ii) the absolute velocity of the main structure. Subsequently, dual loops on each side of the damper are used. One to detune the HMD (negative stiffness feedback), and the other to damp the main structure (direct velocity feedback). Fig 2 illustrates that since the relative velocity or absolute acceleration will be measured, so integration has to be added to the control laws M1 and M2. The damper's rotational movement allows it to absorb not only vertical but also angle disturbances in some cases. Also, as long as we use an active impact in our design, we have to keep in mind the necessity of restricting the amount of energy required for absorption operation. We mentioned it in section 1. Indeed, increasing the frequency bandwidth where the absorber suppresses the vibrations has been discussed in many previous studies, such as in [8]. So, in this article, we will try to demonstrate the superiority of TDOF over SDOF by comparing them first without any active parts and then improving the concept by adding a single loop PD controller to the TDOF as an active part. Let's start with our basic design shown in Fig. 1. The main structure has a natural frequency $w_s = \sqrt{k_s/m_s}$ and damping ratio $\zeta_s = c_s/2\sqrt{k_s m_s}$ and is subjected to a base excitation x_0 , an external disturbance force f . We will use a harmonic disturbance as an external force in this study because it represents a more realistic situation in a TMD's field of application (hybrid or not) x_d is translation and θ_d is rotation. m_d is the mass of the damper and I_d is the rotational inertia about its center of mass $I_d = m_d \rho^2$ where ρ is the radius of gyration. The absorber is connected to the primary system at distances d_1 and d_2 from its center of mass via springs and dashpots. Let's consider first the case where $d_1 = d_2 = d$. We will replace the spring and the damping device by control-force vector $[u_1, u_2]^T$ where

$$\begin{aligned} u_1 &= k_1(x_1 - x_s) + c_1(x'_1 - x'_s) \\ u_2 &= k_2(x_2 - x_s) + c_2(x'_2 - x'_s) \end{aligned} \quad (1)$$

x_1, x_2 are the displacements of the damper in the direction of x_s at the connection locations.

$$\begin{aligned} m_d x''_D &= -u_1 - u_2 \\ I_d \theta'' &= u_1 d - u_2 d \end{aligned} \quad (2)$$

$$m_s x''_s = u_1 + u_2 - k_s x_s - c_s x'_s + k_s x_0 + c_s x'_0 + f \quad (3)$$

According to the proposed model, we note that:

$$x_d = \frac{x_1 + x_2}{2}, \theta_d = \frac{x_2 - x_1}{2d}, I_d = m_d \rho^2 \quad (4)$$

$$\begin{aligned} \frac{m_d}{2} (x''_1 + x''_2) &= -u_1 - u_2 \\ \frac{m_d}{2} (x''_2 - x''_1) &= u_1 \frac{d^2}{\rho^2} - u_2 \frac{d^2}{\rho^2} \end{aligned} \quad (5)$$

All the previous equations are covered in matrix form:

$$\begin{aligned} \begin{bmatrix} m_d/2 & m_d/2 & 0 \\ -m_d/2 & m_d/2 & 0 \\ 0 & 0 & m_s \end{bmatrix} \begin{bmatrix} x''_1 \\ x''_2 \\ x''_s \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & c_s \end{bmatrix} \begin{bmatrix} x'_1 \\ x'_2 \\ x'_s \end{bmatrix} + \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & k_s \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_s \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \\ c_s \end{bmatrix} x'_0 + \begin{bmatrix} 0 \\ 0 \\ k_s \end{bmatrix} x_0 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} f \end{aligned}$$

$$+ \begin{bmatrix} -1 & -1 \\ \frac{d^2}{\rho^2} & -\frac{d^2}{\rho^2} \\ 1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (6)$$

Or

$$Mq'' + Cq' + Kq = B_{00}X_0 + B_{01}X'_0 + B_f f + B_u u \quad (7)$$

$$\text{where } q = [x_1, x_2, x_s]^T$$

If we want to use the model as a HMD then,

$$\begin{bmatrix} 0 & -1 & -1 \\ 0 & \frac{d^2}{\rho^2} & -\frac{d^2}{\rho^2} \\ 1 & 1 & 1 \end{bmatrix} f \text{ where } f = [f_s, f_{a1}, f_{a2}]^T$$

let us assume that all the initial values equal zero.

$$x = \begin{bmatrix} q \\ q' \end{bmatrix} \quad (8)$$

$$x' = Ax + B_1 f + B_2 u; u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (9)$$

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \quad (10)$$

$$B_1 = \begin{bmatrix} 0 \\ M^{-1}B_f \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ M^{-1}B_u \end{bmatrix} \quad (11)$$

The cost output can be interpreted as the absolute or relative displacement, velocity, or acceleration of the primary system, u is a matrix given by (1) as a state feedback matrix P multiplied by the measurement output y. So:

$$u = \begin{bmatrix} -k_1 & k_1 & c_1 & 0 & 0 \\ -k_2 & 0 & 0 & k_2 & c_2 \end{bmatrix} y = Py \quad (12)$$

where

$$y = [x_s, x_1 - x_s, x'_1 - x'_s, x_2 - x_s, x'_2 - x'_s]^T$$

To make the comparison easier, we divided the frequency band into many characteristic frequencies after evenly distributing the stiffness and damping device values of the SDOF into two spring-dashpot couples for the TDOF concept. TDOF TMD without any active parts had a better overall performance in the critical frequencies like resonance frequency of the main structure $w = 35.7$ rad/sec where TDOF shows more efficiency than SDOF, see Fig. 4. Let's start our study when $w \ll w_n = 10$ rad/s with arbitrary constant values for stiffness and damping device. The bode plot in this part of frequencies shows constant values for the amplitude and phase in both models Fig.3(b).

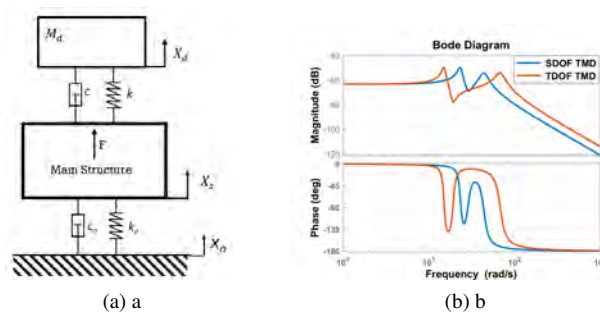


Fig. 3: SDOF TMD

For critical frequencies $w_1 = w_{SDOF} = 23.3$ rad/sec and $w_2 = w_{TDOF} = 14.9$ rad/sec, we can see that the magnitude of TDOF TMD at resonance frequency w_1 is around 0.3×10^{-3} (m) with phase (-14.9) deg Fig. 5 (b). On the other side, SDOF TMD reaches around 1×10^{-3} for a peak amplitude at resonance frequency w_2 and phase

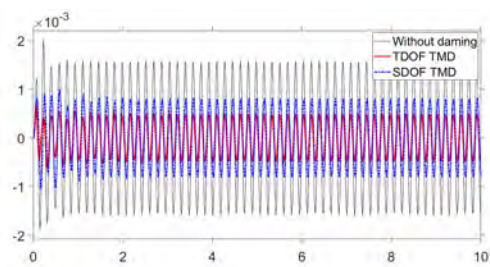


Fig. 4: Absolute displacement X_S at natural frequency of main structure.

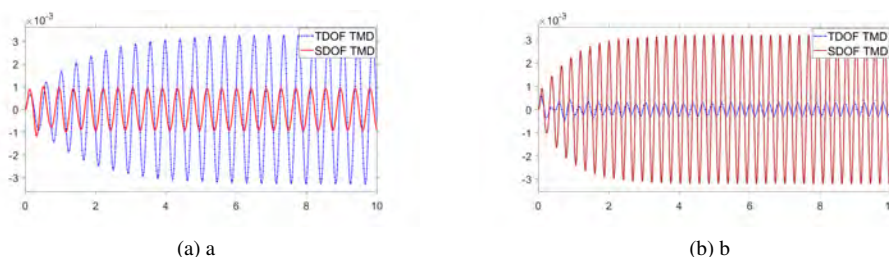


Fig. 5: Absolute displacement x_s/x_0 at critical frequencies (a) 14.9 rad/sec , (b) 23.3 rad/s.

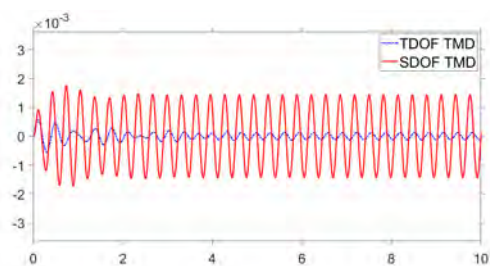


Fig. 6: Absolute displacement X_S at frequency $w=19.5$ rad/sec

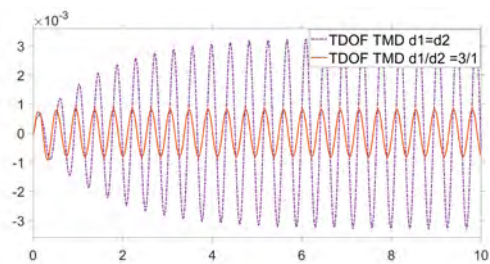


Fig. 7: Absolute displacement X_S at frequency $w_n=14.9$ rad/sec, $d_1 = d_2$ (dashed) and $d_1/d_2 = 3/1$ (line)

(-3.85) deg Fig. 5 (a). Also at frequency $w=19.5$ rad/s it is clear that TDOF TMD has a higher performance than SDOF TMD Fig. 6 where $Amp_{(SDOF)} = 1.5 \times 10^{-3}$ (m) and $Amp_{(TDOF)} = 0.25 \times 10^{-3}$ (m).

The second case is changing the dimensions d_1, d_2 when $d_1 \neq d_2$. As is obvious in Eq. 6, the performance of the TDOF TMD system depends on the ratio of the absorber's radius of gyration ρ to the distance d from the mount points to the center of mass of the absorber. So, according to Eq. (6) and Fig. 1 when $d=0$, that corresponds to the optimal SDOF TMD. Even if the ratio 3/1 could be not an optimal rate, let's use it for this study as a ratio d_1/d_2 and substitute it in the equation while keeping ρ constant. The results show that the performance of the damper is increased at the resonance frequency 14.9 rad/sec compared with the same model when $d_1 = d_2$ Fig. 7 with approximately (-80) deg as a phase-shift.

Fig. 8(b) illustrates the relative displacement spectrum (X_1-X_s) of TDOF TMD at frequency $w=66$ rad/sec where the amplitude for both tested systems is the same Fig. 8(a). One can see in the figure that the superior performance of changing the dimensions of d_1, d_2 is obtained at the cost of a much larger relative displacement of

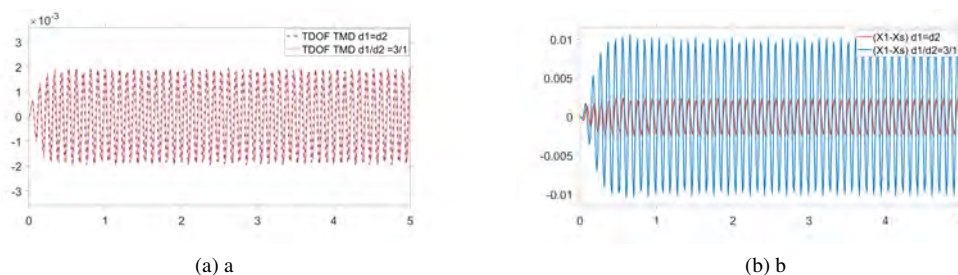


Fig. 8: (a).Absolute displacement x_s/x_0 when $d1=d2$ (dashed) and when $d1/d2=3/1$ (line). (b).Relative displacement $(X1-Xs)$ when $d1=d2$ (red) and $d1/d2=3/1$ (blue) $w=66$ rad/s

the moving mass (m_d).

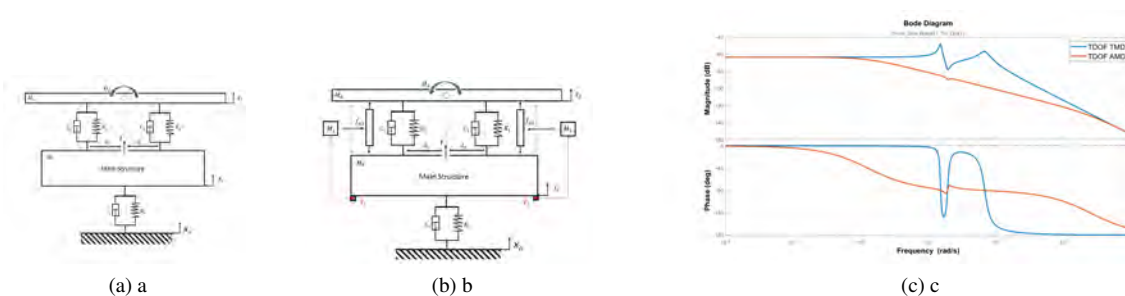


Fig. 9: (a) TDOF TMD (b) TDOF AMD (c) Bode plot

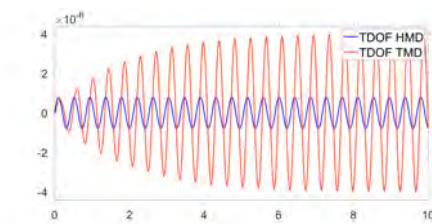


Fig. 10: Absolute displacement X_s at frequency $w=14.9$ rad/sec TDOF-TMD(red), TDOF-HMD (blue)

In fact, achieving optimal damper performance without combining active and passive parts to create an HMD with fail-safe behavior is impossible. PD controller as a single loop controller is proposed for this study. So the equations: $M_1 = g_{c1}s + g_{k1}$ and $M_2 = g_{c2}s + g_{k2}$ where $g_{c1} > 0$, $g_{c2} > 0$. In the bode plot Fig. 9, we can see the increase in gain and phase, as well as the response of the damper at the peak amplitude of TDOF TMD in the time domain Fig. 10 where the residual motion of m_s is more than 5 times smaller with the HMD than with the TMD with amplitude 0.8×10^{-6} .

6 conclusion

A historical overview of the most commonly created and developed types of dampers is presented. The superiority of the two-degree of freedom (displacement and rotational movements) dampers over a single degree with only displacement flexibility is demonstrated in this article by comparing the performance of TDOF TMD with SDOF TMD for a very wide frequency range. In the end, a hybrid mass damper with dual loop controllers is used to improve the damper's performance in both the time and frequency domains. The industrial application will need to provide derivative control filtering, as well as AMD performance, which will be thoroughly tested on various industrial applications in future research, including comparative performance analysis and energy consumption reduction.

Tab. 1: Identified parameters of the models:

Parameters	Values	Units	Parameters	Values	Units
ms	1.1570	Kg	cs	4.4317	N/m/s
md	0.5200	Kg	k	423.9525	N/m
ks	1482.2504	N/m	c	2.4198	N/m/s

Tab. 2: Nomenclature

Abbreviation	Definition
AMD	Active Mass Damper
DLC	Dual-Loop-Controller
DVA	Dynamic Vibration Absorber
HMD	Hybrid Mass Damper
HVA	Hybrid Vibration Absorber
PSS	Passive Suspension System
SDOF	Single-Degree-Of-Freedom
TDOF	Two-Degree-Of-Freedom
TMD	Tuned Mass Damper

Acknowledgement

The author has been supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS20/159/OHK2/3T/12.

References

- [1] JP DenHartog. Mechanical vibrations, (1956), 373.
- [2] GB Warburton. Optimum absorber parameters for various combinations of response and excitation parameters. *Earthquake Engineering & Structural Dynamics*, 10(3):381–401, 1982.
- [3] L Kitis, BP Wang, and WD Pilkey. Vibration reduction over a frequency range. *Journal of Sound and Vibration*, 89(4):559–569, 1983.
- [4] André Preumont and Kazuto Seto. *Active control of structures*. Wiley Online Library, 2008.
- [5] Andreas H Vonflotow, Andrew Beard, and Don Bailey. Adaptive tuned vibration absorbers: tuning laws, tracking agility, sizing, and physical implementations. *nce*, pages 437–454, 1994.
- [6] N Olgac and BT Holm-Hansen. A novel active vibration absorption technique: delayed resonator. *Journal of Sound and Vibration*, 176(1):93–104, 1994.
- [7] Dan Pilbauer, Tomáš Vyhřídál, and Wim Michiels. Optimized design of robust resonator with distributed time-delay. *Journal of Sound and Vibration*, 443:576–590, 2019.
- [8] Matěj Kuře, Tomáš Vyhřídál, Wim Michiels, and Islam Boussaada. Spectral design of robust delayed resonator by double-root assignment. *IFAC-PapersOnLine*, 51(14):72–77, 2018.
- [9] Kazunobu Yoshida and Itaru Matsumoto. Vibration suppression control for a multi-degree-of-freedom structural system using an amd with restricted stroke. In *2009 International Conference on Networking, Sensing and Control*, pages 912–917. IEEE, 2009.
- [10] MA Acar and C Yilmaz. Design of an adaptive–passive dynamic vibration absorber composed of a string–mass system equipped with negative stiffness tension adjusting mechanism. *Journal of Sound and Vibration*, 332(2):231–245, 2013.
- [11] Jacob Pieter Den Hartog. *Mechanical vibrations*. Courier Corporation, 1985.
- [12] YZ Wang and SH Cheng. The optimal design of dynamic absorber in the time domain and the frequency domain. *Applied Acoustics*, 28(1):67–78, 1989.

- [13] Hui Zhang, Yang Shi, and Aryan Saadat Mehr. Robust non-fragile dynamic vibration absorbers with uncertain factors. *Journal of Sound and Vibration*, 330(4):559–566, 2011.
- [14] Sang-Myeong Kim, Semyung Wang, and Michael J Brennan. Dynamic analysis and optimal design of a passive and an active piezo-electrical dynamic vibration absorber. *Journal of sound and vibration*, 330(4):603–614, 2011.
- [15] Lei Zuo and Samir A Nayfeh. The two-degree-of-freedom tuned-mass damper for suppression of single-mode vibration under random and harmonic excitation. 2006.
- [16] Dean Karnopp, Michael J Crosby, and RA Harwood. Vibration control using semi-active force generators. 1974.
- [17] D Hrovat, DL Margolis, and M Hubbard. An approach toward the optimal semi-active suspension. 1988.
- [18] Isao Nishimura, Takuji Kobori, Mitsuo Sakamoto, Norihide Koshika, Katsuyasu Sasaki, and Satoshi Ohru. Active tuned mass damper. *Smart Materials and Structures*, 1(4):306, 1992.
- [19] TPJ Van der Sande, BLJ Gysen, IJM Besselink, JJH Paulides, EA Lomonova, and H Nijmeijer. Robust control of an electromagnetic active suspension system: Simulations and measurements. *Mechatronics*, 23(2):204–212, 2013.
- [20] Jiangtao Cao, Honghai Liu, Ping Li, and David J Brown. State of the art in vehicle active suspension adaptive control systems based on intelligent methodologies. *IEEE transactions on intelligent transportation systems*, 9(3):392–405, 2008.
- [21] Panshuo Li, James Lam, and Kie Chung Cheung. Multi-objective control for active vehicle suspension with wheelbase preview. *Journal of Sound and Vibration*, 333(21):5269–5282, 2014.
- [22] YL Cheung, WO Wong, and L Cheng. Design optimization of a damped hybrid vibration absorber. *Journal of sound and vibration*, 331(4):750–766, 2012.
- [23] MH Tso, Jing Yuan, and WO Wong. Design and experimental study of a hybrid vibration absorber for global vibration control. *Engineering structures*, 56:1058–1069, 2013.
- [24] MH Tso, Jing Yuan, and WO Wong. Suppression of random vibration in flexible structures using a hybrid vibration absorber. *Journal of sound and vibration*, 331(5):974–986, 2012.
- [25] Peter Hagedorn and Gottfried Spelsberg-Korspeter. *Active and passive vibration control of structures*. Springer, 2014.
- [26] Christophe Collette and Simon Chesne. Robust hybrid mass damper. *Journal of sound and vibration*, 375:19–27, 2016.
- [27] Simon Chesne, G Inquieté, P Cranga, F Legrand, and B Petitjean. Innovative hybrid mass damper for dual-loop controller. *Mechanical Systems and Signal Processing*, 115:514–523, 2019.

ANALYSIS OF LOCOMOTIVE MOTOR TORQUE IMPACT AT CHANGEABLE DYNAMIC OPERATION CONDITIONS

*A. Atiyah*¹

¹ Faculty of Mechanical Engineering CTU in Prague, ahmad.atiyah@fs.cvut.cz

Abstract: In today's locomotives, various undesirable dynamic phenomena affect the torque efficiency transmitted to the wheels. The impact of these dynamics ranges from minor effects to major effects that can involve the locomotive as a whole, as in the case of a slip. In this paper, we analysed the behavior of the torque transmitted to wheels during variable levels of adhesion. The results showed a comparison between the motor torque value at different levels of adhesion and its value under normal operating conditions when the adhesion value is at its best.

Keywords: Wheels; Torque; Adhesion; Slip; Asynchronous Motor; Locomotive; Dynamic

1 Introduction

Modern locomotives play an important role in transportation compared with other means of transport, as they are characterized by strength, flexibility, reliability, high performance, and low-cost operation.

These features come with many challenges related to the external and internal dynamics which has a high impact on the traction system. One of these challenges is what is known as the adhesion phenomena, which represents the friction force between the wheels and the rails, and is influenced by the external ambient conditions of the train and the rail condition (wet or dry, etc.).

Another dynamic phenomenon that affects the safe operation of the train is what is known as slip, which occurs when the wheels circumference velocity exceeds the linear speed of the train itself as a result of applying a traction force to the wheels greater than the friction force of the rail itself.

These undesirable dynamic conditions, has a direct impact on the electrical and mechanical systems of the train, in particular the torque of the motor transmitted to the wheels, as the wheels are linked to locomotive traction motor via the gearbox, and when slip occurs, the value of the torque drops significantly due to the decrease in adhesion.

2 Slip and Adhesion

In rail transport, Adhesion is a phenomenon without which it cannot be realized. There is therefore a great interest in its use to the maximum extent possible to increase the efficiency of rail transport. The geometric-kinematic context of adhesion is shown in Fig. 1.

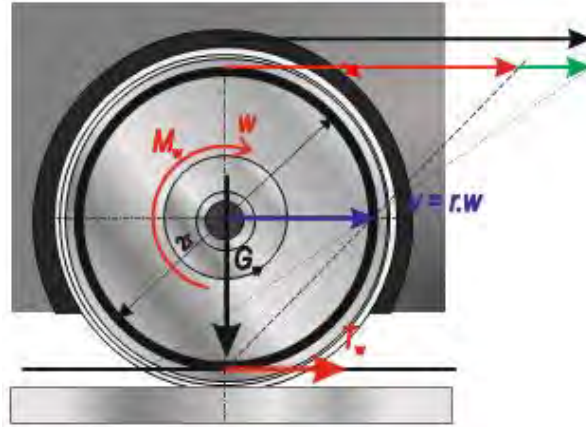


Fig. 1. Context of adhesion between wheel and rail

In the case of an ideal rolling motion, a wheel of diameter $d = 2r$ rotating at an angular velocity w moves to a distance equal to the circumference of the wheel $2\pi r$ in a time equal to $T = 2\pi/w$, i.e., that it performs a horizontal motion with speed $v = 2\pi r / (2\pi/w) = wr$.

During train operation, when a decrease in the adhesion between the wheels and the rail occurs, the speed of the wheels increases by slip ratio $(r.w-v)/v$ [2].

The value of the adhesion coefficient depends not only on the wheel slip but it is highly variable according to the conditions prevailing at the point of contact of the wheel with the rail, such as humidity, temperature, pollution by oil, dust, leaves, etc. Its direct on-line evaluation in the normal operation is very difficult.

According to Popovici table shown in Fig.2, In the region of low values of the slip, the adhesion coefficient increases with the increasing value of the slip. Due to the unchanging wheel weight load Gw it means increase of the wheel tangential force, i.e., more effective traction up to certain achievable maximum. It is clear that in an optimal rail vehicle ride should be utilized this maximum, but there must be a protection against an overrun into the area where the increase of the slip leads to a reduction of the adhesion coefficient. A ride under such conditions is unwanted, delivered torque is not used in acceleration wheel turns quicker then it corresponds vehicle movement, adhesions go into friction with all consequences such as wearing and mechanical stress.

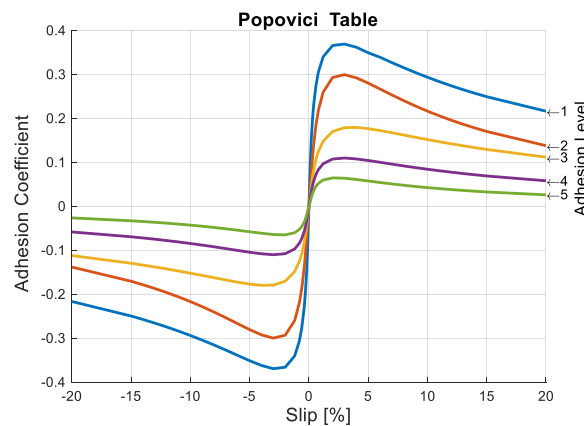


Fig. 2. Adhesion coefficient-Slip characteristics from Popovici table

In order to avoid slip occurrence, usually, locomotives drive unit is equipped with so-called Slip controllers, where it takes the responsibility to regulate (reduce) the asynchronous motor torque in such a way that the slip will be suppressed [1]. In our Simulink model, such a slip controller has been not yet developed, and alternatively, attention was paid to optimize traction motor torque during the loss of adhesion and slip occurrence by analyzing the torque response at changeable adhesion levels and different initial operation conditions, in addition to controlling the motor torque using the vector control method.

3 Asynchronous Motor and Wheelset

The main task of the locomotive wheel-set drive system is ensuring delivering the adequate torque value from the asynchronous motor control system. In case of the absence of sensors that measure the wheel speed, the task of controlling the entire drive system falls to the motor control system, which means that the wheelset velocity is determined by the asynchronous motor velocity which is measured using a sensor mounted at the motor shaft.

It is worth here explaining the mutual relationship between the motor and the wheelset drive system. Thus, when the adhesion value decreases, the speed of the wheels increases by slip ratio. This increase in wheel speed is linked with the asynchronous motor rotational speed through gears. and thus, the motor angular velocity has a direct influence on the internal dynamics and variables behaviors of the asynchronous motor via the so-called motor slip.

The motor angular velocity is used as an input signal to the motor control system. The increase of the wheels speed is matched by a sudden decrease in the value of the torque transmitted to the wheels, and from simulation results, we noticed that the decrease in adhesion value has an impact on both motor rotational velocity and the motor voltage which results in an increase in the voltage value above its nominal value. This also results in an increase in the stator frequency [1].

The significance of this effect becomes apparent if we know that the magnetic flux linkage which has a direct impact on controlling torque must be maintained at a constant value, and this can be achieved by maintaining the motor voltage within the nominal limits. The maximum torque of the asynchronous motor is directly determined by the maximum value of the stator current, as shown in the relation (6). We have observed via simulation experiments, that when the value of adhesion decreases, the decrease in torque is accompanied by a slight decrease in the value of the stator current in order to keep the magnetic flux constant. Since the current is a measured quantity, then, it's good to consider the possibility of early detection of wheel slip by generating a suitable delay for the Anti-slip controller to respond in the case of a continuous decrease in the current value.

In order to achieve the desired dynamics of the locomotive movement, the concern not only includes generating sufficient motor torque but also extends to involve the enhancement of the torque control quality using different control methods. If an inadequate tractive torque is detected, it may lead to an inadequate wheel slip and a subsequent limiting vehicle dynamic.

Studying the different asynchronous motor control methods used in the locomotive drive unit helps in determining the torque behavior of the ASM more effectively, and can also lead to an increase of the locomotive's response to unwanted dynamic changes affect it, and it may even be possible to predict these dynamic changes before they occur.

3.1 Asynchronous Motor Model

The asynchronous motor is considered the main component of a railway locomotive, it is responsible for delivering the adequate torque value to the wheelset in order to let the train move on the rail. In this paper, the ASM model used is controlled only via the FOC method without VSI.

To obtain an ASM model, Clark and park transformation has been used for decades [4][5] to convert the motor equations from three-axis reference coordinates "abc" into a two-axis " $\alpha\beta$ " stationary reference frame or rotating frame "dq0" [6]. There are multiple reasons behind using those transformation methods, one of these reasons is the presence of time-varying inductances in the motor voltage equations [5]. Clark transformation matrix T_c can be expressed by the following equation:

$$T_c = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \quad (1)$$

and thus, we can easily obtain the $i_{s\alpha}, i_{s\beta}$ currents which are of a sinusoidal nature. To create an efficient control method for ASM, the converted motor quantities using Clark transformation should be of continuous nature, and that can be achieved by means of the Clark to Park transformation matrix represented in the Fig.3 using the following equation:

$$\begin{bmatrix} i_{s,dq} \end{bmatrix} = T_p \cdot \begin{bmatrix} i_{s,\alpha\beta} \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} = \begin{bmatrix} \cos(\gamma_s) & \sin(\gamma_s) \\ -\sin(\gamma_s) & \cos(\gamma_s) \end{bmatrix} \cdot \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix}$$

Where is i_{sd}, i_{sq} are the stator currents in rotating reference frame rated in [A].

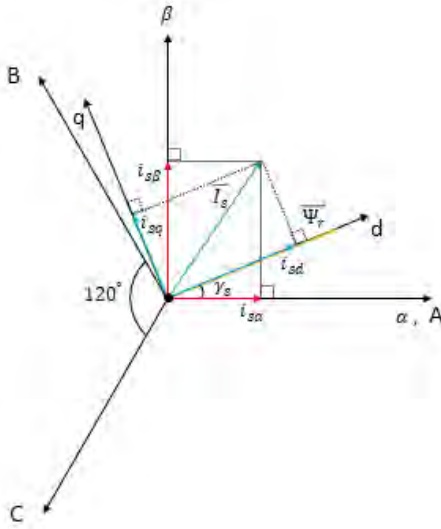


Fig. 3. Stator Current space vector in stationary and rotating reference frames

The stator and rotor voltages in stationary frame can be written as follows:

$$\begin{aligned}
 u_{s\alpha}(t) &= r_s i_{s\alpha}(t) + \frac{d}{dt} \psi_{s\alpha}(t) \\
 u_{s\beta}(t) &= r_s i_{s\beta}(t) + \frac{d}{dt} \psi_{s\beta}(t) \\
 0 &= u_{r\alpha}(t) = r_r i_{r\alpha}(t) + \frac{d}{dt} \psi_{r\alpha}(t) + \omega_m p_p \cdot \psi_{r\beta}(t) \\
 0 &= u_{r\beta}(t) = r_r i_{r\beta}(t) + \frac{d}{dt} \psi_{r\beta}(t) - \omega_m p_p \cdot \psi_{r\alpha}(t)
 \end{aligned} \tag{3}$$

The equation which gives the electromechanical torque of an asynchronous motor can be expressed as follows:

$$M(t) = \frac{3}{2} p_p \cdot (\psi_{r\alpha}(t) \cdot i_{s\beta}(t) - \psi_{r\beta}(t) \cdot i_{s\alpha}(t)) \tag{4}$$

Where $M(t)$ represent the electromechanical torque rated in [N.m].

3.2 Field Oriented Control

So far, a variety of vector modelling methods like FOC and DTC [3][7][8][9] has been used in practice to control the ASM.

For long time, FOC considered as the most common method to control AC motor in practice. FOC control method is represented using what is known as direct quadratic coordinates ($dq\theta$ coordinates), and this coordinate system rotates synchronously with rotor flux vector. In FOC, the torque is proportional to the cross product of stator current and flux vector. By decoupling the torque and flux vectors form each other, the control of both torque and flux is quite similar to DC motor control. In our Simulink model, we developed an ASM control based on FOC. The torque vector is aligned with q-axis, and the flux vector is aligned with d-axis, which result in:

$$\psi_r = \begin{bmatrix} \psi_{rd} \\ \psi_{rq} \end{bmatrix} = \begin{bmatrix} |\psi_r| \\ 0 \end{bmatrix} \tag{5}$$

And hence $\psi_{rq} = 0$, the motor torque can be rewritten as:

$$M(t) = \frac{3}{2} p_p \cdot \psi_{rd}(t) \cdot i_{sq}(t) = \frac{3}{2} p_p \cdot |\psi_r(t)| \cdot i_{sq}(t) \tag{6}$$

We can notice from equation (6), that we can control the motor torque with stator current i_{sq} by keeping the rotor flux at a constant value [10]. Fig.3 shows the FOC block diagram of ASM.

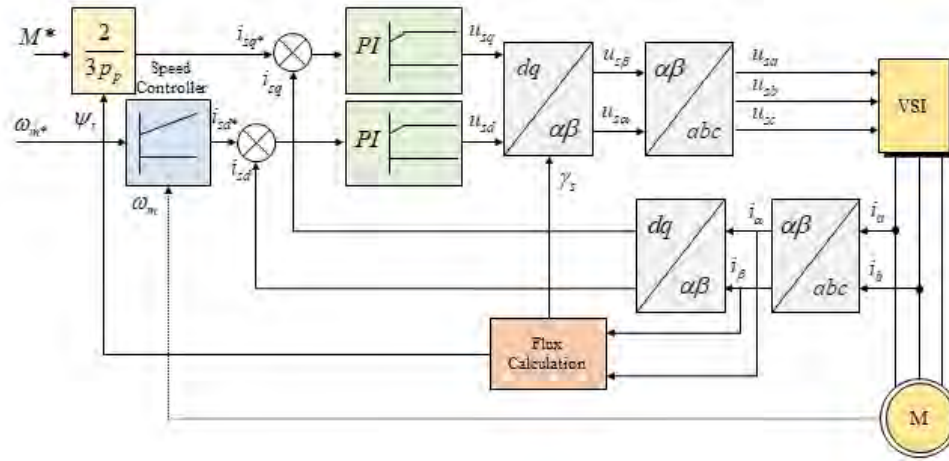


Fig. 4. Block diagram of asynchronous motor

From Fig.4, two PI controllers are used to regulate the dq stator currents. The PI output voltages are transformed to stationary voltages using inverse Park to Clark transformation matrix:

$$\begin{aligned} \begin{bmatrix} u_{s,dq}(t) \end{bmatrix} &= T_p \cdot \begin{bmatrix} u_{s,\alpha\beta}(t) \end{bmatrix} \\ \begin{bmatrix} u_{s\alpha}(t) \\ u_{s\beta}(t) \end{bmatrix} &= \begin{bmatrix} \cos(\gamma_s) & -\sin(\gamma_s) \\ \sin(\gamma_s) & \cos(\gamma_s) \end{bmatrix} \cdot \begin{bmatrix} u_{sd}(t) \\ u_{sq}(t) \end{bmatrix} \end{aligned} \quad (7)$$

We can convert the resulting stationary U_{sd}, U_{sq} voltages to 3-phase voltages by means of Clark inverse matrix.

4 Simulation Results

So far, our Simulink model has satisfied the mechanical part, while the electrical part is still subject to further development in order to optimize the motor torque and enhance the countability of the locomotive drive.

The asynchronous motor model used is modelled according to the parameters of a real ML4550-K / 6 motor which has a maximum torque of 10000N.m[11]. A numerical simulation has been carried out using Matlab/Simulink to show how motor torque transmitted to the wheels at different operation conditions and under undesired dynamics changes.

Table 1, shows the parameter values for the simulation experiments. the torque value was set to a range of 4200-8400 N.m at a normal operating velocity of 100 km.h⁻¹ and at a low operating velocity of 10 km.h⁻¹.

All simulation experiments were performed in two stages, in the first stage, no variable adhesion levels are taken into account. While in the second stage, the values of adhesion levels were changed from 1 - 5, which are shown in Fig.2.

The initial train velocity is of Step signal, while the torque setpoint is a ramp signal.

Tab. 1. Simulation parameters of the traction motor

Simulation Test	Longitudinal Velocity km/h	Torque N.m	Adhesion Level	Adhesoin Time
I	100	4200	1	4.5-9.5-14.5-17.5
II	100	4200	1-5	
III	100	8400	1	
IV	100	8400	1-5	
V	10	4200	1	
VI	10	4200	1-5	
VII	10	8400	1	
VIII	10	8400	1-5	

From Fig.5(a), we notice that when a 4800 Nm torque setpoint is applied at a speed of 100 km.h⁻¹ with no change in the adhesion level, the motor torque tracks the desired torque setpoint with a negligible steady-state error, and the response speed of the motor torque relative to the desired torque setpoint is fast with a low torque ripples amplitudes transmitted to the wheels. When changing the adhesion levels according to the simulation times shown in Table 1, we noticed a significant decrease in the motor torque ranges between 56 - 159 Nm as shown in Fig.5(b).

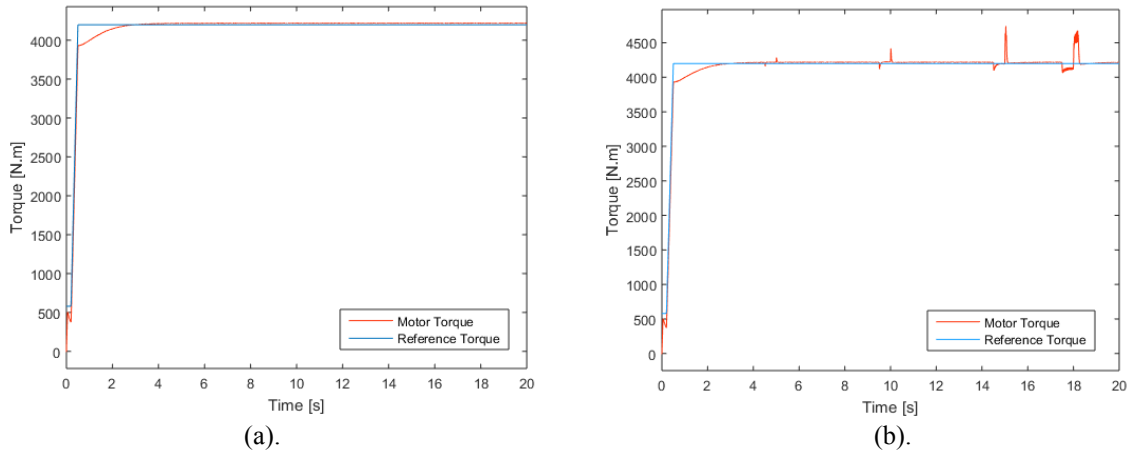


Fig. 5. a) Motor torque response at 4200N.m and 100km.h⁻¹ with no adhesion changes.
 b) Motor torque response at 4200N.m and 100km.h⁻¹ with changeable adhesion levels

By setting the desired torque value at 8400 N.m which represents the nominal value of the locomotive motor torque at an operating speed of 100 km.h⁻¹ and without changing the adhesion level, the motor torque is unable to track the desired torque value. This could be clearly shown in Fig.6(a), where the steady-state error increases gradually with time elapsed. When changing the adhesion level from 1-5, we noticed a significant decrease in the motor torque value at the adhesion levels (2-3), and a severe non-linear drop in its value at the adhesion level 5, which represents a case of slip occurrence in the locomotive wheels at Rail contact as shown in Fig.6(b).

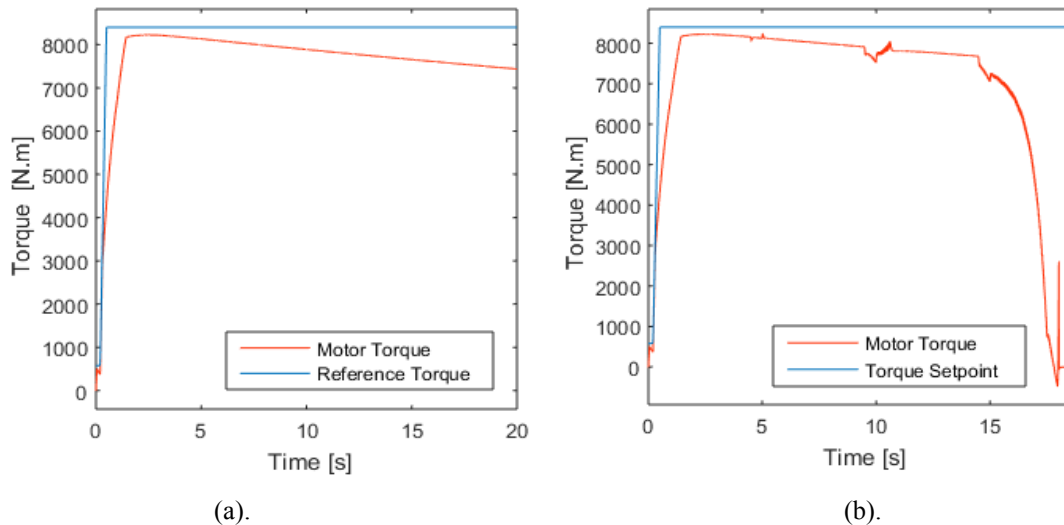


Fig. 6. a) Motor torque response at 8400N.m and 100km.h⁻¹ with no adhesion changes
 b) Motor torque response at 8400N.m and 100km.h⁻¹ with changeable adhesion levels

At low operating speeds of the locomotive at 10 km.h⁻¹, with desired torque value of 4200 N.m without changing the adhesion levels, the motor torque response is as shown in Fig.7(a), where the motor torque response speed is slower compared to the torque response in Fig.5(a).

When changing the adhesion levels, we noticed that the motor torque failed to track the desired torque value at the fifth level which represents the worst adhesion value as shown in Fig.7(b).

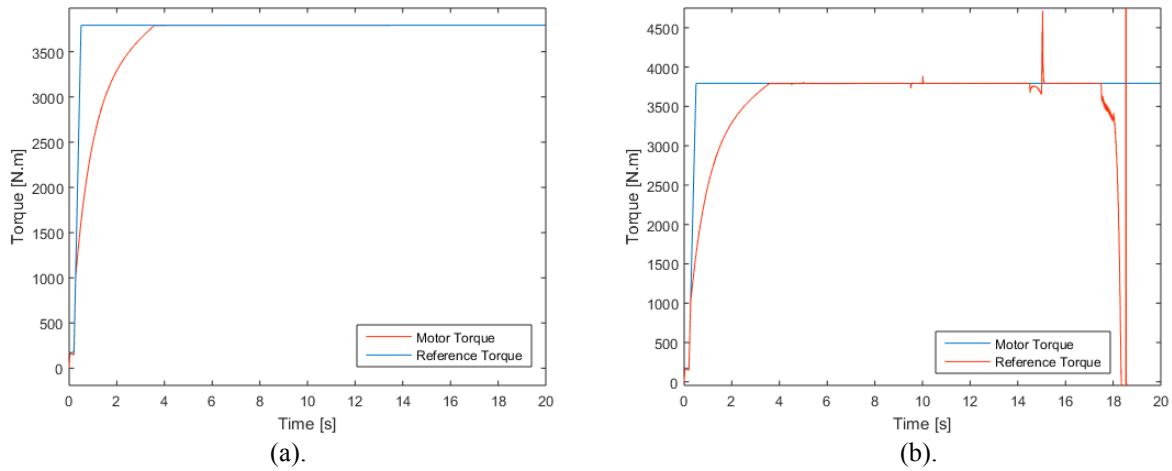


Fig. 7. a) Motor torque response at 4200N.m and 10km.h⁻¹ with no adhesion changes
 b) Motor torque response at 4200N.m and 10km.h⁻¹ with changeable adhesion levels

When we set a torque value of 8400N.m at a speed of 10 km.h⁻¹ without changing the adhesion level, we noticed a very slow response of the motor torque related to the desired torque value as shown in Fig.8(a). By changing the adhesion levels, we noticed a nonlinear decrement in the motor torque when the fourth adhesion level is reached, which reflects an occurrence of slip as shown in Fig.8(b).

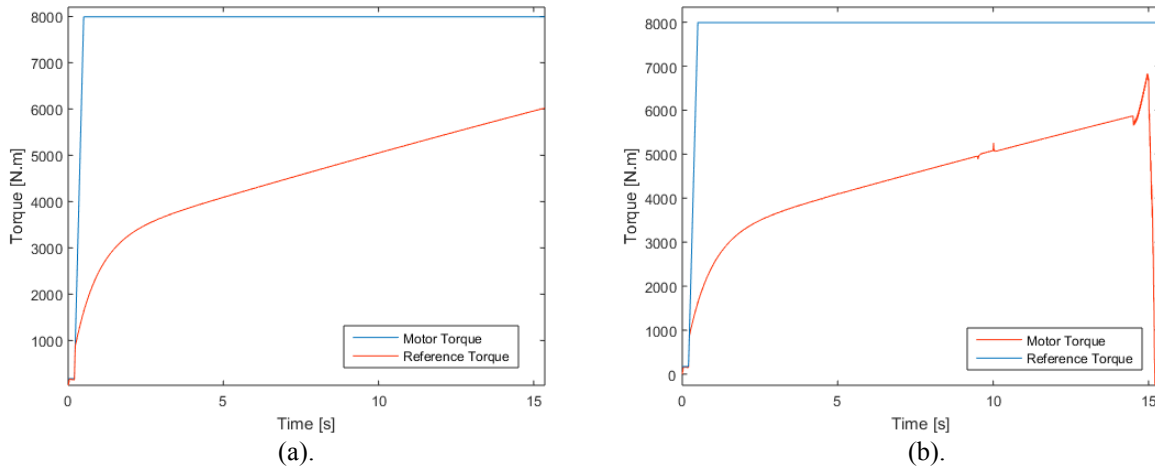


Fig. 8. a) Motor torque response at 8400N.m and 10km.h⁻¹ with no adhesion changes
 b) Motor torque response at 8400N.m and 10km.h⁻¹ with changeable adhesion levels

5 Conclusion

Studying the effect of the torque dynamic behavior of the asynchronous motor at specific operating conditions on the dynamics of the locomotive such as the movement of its wheels, taking into account changes in the adhesion conditions provides us with the possibility of early detection of dynamic phenomena occurrence such as slip.

Thus, this may help us to develop the current simulation model to include the possibility of optimizing locomotive motor torque and make it able to adapt to changing driving and operating conditions.

The simulation results address the importance of obtaining an appropriate motor control model capable of meeting the changing driving requirements of the traction motor.

acknowledgement

Work has been supported by the SGS grant 19/158/OHK2/3T/12.

Symbols

The symbols used in this paper are listed as follows.

$i_{s,dq}$	stator current in rotary frame (A)
$i_{s,\alpha\beta}$	stator current in stationary frame (A)
$u_{s,dq}$	stator voltage in rotary frame (V)
$u_{s,\alpha\beta}$	stator voltage in stationary frame (V)
$\psi_{s,dq}$	stator flux linkage in rotary frame (Wb)
$\psi_{s,\alpha\beta}$	stator flux linkage stationary frame (Wb)
$i_{r,dq}$	rotor current in rotary frame (A)
$i_{r,\alpha\beta}$	rotor current in stationary frame (A)
$u_{r,dq}$	rotor voltage in rotary frame (V)
$u_{r,\alpha\beta}$	rotor voltage in stationary frame (V)
$\psi_{r,dq}$	rotor flux linkage in rotary frame (Wb)
$\psi_{r,\alpha\beta}$	rotor flux linkage stationary frame (Wb)
r_s	stator resistance (Ω)
r_r	rotor resistance (Ω)
ω_m	angular speed ($\text{rad}\cdot\text{s}^{-1}$)
$M(t)$	electromechanical torque (N.m)

References

- [1] Y. Shimizu, K. Ohishi, T. Sano, S. Yasukawa, and T. Koseki, "Antislip re-adhesion control based on disturbance observer considering bogie vibration," in Proc. Eur. Conf. Power Electron. Appl., 2007, pp. 1–10.
- [2] P. Pichlik and J. Zdenek, "Locomotive velocity estimation for a slip control purpose by an unscented Kalman filter," 2017 18th International Scientific Conference on Electric Power Engineering (EPE), 2017, pp. 1-5, doi: 10.1109/EPE.2017.7967287.
- [3] R. Souad and H. Zeroug, "Comparison between direct torque control and vector control of a permanent magnet synchronous motor drive," 2008 13th International Power Electronics and Motion Control Conference, 2008, pp. 1209-1214, doi: 10.1109/EPEPMC.2008.4635433.
- [4] W. C. Duesterhoeft, M. W. Schulz and E. Clarke, "Determination of Instantaneous Currents and Voltages by Means of Alpha, Beta, and Zero Components," in Transactions of the American Institute of Electrical Engineers, vol. 70, no. 2, pp. 1248-1255, July 1951, doi: 10.1109/T-AIEE.1951.5060554.
- [5] R. H. Park, "Two-reaction theory of synchronous machines generalized method of analysis-part I," in Transactions of the American Institute of Electrical Engineers, vol. 48, no. 3, pp. 716-727, July 1929.
- [6] Paul C. Krause; Oleg Wasynczuk; Scott D. Sudhoff, "Analysis of Electric Machinery and Drive Systems", IEEE, 2002.
- [7] Takahashi, I.; Noguchi, T. "A New Quick-Response and High-Efficiency Control Strategy of an Induction Motor". IEEE Trans. Ind. Appl. 1986, IA-22, 820–827.
- [8] Depenbrock, M. "Direct self-control (DSC) of inverter fed induction machine". In Proceedings of the 1987 IEEE Power Electronics Specialists Conference, Blacksburg, VA, USA, 21–26 June 1987; pp. 632–641.
- [9] Casadei, D.; Serra, G.; Stefani, A.; Tani, A.; Zarri, L. "DTC drives for wide speed range applications using a robust flux-weakening algorithm". IEEE Trans. Ind. Electron. 2007, 54, 2451–2461.
- [10] R. Garg, P. Mahajan, N. Gupta and H. Saroa, "A comparative study between field oriented control and direct torque control of AC traction motor," International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), Jaipur, 2014, pp. 1-6.
- [11] <https://www.skoda.cz/produkty/lokomotivy/rychlikove-lokomotivy/>

IDENTIFICATION AND CONTROL USING GEA

Adrian Saldanha, Milan Hofreiter

Czech Technical University in Prague, Faculty of Mechanical Engineering

Abstract: This article presents the Guiding Evolutionary Algorithm (GEA) for optimization and its application in parametric identification and control of several processes. The key advantage of using this optimization approach is that it can be used for identification of linear as well as non-linear systems. The algorithm has previously been applied successfully to optimizing a number of benchmark problems and the results were shown to be quite promising. This article applies the GEA optimization approach to the problem of identification and control of a dynamic system. The approach is verified experimentally on a physical setup and the feasibility of this approach is studied and evaluated for practical implementation.

Keywords: optimization, identification, control

1 Introduction

The Guiding Evolutionary Algorithm (GEA) was proposed by Cao et al in [1]. The algorithm was introduced to exploit some of the advantages of already existing optimization such as the Bat Algorithm (BA) [2], the Particle Swarm Optimization (PSO) [3] and the Genetic Algorithm (GA) [4]. Like the BA and GA, the GEA is also a meta-heuristic algorithm which means that the optimization is problem independent and thus can be used in a wide range of applications. The GEA bears strong resemblance with the BA and the PSO in its particle-based movement, but unlike the two, it encompasses an added mutation operator which, in essence helps avoiding the solution getting stuck in a local optimum.

The main purpose of this article is to introduce the optimization method called the ‘Guiding Evolutionary Algorithm’ and secondly to apply the concept to a control problem involving system identification and control of dynamic systems. The results of the identification and control are finally applied to an experimental setup and the results are drawn.

The following sections are organized as follows: Section (2) introduces the concept of optimization and explains the Guiding Evolutionary Algorithm (GEA). Section (3) explains the procedure of system identification of an unknown system, following which, in Section (4) the procedure for tuning a PID controller to control the previously identified model is described. Section (5) consists of an experimental setup and the concepts described in the previous sections are applied to a real system. Finally, section (6) presents the conclusion.

2 Optimization

As part of this article, the GEA algorithm is investigated as a potential method for solving non-linear optimization algorithms. Like most nature-based optimization algorithms, the GEA consists of a number of ‘particles’ or ‘solutions’ randomized over a higher-dimensional domain, each solution with an associated cost-function. Added to this, the algorithm, consists of three main operators namely, ‘Crossover’, ‘Mutation’, and ‘Selection’. The *crossover* operator facilitates adequate mixing within the solution space. With each generation or iteration, every solution moves slowly towards the best solutions, i.e. the particle with the optimal cost function. The *mutation* operator is necessary in order to avoid the particles settling at a local optimum solution. It provides the essential exploratory framework for *diversification* of the solutions. Finally, the *selection* operator ensures that the

particle with the optimum cost function is selected as the new best. With the three operations in place, with every new generation, the solution slowly converges towards the best solution, which, may or may not be the global optimum. Note that heuristic algorithms in general, do not guarantee arrival at the global optimum but are highly efficient in finding the optimum result relatively quickly. Thus, these serve as an efficient way of searching through a given space to arrive at an acceptable solution in least time.

2.1 Guiding Evolutionary Algorithm (GEA)

The pseudocode for the GEA algorithm is shown in figure (1). The Guiding Evolutionary Algorithm, while similar to the BA and PSO, is essentially simpler to use due to the fewer number of parameters required for tuning the optimization, and unlike the two, it consists of an additional mutation operator which helps to avoid the solution settling at a local optimum.

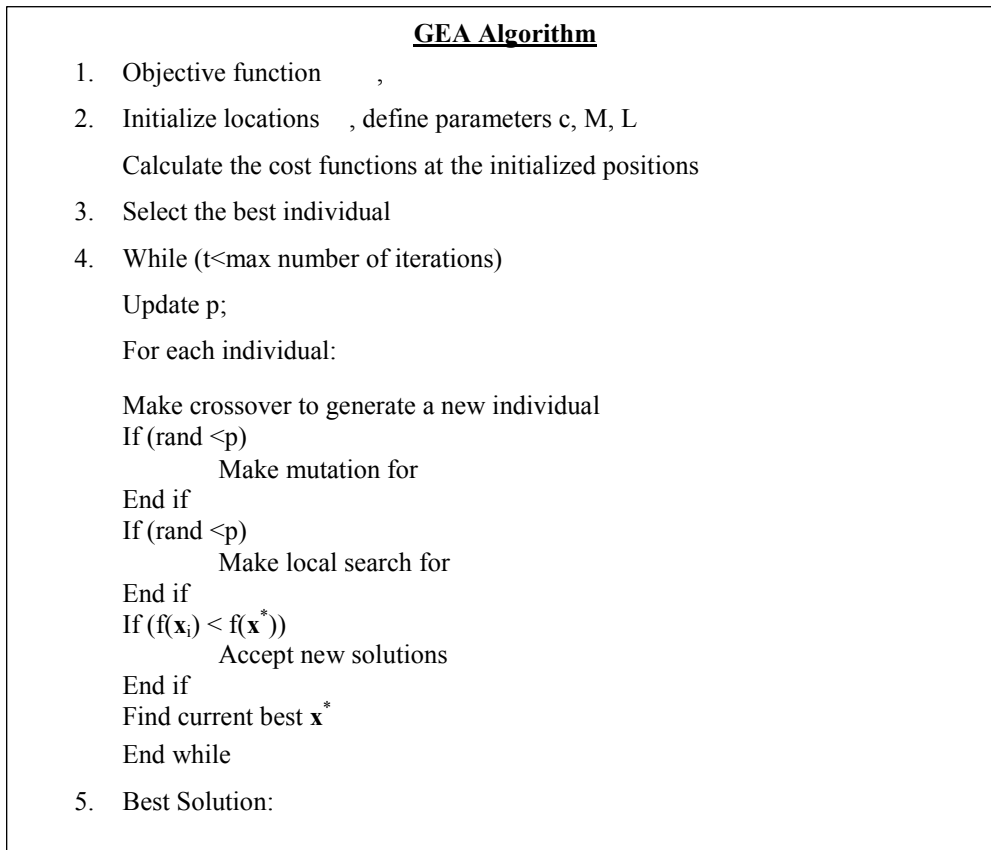


Figure 1 - Pseudocode - GEA Algorithm

1. Crossover: The crossover for GEA is given by:

$$x_i^t = x_i^{t-1} + (x_*^{t-1} - x_i^{t-1}) * \beta \quad (1)$$

Where,

β : Step length of position increment, uniformly distributed r.v

The step length β defines the rate of convergence and is generally between 1 and 2. A higher value represents faster convergence.

2. Mutation: The mutation provides the required exploratory mechanics for optimization. It can be given by the equation:

$$x_i^t = x_i^{t-1} + \epsilon M \quad (2)$$

Where,

ϵ : Uniform r.v [-1, 1]

M : Mutation vector,

$M_j = (\max(x_{ij}^{t-1} - a, b - x_{ij}^{t-1})) ; [a, b] = \text{range of } j^{\text{th}} \text{ dimension}$

According to GEA, the probability of mutation is given by the following:

$$p = c * \ln \frac{T_{max}}{T_{max}-t} \quad (3)$$

Where,

T_{max} : max number of generations

t : current generation

c : 0.2 (constant)

The above equation shows that the probability of mutation increases as the generations pass, thus it helps ensuring that the solution does not settle around the local maxima / minima.

3. Local Search: As before, the local search provides the necessary exploitative dynamics and can be given as follows:

$$x_i^t = x_i^{t-1} + \epsilon L \quad (4)$$

Where,

L : Local search vector

$L_j = 0.1 (b - a) ; [a, b] = \text{range of } j^{\text{th}} \text{ dimension}$

Similar to the mutation vector, the local search functions similar to the mutation except that it serves to find a solution around the current best unlike mutation, which helps to find a new solution around the unsearched territory.

Again, the probability of local search is given by p defined by equation (3).

2.2 Parameterization

Before simulating the above functions, the appropriate parameters must be set correctly to ensure that the algorithm works in an efficient manner. One of the advantages of the GEA lies in its inherent simplicity in that it consists of only a few parameters which can be easily tuned to ensure quick convergence. In this case, the main parameters for tuning are β and c which are taken directly from literature [1] as:

Parameter	Value
c	0.97
β	[0, 2]

The value of β determines how quickly or slowly the solution moves towards the best solution. A higher value of this parameter indicates quicker convergence, but it can also result in the final solution moving about the optimum value, whereas a lower value indicates a slower convergence to the optimum. For the following simulations, the value is taken as 2.

2.3 Results

The above algorithm was evaluated in the article [5] and it was shown that the GEA performs considerable better than some of the other existing algorithms namely, BA and PSO in terms of convergence rate and its tendency to reach the global optimum solution. Furthermore, it was shown that to achieve reaching the global optimum, it was necessary to apply the modified mutation and local search operations to equations (2) and (4) as:

$$x_i^t = x_i^{t-1} + \epsilon M_{j,*} (rand(1, j) < p) \quad (5)$$

And,

$$x_i^t = x_*^{t-1} + \epsilon L \cdot (\text{rand } 1, j < p) \quad (6)$$

The above modifications improve the exploratory tendencies of multimodal functions, by performing a probabilistic mutation and local search operation to the individual dimensions of each solution rather than to the solution as a whole. This helps avoiding the solution jumping over the optimum value, thereby enhancing the convergence characteristics.

3 Relay-based Feedback Identification

Following the above-described optimization approach, the next step is to apply this same algorithm in parametric identification of a dynamic system. The goal of the system identification task is to perform *black box identification* of an unknown process. The advantage of black box identification is the fact that this approach necessitates no knowledge about the physical system, but instead, uses the experimental data which includes inputs and outputs and a certain defining factor in terms of the cost function to identify the system to a reasonable accuracy. In this case, the ITAE (*Integral Time Absolute Error*) Method is used as a criterion for the cost function. With system identification, the end goal is to find an accurate model of the process so as to control it correctly.

3.1 System Schematic

The identification schematic can be seen in the below figure (2).

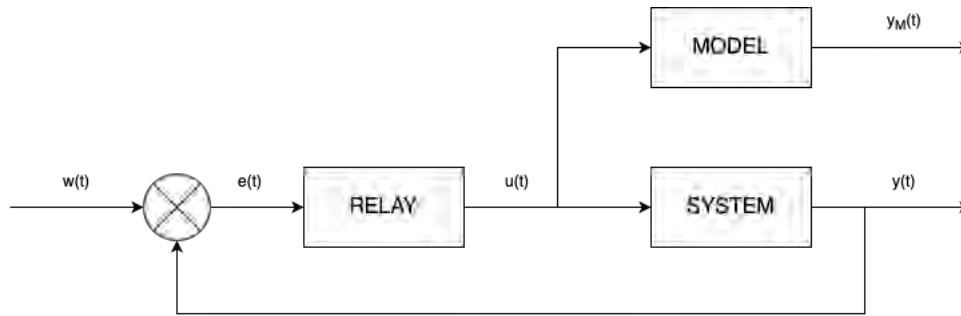


Figure 2 - Experimental Block Diagram Using Simulink

The above figure shows the process *or system* that is to be identified and then controlled. The input u is given by the output of the on-off relay which is used to generate sustained oscillations within the closed-loop system as proposed by Astrom and Hagglund [6]. Once the closed-loop system is running automatically, the output from the controller ‘ u ’ is fed as input to the model $G_M(s)$ which, is considered to be a Second Order Plus Time Delay (SOPTD) model. The reason for using an SOPTD model is that this model can represent almost any linear system. As explained by Ramakrishnan and Chidambaram, the SOPTD model can incorporate various processes such as under-damped and higher order processes in which case, an FOPTD model is not sufficient [7]. Furthermore, SOPTD models can also be used for unstable processes in which case, an FOPTD model is not sufficient.

Thus, by knowing the input ‘ u ’, the outputs ‘ y ’ and ‘ y_M ’, we can proceed with identification of the process.

3.2 Problem Statement

First, the terms related to the problem are defined as follows:

1. Model: The SOPTD model describe above can be represented as follows:

$$G_M s = \frac{K}{a_2 s^2 + a_1 s + 1} e^{-sT_d} \quad (7)$$

Where, ‘ K ’ = Process gain, ‘ θ ’ = Time Delay, ‘ a_1 ’ and ‘ a_2 ’ are dynamic constants of the transfer function.

2. Parameters: Using the SOPTD model from equation (7), the goal is to identify the parameters as follows:

$$\mathbf{x} = K, a_1, a_2, T_d^T \tag{8}$$

3. Cost function: The criterion for optimization can be given by the cost function J which can be given by:

$$J = \int_0^{T_k} y_m t - y t^2 dt \tag{9}$$

Where,

y_m = Model output

y_t = Process output

T_k = Simulation time

The above cost function uses the ITAE criterion as our optimization constraint for minimizing the error over a pre-set simulation time.

4. Constraints: The constraints restrict the upper and lower limits of each parameter

$$K, a_1, a_2, T_d > 0 \tag{10}$$

Thus, the problem statement can be reduced to the following:

Estimate parameters \mathbf{x} of process model G_M by minimizing the cost function J subject to the constraints

$$K, a_1, a_2, T_d > 0.$$

4 PID Control

With the identification procedure described in the previous section, the next step is controlling the actual process using the identified model of the system.

4.1 Closed-Loop System schematic

In this article, the controller in use is a PID controller. The block diagram of the controller can be shown in the below figure.

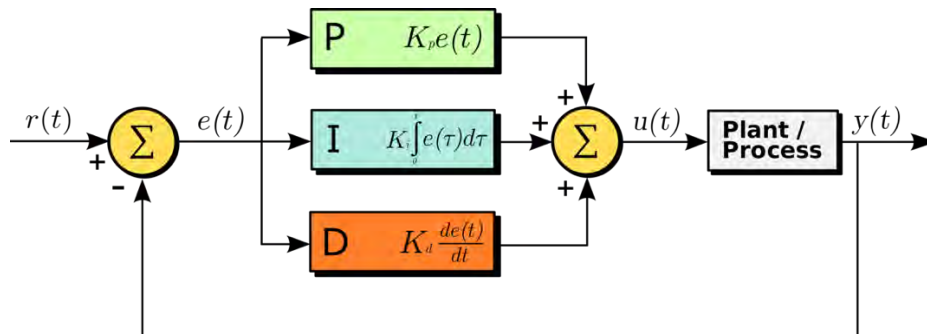


Figure 3 - PID Controller Block Diagram [8]

The equation for the controller in Laplace domain can be given by:

$$R s = r_p + \frac{r_i}{s} + r_d s \tag{11}$$

Where,

r_p = Proportional Gain – P-term K_p

r_i = Integral Gain – I-term (K_i)

r_d = Derivative Gain – D-term (K_d)

It must be noted that in the above figure (3), the real unknown process is to be controlled using only the

identified model with no more information about the system.

4.2 Tuning Methods

To tune the controller is to find suitable values of the proportional, integral and derivative gain so as to achieve the desired control objectives. In this article, the PMC method [9] for tuning the controller is used.

Before applying the tuning procedure, the SOPTD Model $G_M s = K \cdot \frac{e^{-sT_d}}{a_2s^2+a_1s+1}$ obtained in section (3) is reduced to one of the below forms:

$$G s = \frac{(K\omega_0^2)e^{-sT_d}}{s^2+2\zeta\omega_0s+\omega_0^2} \text{ for oscillatory processes} \tag{12}$$

Where,
 ω_0 = oscillation frequency
 ζ = damping ratio

or

$$G s = \frac{Ke^{-sT_d}}{T_1s+1 T_2s+1} \text{ for non-oscillatory processes} \tag{13}$$

Where,
 T_1, T_2 = Time constants of modelled system

The relations for tuning the controller are given by the below tables

a) Non-oscillatory Processes ($\zeta \geq 1$)

Table 1 - Tuning Equations (Non-Oscillatory Process)

No.	Parameter	PMC	
		$T_d > 0$	$T_d = 0$
1	r_p	$r_p = a_1 \cdot r_i$	$r_p = a_1 r_i$
2	r_i	$r_i = \frac{\pi - 2\gamma}{2 \cdot K T_d}$ or $r_i = \frac{\pi}{2T_d \cdot m_A \cdot K}$	$r_i = \frac{1}{K\tau_c}$
3	r_D	$r_d = a_2 \cdot r_i$	$r_d = a_2 r_i$

In the above equations, the parameter τ_c is the closed loop time constant and can be estimated using a general rule:

$$\tau_{dom} > \tau_c > T_d \tag{14}$$

Where,

τ_{dom} = dominant time constant of the process

As a thumb rule, we shall estimate the value of τ_c as: $\tau_c = \frac{\tau_{dom}}{3}$.

For PMC Tuning, the parameters γ and m_A refer to the phase margin and gain margin respectively. These values are generally in the range:

$$\frac{\pi}{6} < \gamma < \frac{\pi}{3}; 2 < m_a < 5 \tag{15}$$

It must be noted that the PMC tuning method is suitable for an SOPTD Model only.

b) Oscillatory Processes ($\zeta < 1$)

Table 2 - Tuning Equations (Oscillatory Process)

No.	Parameter	PMC	
		$T_d > 0$	$T_d = 0$
1	r_p	$r_p = a_1 \cdot r_i$	$r_p = a_1 r_i$
2	r_I	$r_i = \frac{\pi - 2\gamma}{2 \cdot K \cdot T_d}$ or $r_i = \frac{\pi}{2T_d \cdot m_A \cdot K}$	$r_i = \frac{1}{K\tau_c}$
3	r_D	$r_d = a_2 \cdot r_i$	$r_d = a_2 r_i$

Using the identified model of the system and the formulas listed in the above table, the tuning of the actual process is carried out.

5 Identification and Control on Physical System

To verify the theory and algorithms presented in the previous sections, we apply the same concepts of identification and PID control on a real system from the Automatic Control Laboratory.

5.1 Experimental Setup

The system which we will be using is a combination of two chambers arranged vertically in a tube with a system of interconnecting valves. The schematic of the setup can be seen in the below figures.

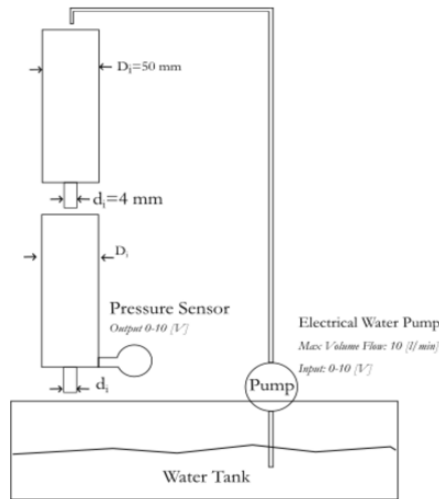


Figure 4 - Two tanks - Functional Diagram

The experiment is a combination of two hollow chambers arranged vertically in a tube with a system of interconnecting valves. Pump C1 supplies water to the upper chamber, thus, delivering a pressure head which causes the accumulated water in the upper chamber to trickle down into the lower chamber and eventually back into the tank. A pressure differential sensor is used to map the height of the water in the lower column. Since the upper and lower tanks are connected to each other in series, the system is, by default a second-order system. To perform the experiment, we need to supply an input signal 0-10V to the pump via the MATLAB / SIMULINK software from the PC which is then transmitted to the pump which in turn pumps up water to the upper chamber A1. From the bottom hole of the upper chamber, water trickles down to the lower tank B1. The system output is measured in terms of the height of water column in the lower chamber B1. This output is measured by a pressure sensor which converts the pressure head to an equivalent height of water column. Before the experiment is performed, it is necessary to test the system for its static characteristics so that we supply the inputs and obtain the readings at around the operating point.

5.2 Simulink Schematic Setup

Similar to the Simulink setup in section (3), the experimental Simulink Schema of the physical system is arranged likewise, except that in this case, the input is not fed directly to the theoretical model (see figure (5)), but instead is done separately once the readings have been collected. Therefore, there are two separate schematics, one for the physical system and the other, the *Identification Schematic* similar to the one shown in figure (2)..

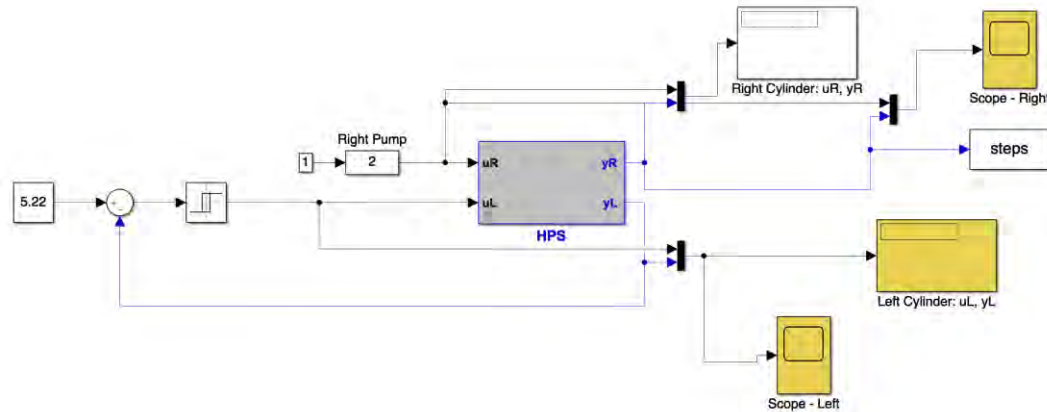


Figure 5 - Simulink Schematic - Physical system

As can be seen from the schematic in Figure (5), the Simulink model is supplied with an input from the relay controller which in turn causes the system to auto-oscillate. From this setup, the input to the system u_L and the output y_L are recorded from the practical setup.

Once the values are collected, these are now fed to the simulation setup (*Identification Schematic*) which operates within the MATLAB / Simulink environment only. The advantage of this is that using just one reading from the physical setup, we can simulate the theoretical model a number of times until we obtain the model which nearly describes the unknown process. For optimization, the same input u_1 which was supplied to the real process is fed to the theoretical SOPTD model to obtain its output y_m . By comparing the model output y_m with the real output y_1 , the previously described method using the ITAE criterion can be applied.

5.3 Identification Process Model

Using the above-described procedure and the one described in section (3), the identified parameters \mathbf{x} is given by:

$$\mathbf{x} = 1.1179, 1.295, 35.17, 4.43 \quad (16)$$

or

$$G_M s = \frac{1.1179}{1.295s^2 + 35.17s + 1} e^{-4.43s}$$

5.4 PID Control

Using the model obtained in the previous section, the experimental process is controlled using the PID control method detailed in section (4.2). The tuning parameters for control of the identified process from equation (16) are computed as follows:

Table 3 - PID Tuning Parameters - Physical System

No.	Parameter	PMC
1	Proportional Gain, r_p	0.961
2	Integral Gain, r_i	0.027
3	Derivative Gain, r_d	0.035

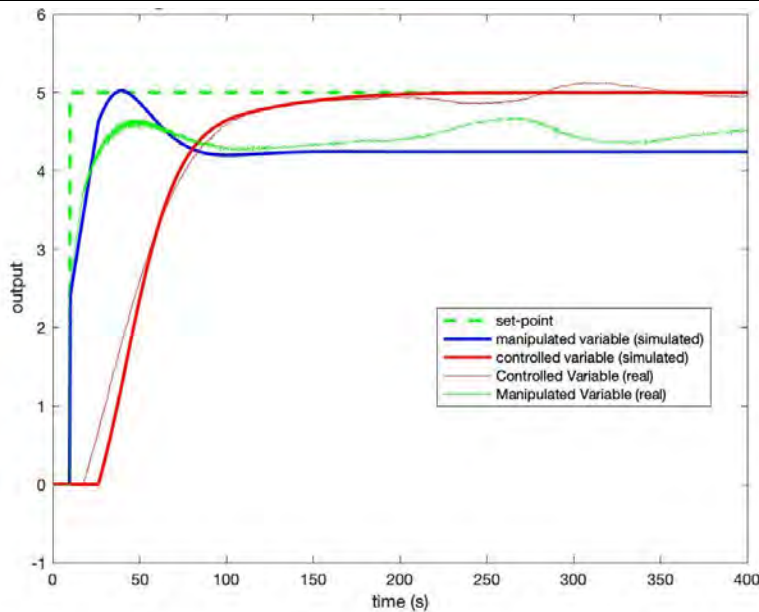


Figure 6 - PID tuning results

The figure (6) shows the PID tuning results of the physical setup. As can be seen from the figure, the controlled variable from the real setup closely matches the controlled variable from the simulated model and thus, is a close reflection of the actual process. Additionally, the PID parameters selected using the PMC tuning procedure are extremely precise and with minimum overshoot.

6 Conclusion

The GEA algorithm is evaluated as a potential method for optimization of a wide variety of functions. In this article, the algorithm is used perform system identification of an unknown process from which the controller parameters are calculated. One of the main advantages is that the method is suitable for different types of processes, be it linear or non-linear processes. Additionally, for higher order controllers, optimization of the controller can be performed using the GEA, thus making it a universal approach in solving problems of greater complexity.

Acknowledgement

This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS19/158/OHK2/3T/12

REFERENCES

- [1] L. a. X. L. a. G. E. Cao, "A Guiding Evolutionary Algorithm with Greedy Strategy for Global Optimization Problems," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1-10, 2016.
- [2] X.-S. Yang, "A New Metaheuristic Bat-Inspired Algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, Berlin, Springer, 2010, pp. 65-74.
- [3] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, Perth, 1995.
- [4] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*, Cambridge, MA: MIT Press, 1992.
- [5] A. Saldanha and M. Hofreiter, "Relay Feedback Identification using Guiding Evolutionary Algorithm," in *New Methods and Practices in the Instrumentation, Automatic Control and Informatics 2020*, Lobec, 2020.
- [6] K. Astrom and T. Hagglund, "Automatic Tuning of Simple Regulators with Specifications on Phase and Amplitude Margins," *Automatica*, Volume 20, Issue 5, pp. 645 - 651, 1984.
- [7] V. Ramakrishnan and M. Chidambaram, "Estimation of a SOPTD transfer function model using a single asymmetrical relay feedback test," *Computers & Chemical Engineering*, Volume 27, Issue 12, pp. 1779-1784, 2003.
- [8] "PID Controller," 09 08 2020. [Online]. Available: https://en.wikipedia.org/wiki/PID_controller.
- [9] M. Hofreiter, *Zaklady Automatickeho Rizeni*, Prague: CVUT, 2016.
- [10] A. Gupta, *Bat Optimization Algorithm*, MATLAB Central File Exchange, 2020.

APLIKACE PRO INDUSTRIAL EDGE (APPLICATION FOR INDUSTRIAL EDGE)

Jakub Znamenáček¹

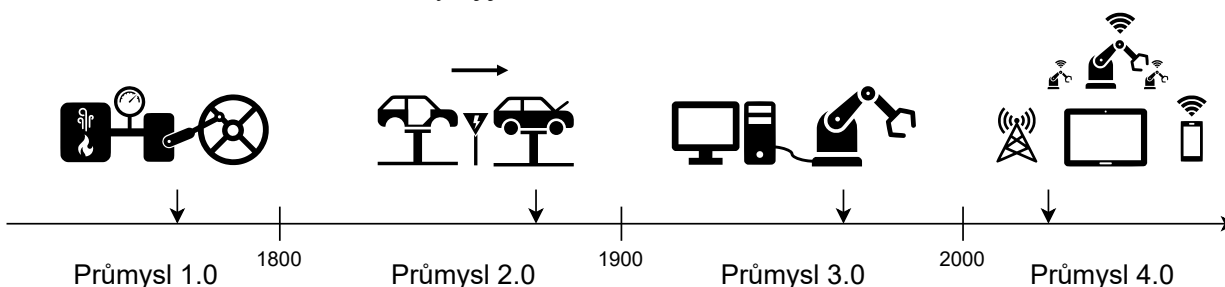
¹ Ústav přístrojové a řídicí techniky, fakulta strojní, ČVUT v Praze, jakub.znamenacek@fs.cvut.cz

Abstrakt: Obsahem této práce je vývoj backendu aplikace pro systém Industrial Edge v programovacím jazyce Java. Tato aplikace je určena pro monitorování lisovacího procesu prostřednictvím protokolu OPC UA a zajišťuje detekci odchylek od uživatelem stanovené tolerance, zpětnou vazbu pro PLC řídicí lis a případné ukládání zjištěných problémů do databáze za účelem pozdějšího zobrazení. Kromě samotného vývoje backendu je část práce věnována též tvorbě Dockerového obrazu celé aplikace, jenž je možný použít se systémem Industrial Edge.

Klíčová slova: Industrial Edge, Java, backend, lis, tváření kovů, OPC UA, Docker

1. Úvod

Průmysl prochází neustálým technologickým vývojem, který vede ke zvýšení kvality a efektivity výroby. Tento vývoj je často rozdělován do čtyř kategorií viz [1], a to na první průmyslovou revoluci (průmysl 1.0), pro kterou byl typický přechod od využívání lidské a zvířecí síly k využití parních strojů. Další velkou změnu přineslo zavádění elektrifikace a rozšíření masové výroby, které nastalo v devatenáctém století a je označováno za průmysl 2.0. Dvacáté století poté patřilo rozvoji výpočetní techniky a tento trend se promítl i do výroby. Zavádění PLC pro řízení produkce dalo základy průmyslu 3.0, bez kterého je těžké si představit jakoukoli dnešní továrnu. Nyní se nacházíme na hraně další velké průmyslové revoluce. Za základy průmyslu 4.0 je brána všudypřítomná komunikace mezi zařízeními a sběr dat z nich. Tento vývoj je naznačen na Obr 1.



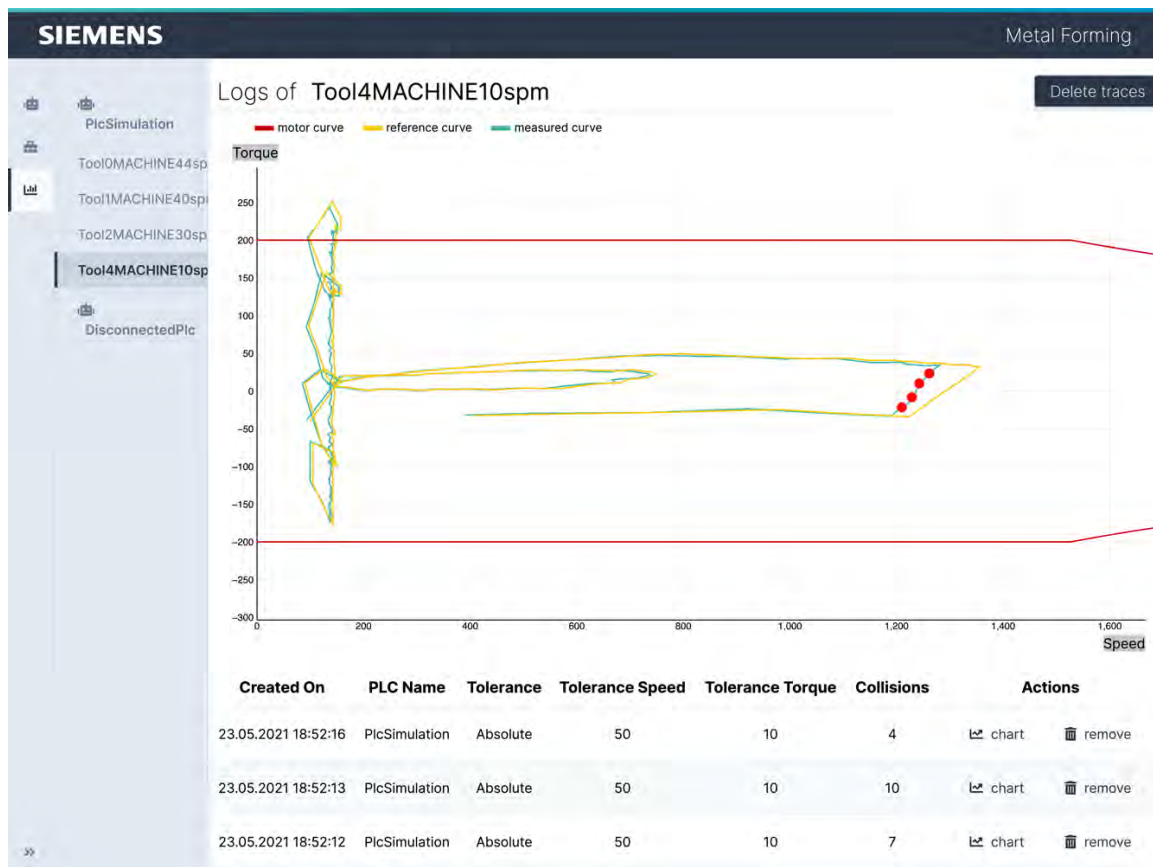
Obr. 1 Vývoj průmyslu

Společnost SIEMENS nedávno přišla se svou platformou Industrial Edge, která je právě zaměřená na průmysl 4.0 a umožňuje snadný sběr, analýzu a sdílení dat z jednotlivých částí výroby. Mezi jednu z jejích výhod patří jednoduchá správa a instalace aplikací, a to jak pro vývojáře, tak pro koncového uživatele. Aplikace jsou založeny na technologii Docker, která umožňuje jejich běh nezávislý na platformě a programovacím jazyce, a tak je není těžké vyvíjet i bez drahého hardwaru, což by mohlo přilákat velké množství developerů.

Tato práce byla zaměřena na vývoj backendové části aplikace právě pro tuto platformu a na vytvoření výsledného dockerového obrazu.

2. Představení aplikace

Aplikace slouží k monitorování lisovacího procesu na základě porovnávání naměřené a referenční křivky reprezentující jeden lisovací cyklus. Tato data jsou získávána přes protokol OPC UA v podobě dvou polí obsahujících 360 hodnot rychlosti, resp. kroutícího momentu. Ovládání celé aplikace probíhá skrze frontend, jehož tvorba nebyla součástí této práce, jenž je rozdělen do tří hlavních sekcí. Správy PLC, správy nástrojů a správy a zobrazení logů. Poslední jmenovaná část je znázorněna na následujícím obrázku.



Obr. 2 Ukázka části frontendu výsledné aplikace pro správu logů

V sekci pro správu PLC je možné vytvořit připojení k jednotlivým PLC, které řídí lisovací proces. Část věnovaná nástrojům umožňuje spuštění či vypnutí automatického monitorování, spuštění kalkulace referenční křivky pro jednotlivé nástroje, nebo úpravu jejich parametrů jako jsou například jméno, požadovaná tolerance či typ zpětné reakce, který je předáván zpět do PLC v případě, že dojde na aktuálním cyklu k překročení uživatelem stanovené tolerance. Jakmile tato situace nastane dojde též k vytvoření záznamu v databázi. Tyto informace je po té možné zobrazit na poslední obrazovce, které je právě na Obr. 2. Zde je vidět graf naměřených hodnot jak pro referenční křivku, tak pro křivku naměřenou. Červenými body jsou zde znázorněny ty části, v kterých došlo k překročení limitů. Pod grafem je též možné dohledat jaká tolerance byla nastavena, či kdy k problému došlo.

Veškerá logika a zajištění komunikace jak s PLC, tak s databází je zpracovávána na backendu, pro předávání dat mezi frontendem a backendem bylo tedy nutné vytvořit vhodné API, které by jednoduše a efektivně předávalo získaná data a umožnilo ovládání jednotlivých úkonů.

3. Architektura backendu

Architekturu backendové části je možné rozdělit do třech hlavních vrstev, a to vrstvy kontrolerů, servisní vrstvy a databázové vrstvy.

3.1 Vrstva kontrolerů

Tato vrstva se stará o zpracovávání dotazů zasílaných z frontendu, převod tříd použitých v backedové části na DTO pro zajištění co největší separace a o kontrolu formátu přijatých dat. Je zde tedy formována struktura API, která je naznačena na následujícím obrázku.

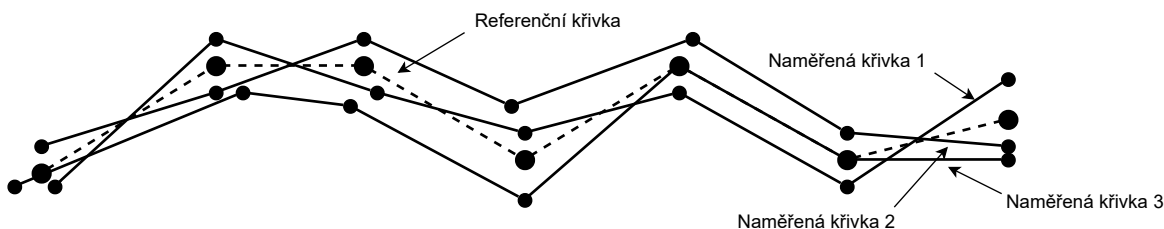


Obr. 3 Struktura API pro zajištění komunikace mezi frontendem a backednem

Endpoints na Obr. 3 slouží ke kontrole všech důležitých částí backendu a komunikace zde probíhá pomocí protokolu HTTP. Kromě této části backend disponuje též částí, která se stará o zasílání informací pomocí protokolu WebSocket, který umožňuje zasílání dat bez prvotního dotazu z frontendu. To se hodí například v situaci, kdy dojde k detekci nového nástroje.

3.2 Servisní vrstva

Zde se nachází jak veškerá důležitá logika, tak i kód zajišťující připojení k jednotlivým PLC. Lze zde tak nalézt část obstarávající výpočet referenční křivky a část sloužící k vyhodnocení právě proběhlého lisovacího cyklu. Obě tyto části jsou založené na návrhovém vzoru pozorovatel, a tak jsou okamžitě informovány o nově přichozících datech z PLC. Pokud je tedy spuštěn výpočet referenční křivky a jsou zaznamenána nová data dojde k její tvorbě na základě uživatelem zvoleného počtu cyklů. Tento proces je naznačen na Obr. 4, kde je možné vidět princip výpočtu referenční křivky ze třech cyklů. Výpočet není nikterak komplikovaný a pro sobě odpovídající body je pouze vypočten jejich aritmetický průměr jak pro rychlosti, tak pro kroutící momenty.

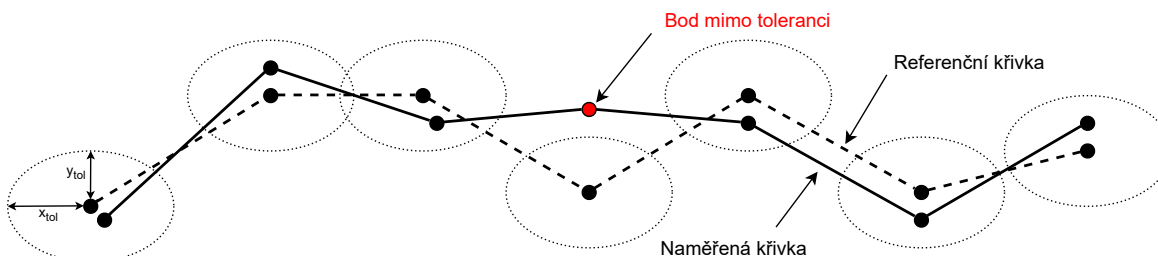


Obr. 4 Výpočet referenční křivky

Takto vypočtená křivka může být poté následně použita jako vzor pro automatické monitorování lisovacího cyklu, kdy nově získaná data nesmí být od jejich bodů více vzdálené než uživatelem stanovené tolerance. Toto je zajištěno tak, že bod musí ležet uvnitř elipsy jejíž střed je v bodě referenční křivky a délky její hlavní a vedlejší

poloosy odpovídají tolerancím viz Obr. 5. Rovnice (1) tedy představuje rovnici, kterou musí porovnávaný bod splňovat.

$$\frac{(x-x_{ref})^2}{x_{tol}^2} + \frac{(y-y_{ref})^2}{y_{tol}^2} \leq 1 \quad (1)$$



Obr. 5 Princip validace aktuálně naměřené křivky v porovnání s křivkou referenční

3.3 Databázová vrstva

Poslední vrstvou je vrstva databázová zajišťující získávání a ukládání dat do databáze. Aplikace využívá k ukládání dat relační databázi PostgreSQL. Struktura tabulek v databázi by šla rozdělit do dvou hlavních sekcí a to sekce, která obsahuje aktuální data o spravovaných PLC a jejich nástrojích a částí, která udržuje historické informace zaznamenané při překročení tolerance.

4. Závěr

V rámci této práce byl vytvořen plně funkční backend v programovacím jazyce Java pro Industrial Edge aplikaci monitorující lisovací proces. Kromě vývoje backendu byly též vytvořeny i předpisy pro tvorbu Dockerových obrazů, kombinujících databázi, backend a frontend. Ty poté umožňují běh aplikace na platformě Industrial Edge. Vytvořená část kódu byla psána technikou TDD, a proto je více než 80 % tříd pokryto unit testy, které při každé kompilaci kontrolují správnou funkčnost programu, což zaručuje jeho vysokou spolehlivost. Kromě testování kódu byla třetí stranou otestována i samotná aplikace v laboratoři společnosti SIEMENS v Německu. V průběhu ověřování funkčnosti aplikace s reálným lisem nedošlo k odhalení žádného problému.

Literatura

- [1] Chanchal Dey a Sunit Kumar Sen. *Industrial automation technologies*. First edition. Boca Raton, FL: CRC Press, 2020. ISBN: 9780367260422.

CONTROLLING MOTION APPS OF MOTION TERMINAL FESTO VTEM BY NODE-RED

Jan Janovský

České vysoké učení technické v Praze, Fakulta strojní, jan.janovsky@fs.cvut.cz

Abstrakt: Článek nejprve přibližuje tzv. Motion Terminal Festo VTEM, nový fenomén spojený s érou průmyslu 4.0 a pojmem digitalizovaná pneumatika. Dále se zabývá vývojářským nástrojem Node-RED, který je ve značné míře používán pro programování IoT (Internet věcí) aplikací. Na konec je představena jednoduchá praktická ukázka práce, kdy je stručně vysvětleno naprogramování aplikace pro komunikaci a řízení produktu Motion Terminal Festo VTEM.

Klíčová slova: node-red, modbus, Motion Terminal Festo VTEM, vizualizace

1 Úvod

Digitalizovaná pneumatika, pojem, pod kterým si lze představit mnoho. Spojení těchto dvou slov si firma Festo AG & Co. KG vzala za své při zveřejnění svého výrobku Motion Terminal Festo VTEM v polovině minulého desetiletí. Nový ventilový terminál, jehož jednotlivé ventily jsou řízeny aplikacemi různých funkcí, kterými lze nahradit až 50 standartních výrobků. [1] Docíleno toho bylo díky dvojici piezovenilům a čtyř sedlových ventilů zapojených do plného můstku a integrovanému systému čidel pro regulaci a diagnostiku v každém z ventilů. Spojením mechaniky, elektroniky a softwaru je vytvořen velice kompaktní výrobek, který dokáže svou schopností přizpůsobit se chtěné průmyslové aplikaci zjednoduší a zrychlí objednávku, montáž i programování. [2] I přes několik let přítomnosti na trhu není tento produkt v průmyslu příliš rozšířený, i přes to že vytváří samotný oddíl na trhu bez zatím znatelné konkurence. Důvodem může být nedostatek zkušeností s jeho prací.

Za internet věcí (Internet of Things) je považována síť fyzických zařízení, která jsou vybavena elektronikou, softwarem, senzory a síťovou konektivitou, které umožňují tyto zařízení propojit a vyměňovat si data či je posílat online službám. [3] Každé zařízení je však schopno samostatně pracovat. Kvůli velké rozmanitosti komunikačních řešení, výrobců zařízení a velké konkurenci je však v dnešní době občas problém takovouto síť jednoduše vytvořit. Každý výrobce chytrých zařízení, která jsou schopna mezi sebou komunikovat, prosazuje svoje řešení. Při použití zařízení od jiných výrobců stoupá riziko nekompatibility či množství času stráveného nad dokumentací výrobků. Pro tento problém společnost IBM vytvořila vývojářský nástroj pro vizuální programování Node-RED. Jeho cílem je spojit hardware zařízení s online službami a vytvořit tak síť pro komunikaci mezi jednotlivými částmi i přes odlišné komunikační protokoly či další problémy. [4]

2 Motion Terminal Festo VTEM

2.1 Komponenty

Motion Terminal se skládá ze 4 až 5 hlavních komponent. Jsou jimi základna, řídicí blok CPX, ovladač Motion Terminal CTMM, ventily VEVM (4-8) a vstupní moduly (volitelné). CPX terminál navazuje spojení s vyšším řídicím systémem buď jako samotný interní ovladač nebo může sloužit jako komunikační uzel. CMMT ovladač vytváří rozhraní mezi CPX terminálem a komponenty umístěnými na terminálu. Obsahuje také rozhraní ethernet pro zpřístupnění rozhraní WebConfig, které slouží především pro začátečnické pochopení jednotlivých aplikací (Motion Apps), které řídí funkci a činnost jednotlivých ventilů. [5]

2.2 Motion Apps

V současné době je k dispozici 10 aplikací, z nichž některé jsou součástí základního balíčku a některé je třeba dokoupit. Licence se vztahují přímo na jedinečný kód výrobku a jsou nepřenositelné mezi zařízeními. Zakoupené aplikace lze používat libovolně často na jakýchkoliv pozicích. [6]

Současné dostupné Motion Apps:

- funkce spínacího ventilu
- proporcionální průtokový ventil
- proporcionální regulační ventil
- proporcionální redukce tlaku podle modelu
- škrcení přívodu a odvětrání
- pohyb ECO
- předvolba času pohybu
- volitelný tlak (ECO)
- soft stop
- diagnostika úniků



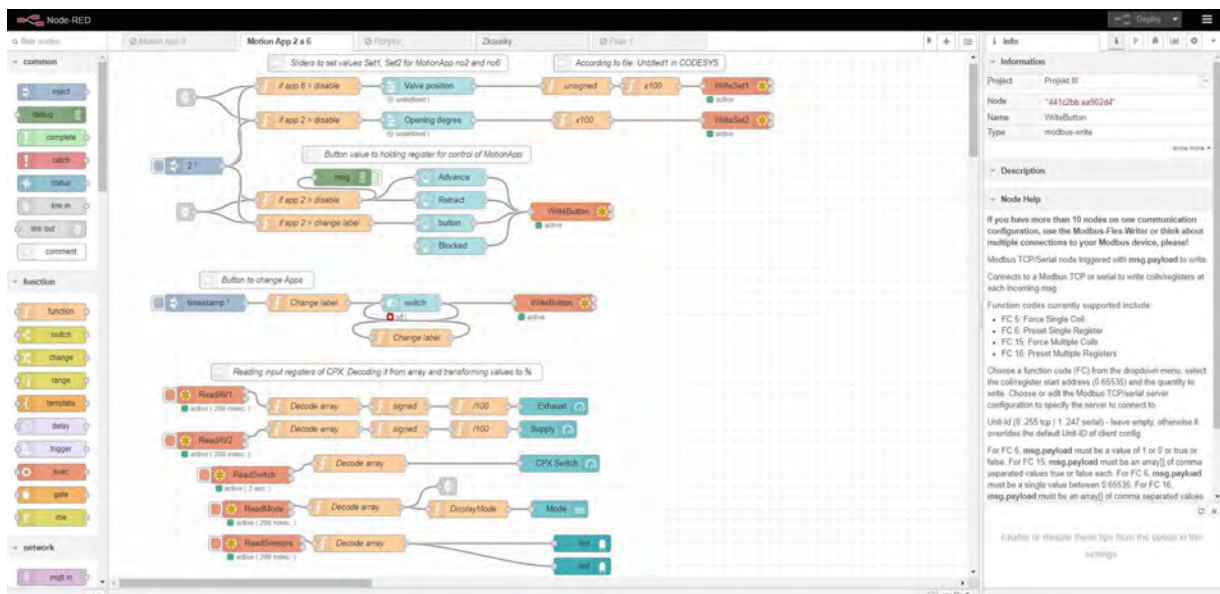
Obrázek 1 Motion Terminal Festo VTEM [2]

2.3 Komunikace mezi CPX a CMMT

Každému z přítomných ventilů na Motion terminálu je přiděleno 6 vstupních a 6 výstupních bajtů pro komunikaci mezi CPX a CMMT ovladači. Skrze tyto bajty lze mohou být přeposílány povely pro spuštění, zastavení či změnění parametrů právě pracující Motion App. Lze také změnit právě běžící aplikaci za jinou dle potřeb programátora v reálném čase. Rozsáhlá dokumentace je dostupná na stránkách výrobce. [5]

3 Node-RED

Vývojářský nástroj pro grafické programování, který poskytuje prohlížečový editor. Funguje na tzv. flow-based principu. Zprávy tečou pomocí větví (branches) mezi jednotlivými uzly (nodes), které mají jasně daný účel. Při přijmutí zprávy s ní něco udělají a poté ji pošlou dále. Další možností je vytvoření vlastních funkcí v jazyce JavaScript. Celý běh programu je postaven na základech Node.js. Díky tomu je vytvořena ideální situace pro běh na levných zařízeních, například Raspberry Pi. [7]



Obrázek 2 Ukázka naprogramování přístrojové desky pro ovládání Motion Apps č. 2, 6

3.1 Základy

Pro úspěšné zahájení projektu je nejdříve si třeba nainstalovat Node.js a Node-RED, vše dostupné jako open-source software. Po spuštění Node-RED je editor přístupný přes jakýkoliv prohlížeč webu na portu 1880, doporučen je však prohlížeč Google Chrome.

Hlavní strana editoru je rozdělena do tří hlavních částí. Vlevo se nachází panel se všemi nainstalovanými uzly (nodes). Největší část uprostřed tvoří pracovní plocha pro programování tzv. toků (flows). Vpravo pak výstupní panel, kde lze zobrazit například informace o právě zvoleném uzlu či výstupní kontrolní konzoli.

Předem nainstalované uzly se označují jako jaderné. Pro přidání rozšiřujících uzlů slouží obsáhlá knihovna, ve které lze hledat rozkliknutím nabídky v pravém horním rohu a vybráním možnosti „Manage palette“. Další možností jak objevit nové uzly je na oficiálních stránkách <https://flows.nodered.org/>. Velmi mocnou rozšiřující funkcí je například programovatelná přístrojová deska (dashboard) jako uživatelské rozhraní.

3.2 Vytváření aplikací

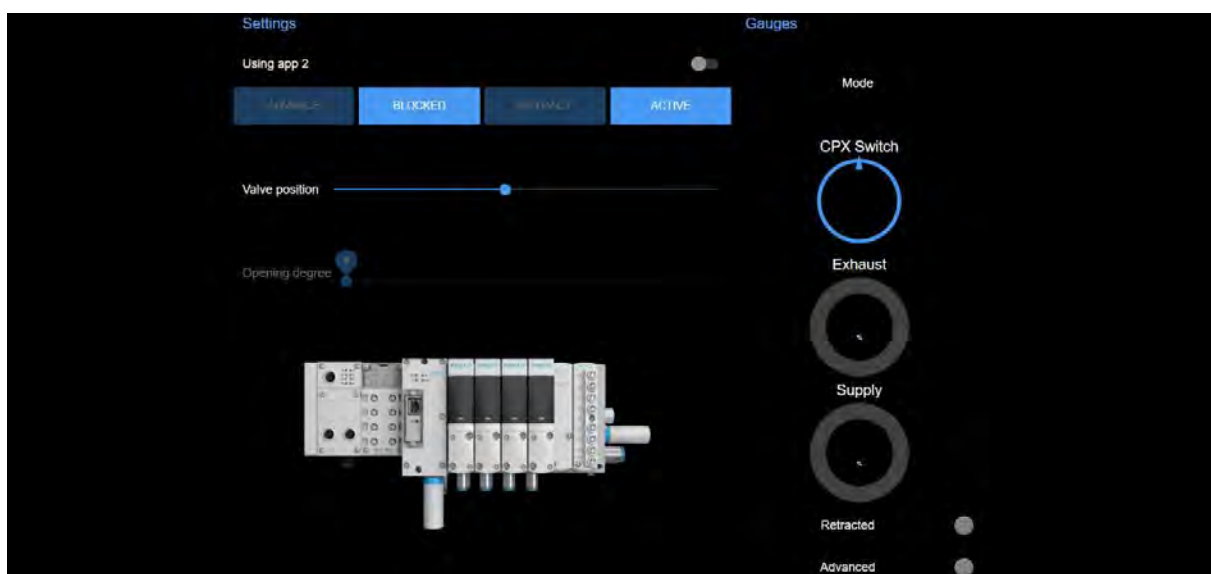
Node-RED funguje na základě toku dat, respektive zpráv. Na začátku každého toku je nutné nejdříve nějakou zprávu vygenerovat, ať už pouhým kliknutím (uzel „inject“), čtením dat z databáze či připojeného zařízení, připojením tlačítka z přístrojové desky atd. Zprávy pak různé uzly zpracovávají dle jejich funkce a posílají je v toku dále. Pro kontrolu programu slouží uzel „debug“, který aktuálně procházející zprávu pošle do kontrolní konzole vpravo. Zprávy jsou standardní JavaScript objekty s několika existujícími vlastnostmi vycházející ze způsobu, jakým byla zpráva generována. [8]

Aplikace, v jazyku Node-RED toky (flows), se vytvářejí jednoduše přesunutím uzlů z levého panelu na pracovní plochu. Každý uzel vyžaduje dodatečné nastavení. Uzly mají vždy buď jeden či více vstupních konektorů a nebo jeden či více výstupních konektorů. Uzly se mezi sebou propojují pomocí drátů, které charakterizují tok zpráv. Propojení mají pouze distribuční funkci, nijak zprávy nemění. Mocným vývojářským názorem je uzel „function“, který obsahuje textový editor a ve kterém lze programovat funkce v jazyce JavaScript.

4 Ukázka ovládání Motion Terminal skrz Node-RED

Cílem této názorné ukázky práce je přiblížit způsob ovládání Motion Apps spolu s ukázkou, jak efektivně využít Node-RED k navázání komunikace a naprogramování ovládacích prvků. Řídicí blok CPX, který je součástí Motion Terminal lze programovat v prostředí CODESYS.

Podrobný návod jak programovat aplikace Motion Terminal Festo VTEM lze najít v dokumentaci společnosti Festo vztahující se ke knihovně bloků pro ovládání ventilů z CPX jako vyššího ovladače. Další možností je dokument ... Stručněji řečeno, do PLC jsou nahrány dva programy POU (Program Organization Unit) obsahující již zmíněný funkční blok a hlavní program napsaný v jazyce ST. K ukázce byly použity Motion App číslo 2 (proporciální průtokový ventil) a číslo 6 (pohyb ECO).



Obrázek 3 Ukázka přístrojové desky (dashboard)

Pro komunikaci mezi počítačem, na kterém běží Node-RED a Motion Terminal jsem vybral protokol Modbus TCP. Pro zprovoznění protokolu na terminálu CPX je třeba přidat pod zařízení rozhraní ethernet, vybrat možnost Modbus_Slave_Device pro vytvoření serveru a nastavit IP adresu, port a ID. V Node-RED je nutno nainstalovat balíček uzlů pro protokol Modbus. Nakonec je nutné zadefinovat vnitřní proměnné vstupních registrů protokolu Modbus v systému CODESYS.

Posledním krokem je naprogramování přístrojové desky jako uživatelského rozhraní v Node-RED. Ta je rozdělena na dvě části. Levá část slouží jako sekce ovládací. Pomocí tlačítek a jezdce lze přepínat mezi aplikacemi a ovládat jejich pracovní hodnoty. K tomuto účelu je využito uzlů typu Modbus_Write, které zapisují aktuální hodnoty podle protokolu na vytvořený server. Pravá část je poté využita pro sledování stavu Motion Apps a připojeného pneumotoru. K tomuto úkolu využívá uzel Modbus_Read, jímž čte aktuální hodnoty výstupních registrů serveru. Inspirace byla načerpána z nativního rozhraní WebConfig.

5 Závěr

Cílem článku bylo přiblížit součásti a funkci produktu Motion Terminal Festo VTEM. Dále seznámit se s vývojářským nástrojem pro vizuální programování především pro oblast internetu věcí Node-RED a popsat základy programování a práce v tomto nástroji. Nakonec poskytnout malou ukázkou možnosti ovládání a diagnostiky Motion Apps skrze přístrojovou desku Node-RED pomocí komunikačního protokolu Modbus TCP.

Literatura

- [1] Digitalizovaný pneumatický systém. *Festo Motion Terminal* [online]. [cit. 2020-03-19]. Dostupné z: <https://www.festo.com/vtem/cs/cms/10169.htm>
- [2] FESTO, . Digitalizace pneumatiky: Festo Motion Terminal VTEM. *Automa*. 2018, (9), 2. ISSN 1210-9592.
- [3] Internet věcí. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001 [cit. 2020-03-19]. Dostupné z: https://cs.wikipedia.org/wiki/Internet_v%C4%9Bc%C3%AD
- [4] PEŠKA, Robert. Node-RED slibuje snadné programování IoT aplikací. *Vyvoj.cz* [online]. 2019 [cit. 2020-03-19]. Dostupné z: <https://vyvoj.hw.cz/node-red-slibuje-snadne-programovani-iot-aplikaci.html>
- [5] *Motion Terminal VTEM: Description, parametrisation, function*. Esslingen: Festo AG & Co. KG, 2018.
- [6] Motion Apps. *Festo Motion Terminal* [online]. [cit. 2020-03-19]. Dostupné z: <https://www.festo.com/vtem/cs/cms/motion-apps.htm>
- [7] *Node-RED* [online]. b.r. [cit. 2020-03-19]. Dostupné z: <https://nodered.org/>
- [8] Understanding and Using The Node-Red Message Object. *Steve's internet guide* [online]. b.r. [cit. 2020-03-19]. Dostupné z: <http://www.steves-internet-guide.com/node-red-message-object/>

ŘEŠENÍ INVERZNÍ KINEMATIKY PLANÁRNÍHO MANIPULÁTORU MLP NEURONOVOU SÍŤÍ

Vladimír Hlaváč

ČVUT v Praze, Fakulta strojní, hlavac@fs.cvut.cz

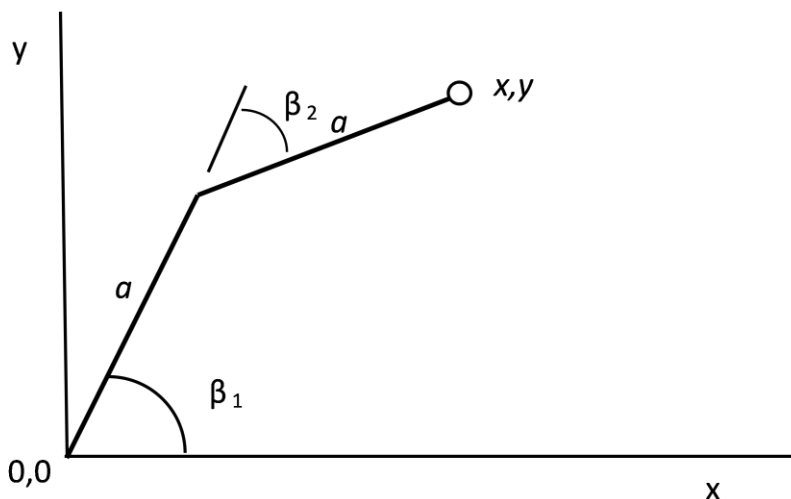
Abstrakt: Vícevrstvá neuronová síť typu MLP (multi-layer perceptron) je schopná interpolovat jakoukoli běžnou spojitou funkci, například kombinaci různých goniometrických funkcí a jejich inverzí. Následující článek popisuje její užití pro řešení úlohy inverzní kinematiky manipulátoru. Zatímco řešenou úlohu planárního manipulátoru lze řešit přímo, pro běžné robotické rameno pracující v prostoru vycházejí již vzorce poměrně komplikovanější a neuronová síť může představovat spolehlivější řešení.

Klíčová slova: Neuronová síť, MLP NN, planární manipulátor, inverzní kinematika.

1. Úvod

Problematika robotických manipulátorů je popsána například v doktorské prezentaci ing. Švejdy ke státní zkoušce [1]. Samotná disertační práce je dohledatelná na <https://portal.zcu.cz/portal/studium/>, zvolit “Kvalifikační práce” a vyhledat podle příjmení a typu práce (disertační). Kratší a přehlednější je disertační práce ing. Čejky [2].

Řešení s použitím neuronové sítě je popsáno v [3], [4]. Články jsou v podstatě totožné. Přináší popis úlohy a grafy z trénování neuronové sítě. V následujícím textu naleznete naproti tomu popis, jak postupovat a vizualizaci dosažených poloh po trénování neuronové sítě.



Obr. 1 Schematické znázornění úlohy. Poloha koncového bodu ramene je dána nastavením úhlů (kloubů) mezi jednotlivými segmenty a jejich délkou. Úhly se samozřejmě počítají jako kladné, pokud jsou proti směru hodinových ručiček; tak, jak je vyznačen úhel β_2 , má zápornou hodnotu.

2. Kinematika planárního robotického manipulátoru

Jak je znázorněno na obr. 1, polohu koncového bodu lze snadno spočítat:

$$x = a \cos \beta_1 + a \cos(\beta_1 + \beta_2) \quad (1)$$

$$y = a \sin \beta_1 + a \sin(\beta_1 + \beta_2) \quad (1b)$$

Tento výpočet se označuje jako přímá kinematika robotického manipulátoru. Většinou potřebujeme řešit opačnou úlohu, k zadaným souřadnicím navrhnout nastavení úhlů v kloubech manipulátoru, kterými jich lze dosáhnout. Tato úloha se označuje jako inverzní kinematika. Pro planární manipulátor je výpočet poměrně jednoduchý. Například můžeme nejprve převést požadovanou polohu koncového bodu do polárních souřadnic:

$$r = \sqrt{x^2 + y^2} \quad (2)$$

$$\varphi = \arctan(x/y) \quad (2b)$$

K dosažení tohoto bodu bude muset být délka ramena manipulátoru rovna r . Abychom jí nastavili, musíme nastavit úhel mezi oběma segmenty. Následující výpočet předpokládá, že délky obou segmentů budou stejné. Pak platí:

$$r = 2a \cos \beta_0 \quad (3)$$

$$\beta_0 = \arccos(r/2a) \quad (3b)$$

Pokud známe úhel mezi oběma segmenty β_0 a požadovaný úhel koncového bodu oproti počátku v polárních souřadnicích, můžeme snadno dopočítat oba nastavované úhly:

$$\beta_1 = \varphi + \beta_0 \quad (4)$$

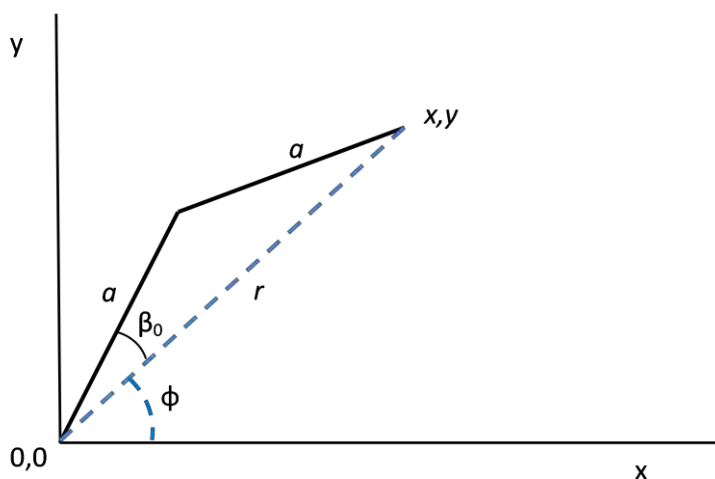
$$\beta_2 = -2\beta_0 \quad (4b)$$

Úloha má samozřejmě dvě řešení, podle toho, zda je rameno vyhnuto doprava, či doleva. Symetrické řešení je:

$$\beta_1 = \varphi - \beta_0 \quad (5)$$

$$\beta_2 = 2\beta_0 \quad (5b)$$

Úlohu lze řešit i když jsou ramena různě dlouhá, jen bude postup trochu komplikovanější.



Obr. 2 Transformace do polárních souřadnic.

3. Řešení neuronovou sítí

3.1 Příprava dat a trénování sítě

Data můžeme připravit přímo v Matlabu podle (1). U manipulátoru známe rozsah úhlů β i délku segmentů. Zvolíme vhodný krok a zadáme příslušné proměnné:

```
b1=(-0.2:0.1:2.4)';
b2=(0:0.1:2.6)';
```

Úhel, který svírají mezi sebou segmenty ramena manipulátoru, je generován vždy buď jako záporný, nebo jen jako kladný, abychom se vyhnuli duplicitnímu řešení kinematiky (pro výpočet inverzní funkce musí být funkce jednoznačná).

Budeme zkoušet různé počty neuronů ve skryté vrstvě. Celkový počet parametrů neuronové sítě lze spočítat:

$$n_v = (n_i + 1)n_h + (n_h + 1)n_o \quad (6)$$

n_v – počet stavitelných parametrů

n_i – počet vstupů

n_h – počet neuronů ve skryté vrstvě

n_o – počet výstupních neuronů

Například pro dva vstupy, dva výstupy a deset neuronů ve skryté vrstvě vychází 52 parametrů (+1 reprezentuje bias, posunutí pracovního bodu neuronu). Obecně platí, že pro trénování neuronové sítě by mělo být desetkrát tolik vzorků, tedy například 520.

Pro znázornění v grafu by ale vycházelo příliš mnoho bodů, proto pro vykreslení grafu připravíme jiné proměnné s mnohem menším počtem bodů:

```
bv1=(-0.2:0.2:2.4)';
bv2=(0:0.2:2.6)';
```

Pro obě dvojice proměnných b_1 , b_2 a bv_1 , bv_2 připravíme k úhlům i souřadnice x, y , zde jen b_1 :

```
x=L1*cos(b1)+L2*cos(b1+b2);
y=L1*sin(b1)+L2*sin(b1+b2);
```

Proměnné L_1 , L_2 reprezentují délky segmentů a musíme je nastavit předem.

Pole musíme spojit, knihovna `nftool` Matlabu pořadí pole vstupů a pole výstupů:

```
beta=[b1,b2];
xy=[x,y];
```

Alternativně můžeme vektor s úhly vygenerovat náhodně. Je to dokonce jednodušší, stačí zadat počet a pak přičíst minimální hodnotu (bývá záporná) a ponásobit rozsahem:

```
br1 = (rand(500,1)-0.2)*2.6;
br2 = rand(500,1)*2.6;
```

Kdyby rozsah obou úhlů byl stejný, šlo by pole hodnot vygenerovat v jednom kroku. Dále postupujeme stejně, vygenerujeme dosažené souřadnice, sloučíme do dvou polí a natrénujeme neuronovou sít.

Pro trénování sítě využijeme knihovnu `nftool`. Požadujeme, aby vstupem sítě byly rektangulární souřadnice, a výstupem úhly v kloubech manipulátoru, kterými jich lze dosáhnout. Takto musíme proměnné pro trénování zadat.

V dalších krocích zvolíme počet neuronů ve skryté vrstvě (osvědčil se mi velmi vysoký počet, například 25), a trénovací metodu. Nejlepší výsledky dává Bayesian regularization.

Na konci si necháme vygenerovat funkci pro Matlab, která pracuje s polem (matrix only function). Doporučuji přejmenovat (zde `nn2reg`) a uložit.

3.2 Zobrazení výsledků

Pro připravené testovací pole si necháme nově vygenerovanou neuronovou sítí spočítat úhly:

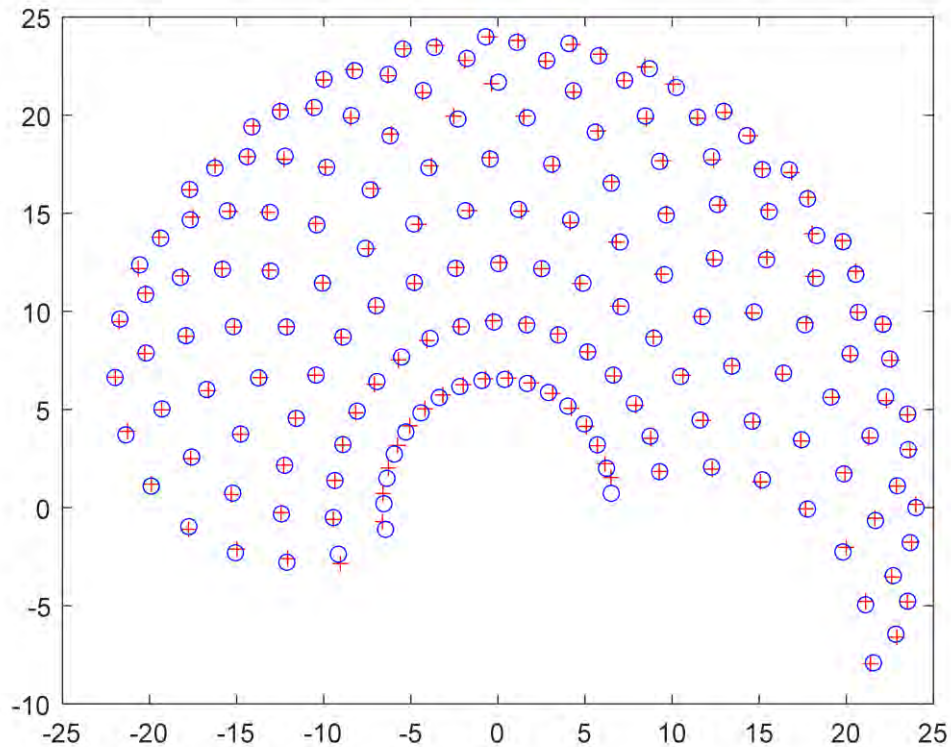
```
b = nn2reg(xy2);
```


Úhly opět dosadíme do přímé kinematiky

```
x=L1 * cos(b(:,1))+ L2*cos(b(:,1)+b(:,2));
y=L1 * sin(b(:,1))+ L2*sin(b(:,1)+b(:,2));
```

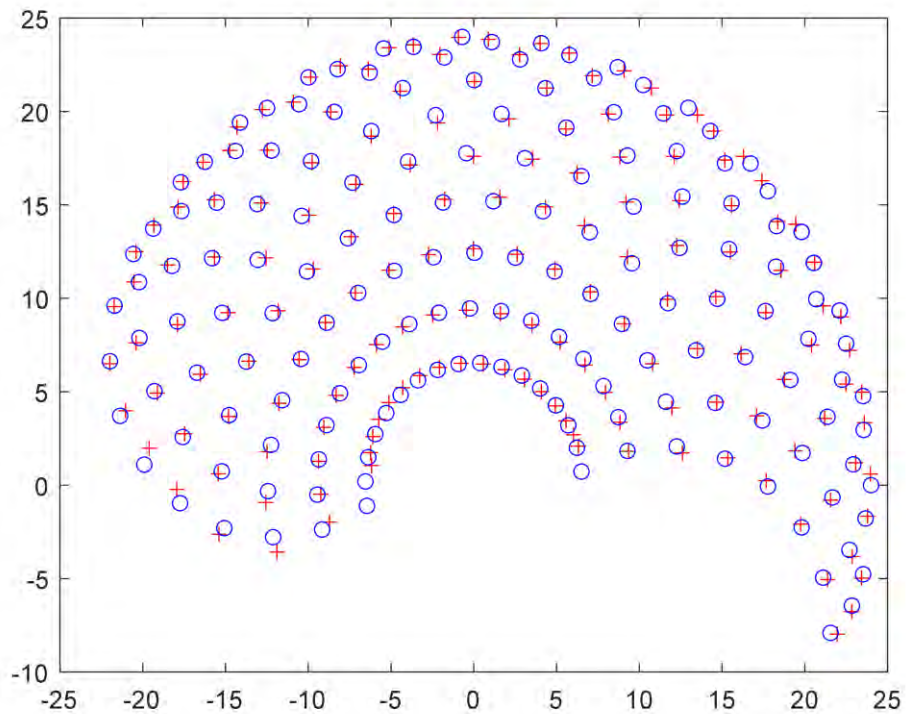
Zobrazíme v grafu. Zde jsou například kolečka jako cíle (správná hodnota) a křížky jako dosažená hodnota:

```
plot(x2,y2,'bo')
plot(x,y,'r+')
```



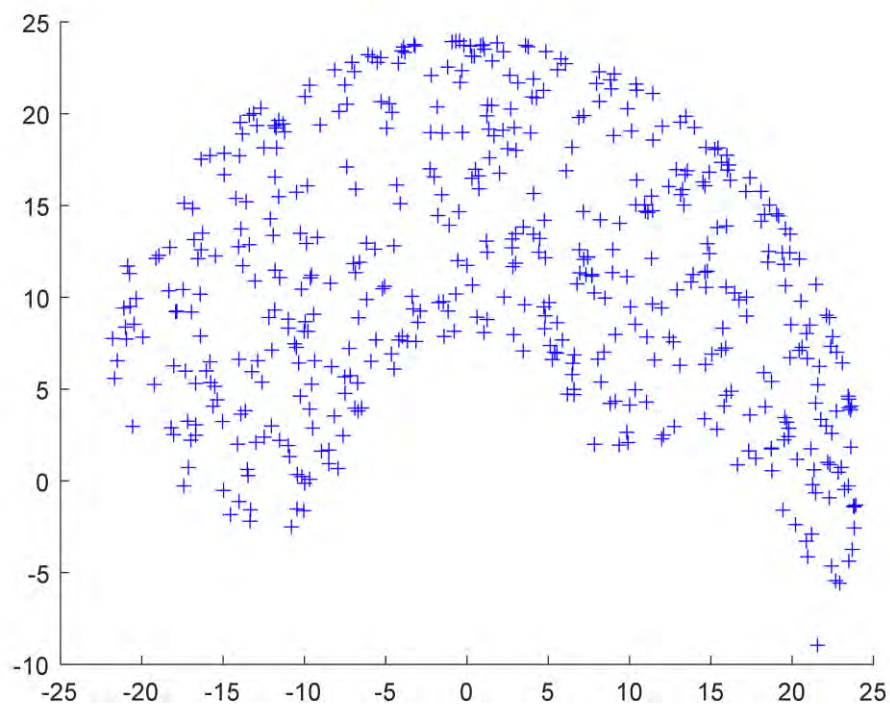
Obr. 3 Výsledek trénování. Modrá kolečka jsou zadaná data x, y , která se zadají jako vstup natrénované neuronové sítě, získané úhly se pomocí (přímé) kinematiky (1) převedou na souřadnice a ty se zobrazí červenými křížky. Délky segmentů byly 14 a 10.

Neuronová síť, jejíž výsledek je na obr. 3, měla 25 neuronů ve skryté vrstvě. Výsledek téhož experimentu, ale pro 15 neuronů:



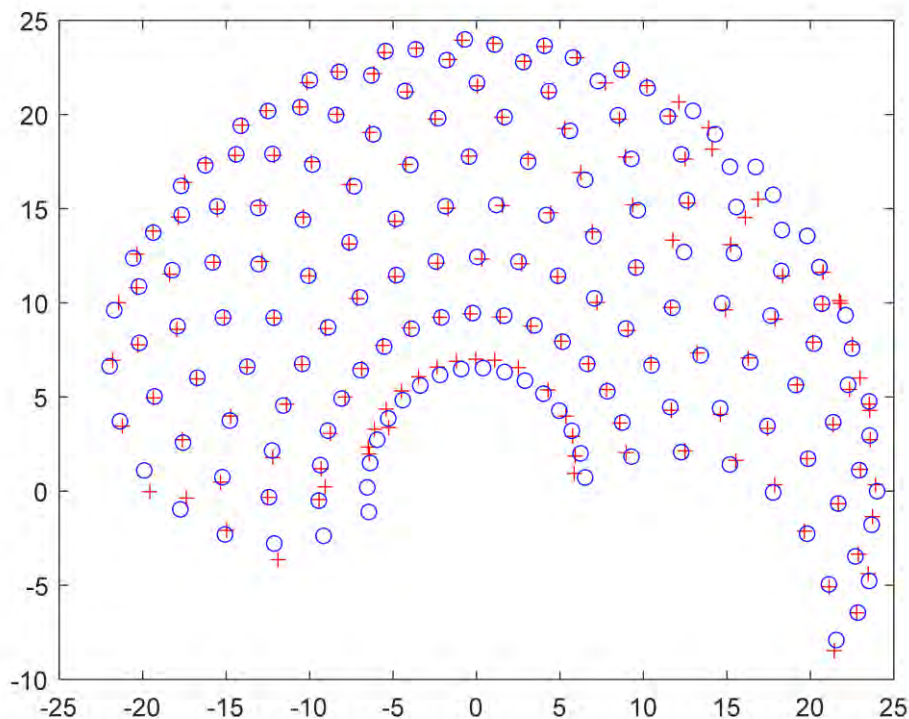
Obr. 4 Výsledek trénování. Stejná trénovací data, ale jen 15 neuronů ve skryté vrstvě.

Pokus byl zopakován s náhodně vygenerovanými daty (náhodně generované body jsou na následujícím obrázku):



Obr. 5 Pro tyto body bylo vytvořeno zadání pro trénování neuronové sítě.

Není ovšem důvod zobrazovat výsledek s náhodnými daty, takže i pro vizualizaci výsledku použijeme pravidelná data:



Obr. 6 Výsledek trénování při použití náhodně generovaných dat (natrénovaná síť byla použita pro stejné body, jako na obr. 3 a obr. 4 a výsledky vyneseny do grafu). Největší rozdíl je v pokrytí okrajových oblastí, kde je možné, že se vzorky pro trénování ani nevygenerovaly (zatímco pravidelně generovaná data obsahují i hraniční body na mezi dosahu manipulátoru).

4. Závěr

Vícevrstvá neuronová síť poskytuje jednoduché řešení inverzní kinematiky manipulátoru. Tuto metodu lze rozšířit nejen na 3D manipulátory, ale také na redundantní manipulátory. Možné řešení je popsáno v [5].

Literatura

- [1] M. Švejda, „Kinematika robotických architektur,“ 2011. [Online]. Available: http://home.zcu.cz/~msvejda/_publications/2011/rigo.pdf.
- [2] J. Čejka, „Rychlé heuristické metody numerického řešení úlohy inverzní kinematiky,“ 2019. [Online]. Available: https://dspace.tul.cz/bitstream/handle/15240/153383/Disertace_Cejka.pdf.
- [3] J. Shah, S.S. Rattan a B.C. Nakra, „Kinematic Analysis of 2-DOF Planer Robot Using Artificial Neural Network,“ *International Journal of Mechanical and Mechatronics Engineering*, sv. 5, č. 9, pp. 1720-1723, 2011.
- [4] J. Shah, S.S. Rattan a B.C. Nakra, „Kinematic Analysis of a Planer Robot Using Artificial Neural Network,“ *International Journal of Robotics and Automation (IJRA)*, sv. 1, č. 3, pp. 145-151, 2012.
- [5] V. Hlavac, „Neural Network for the identification of a functional dependence using data preselection,“ *Neural Network World*, sv. 2, pp. 109–124, 2021. Available: <http://nnw.cz/obsahy21.html>

FESTO MOTION TERMINAL VTEM CONTROLLING UNIVERSAL PRESSING FACILITY

Ladislav Červinka

České vysoké učení technické v Praze, Fakulta strojní, ladislav.cervinka@fs.cvut.cz

Abstrakt: Tato práce se zabývá programováním lisovacího přípravku pomocí ventilů VEVM - Festo Motion Terminálu (FMT) VTEM. Cílem je ukázka možného způsobu programování tohoto fenoménu v prostředí CoDeSys, a to pomocí jazyku ST (structured text). Řešen je tak CoDeSys projekt řízení lisovacího přípravku s využitím instrumentace pomocí technologie Festo Motion Terminal s možností zadávat různé výrobní sekvence, různé parametry výrobních sekvencí, různé výrobní dávky.

Klíčová slova: FTM, FESTO, MOTION, TERMINAL, CPX, VTEM, MECHATRONIKA, PNEUMATICKÉ SYSTÉMY, PRŮMYSL 4.0, MOTION APPS

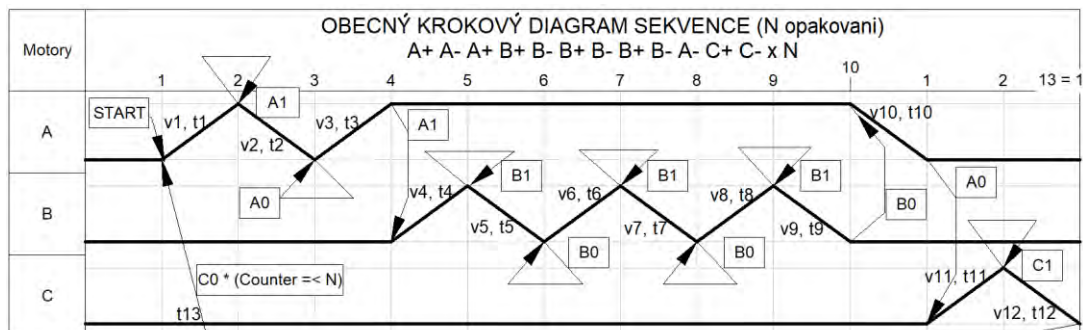
1. Úvod

V dnešní době jsou pneumatické systémy nedílnou součástí průmyslu, ale i běžného života člověka. Díky rychlému vývoji elektroniky a dalších nových technologií se moderní pneumatika pojí s výpočetní technikou. Vznikl tak koncept digitální pneumatiky, který spadá pod Průmysl 4.0. Firma Festo si pojmy jako mechatronika, průmysl 4.0, IoT (internet of things, resp. internet věcí) vzala za své a představila Festo Motion Terminal VTEM (FMT) a ventil VEVM. Jednotlivé ventily tohoto terminálu jsou řízeny aplikacemi funkcí (Motion Apps) a lze tak jediným ventilem nahradit funkcí až 50 standardních výrobků. V této práci je věnována pozornost programování tohoto fenoménu v prostředí CoDeSys.

2. Lisovací přípravek

2.1 Popis úlohy

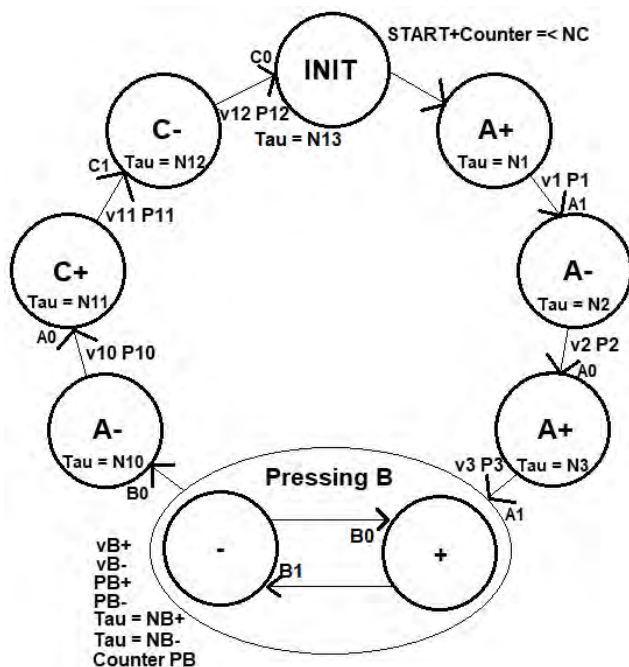
Podle zadání má být po stisknutí tlačítka START postupně zalisováno N obrobků. Raznice upevněná na pneumotoru A jedním dlouhým a krátkým úderem provede zalisování polotovaru. Následně, kdy zůstane motor A ve vysunuté poloze, nalisuje pneumotor B z boku pojistný kolík do obrobku. Nalisování pneumotorem B je provedeno jedním pomalým a dvěma rychlými údery. Nakonec pneumotor C (vyhazovač) vysune hotový kus z lisovacího ukotvení a výroba může pokračovat do dalšího cyklu [1].



Obr. 1 – Krokový diagram úlohy

2.2 Návrh projektu FMT

Celý projekt byl zpracován bez fyzického přístupu k FMT. Proto není možné program odzkoušet a ladit. Algoritmus a jednotlivé kroky byly navrženy pomocí větvení CASE - OF a podmínek IF - THEN. Základní sekvence byla navržena podle obecného krokového diagramu na obr. 1, a to s tím rozdílem, že bylo přidáno cyklování lisování pneumotorem B a díky funkcím FMT bylo přidáno ovládání nejen rychlostí pneumotorů a časových prodlev, ale také tlaků [3]. Stavový diagram je pak v podobě následujícího obr. 2.



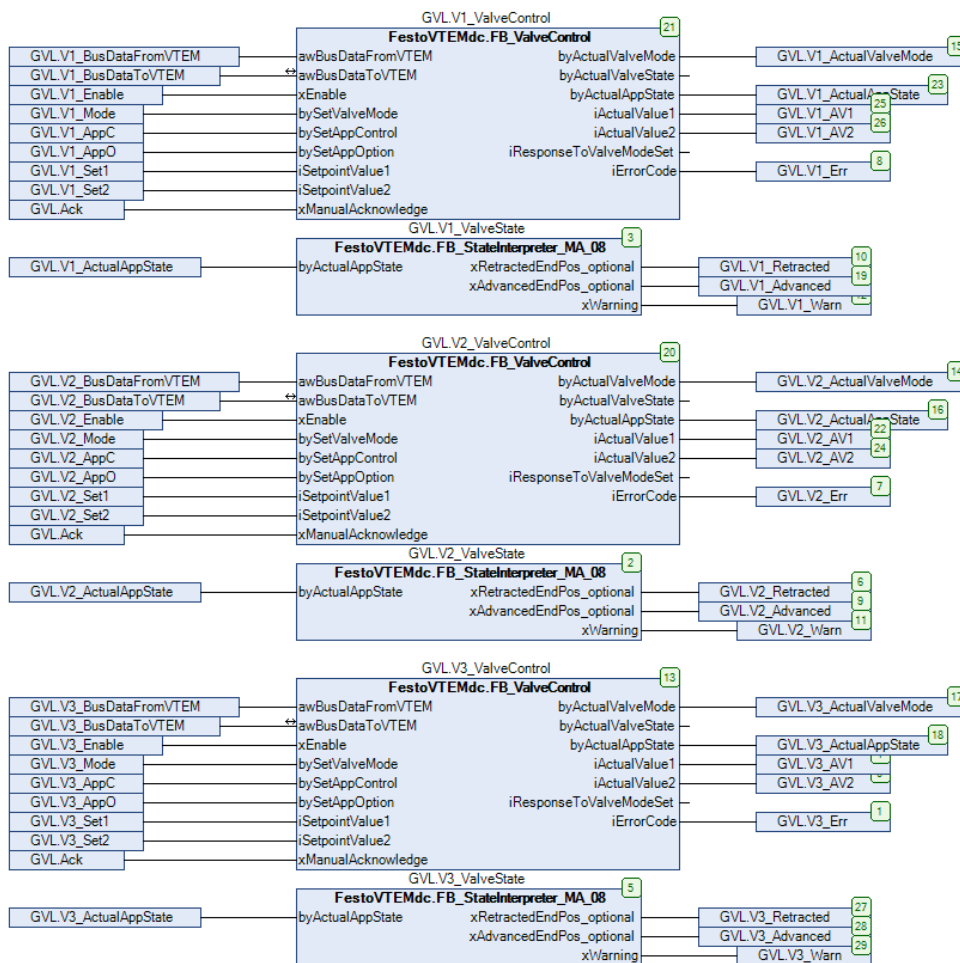
Obr. 2 – Stavový diagram úlohy

Tab. 1 – Popis označení krokového a stavového diagramu

OZNAČENÍ	POPIS
INIT	Inicializace
A+, B+, C+	Akce pneumotorů vysunout
A-, B-, C-	Akce pneumotorů zasunout
vX	Rychlost pístu
PX	Tlak vzduchu
Tau, tx	Časová prodleva
NX	Délka časové prodlevy
Counter	Čítač
N, NC	Limit čítače
START	Startovací tlačítko

Nejprve bylo zapotřebí v rámci projektu zvolit typ řídicího modulu CPX-CEC a programovací jazyk, ve kterém bude program pro PLC založen. Zařízeno bylo zvoleno CPX-CEC-S1-V3 a programovací jazyk strukturovaného textu. Do správce knihoven byla přidána knihovna „Festo_VTEM_DevCon“, která je nutná pro návrh celého projektu [2].

Aby bylo možné začít s programováním, bylo nejdřív nutné vytvořit programovou organizační jednotku (POU) a v ní založit funkční bloky (FB) pro MotionApp č.8, pomocí kterých se řídí všechny 3 pneumotory, resp. ventily [2]. Tyto bloky je možné vytvořit pomocí funkce „Input Asistant“, a to díky přidané zmíněné knihovně. V následujícím obr. 3 jsou funkční bloky „FB_ValveControl“ a „FB_StateInterpreter_MA_08“. Vstupy a výstupy těchto bloků jsou navázány na globální proměnné, se kterými funkční bloky pracují.



Obr. 3 – Zapojení funkčních bloků

2.3 Návrh programu FMT

Hlavní část programu je složená z větvení pomocí CASE – OF a IF – THEN, tedy skoku mezi jednotlivými číslovanými kroky a vykonávání akcí po splnění zadaných podmínek. Na začátku programu se nachází deklarace časovačů a čítačů. Na začátku programu, tj. před částí, která je vytvořená pomocí CASE – OF, se také nachází ovládání tlačítka ON ve vizualizaci programu, stejně jako krátké větvení pro ovládání prvku „spin control“. Typicky se v praxi program pro HMI nebo např. také pro alarmy nachází v jiné části projektu, tedy mimo hlavní sekvencér, ovšem v tomto případě je program pro HMI krátký, a proto bylo souzeno, že celý program bude v jediném celku.

Program začíná v „IDLE“ stavu, ve kterém čeká na přepnutí BOOL vstup „ON_OFF“ z HMI. Po splnění této podmínky se v HMI zobrazí zbylé ovládací prvky HMI.

Pro HMI prvek „spin control“, který ovládá výběr kroku, ve kterém jsou nastavovány parametry rychlost (INT), tlak (INT) a čas (REAL), byl vytvořen „struct“. V tomto prvku je vytvořeno 8 dalších prvků, resp. čísel v poli (array [1..8]), což je počet kroků, ve kterých jsou měněny zmíněné parametry. Nastavená hodnota (INT) v HMI prvku „spin control“ (dále jako „výběr kroku“) se porovnává s aktuální hodnotou v bufferu a pokud se nerovná, hodnota z výběru kroku se do bufferu nahraje. Poté se hodnoty v proměnných zmíněného structu uloží do HMI proměnných, tak se v HMI tyto hodnoty zobrazí. Poslední větev IF zajišťuje ukládání z HMI do hodnot pole structu, a tím tak ukládá parametry všech kroků, ve kterých se prvky structu nachází.

Dále následují větvení pomocí CASE – OF [4]. V nulovém kroku na začátku dochází k resetování / odstranění chybových hlášek ventilů. Následují vstupy pro inicializaci stavů ventilů, čítačů a parametrů ventilů, dále vstupy pro

zapnutí automatického režimu a režimu manuálního ovládání ventilů. Pro zapnutí jednoho nebo druhého režimu je nutné pomocí HMI spustit inicializaci.

V algoritmu sekvenceru automatického režimu se vždy opakují dva kroky pro vykonání jedné celé akce pneumotoru (např. vysunutí pístu). První krok, po splnění IF podmínky (inicializovaný stav, nebo sepnutí koncového spínače z akce předešlého kroku a případně doběhnutí časovače), slouží v této posloupnosti k provedení: resetu časovače, nastavení parametrů rychlosti, tlaku a časové prodlevy po vykonání kroku, vykonání akce pneumotoru (vysunout píst / zasunout píst) a skok do dalšího kroku. Druhý krok slouží pro vykonání časové prodlevy po vykonání akcí z předešlého kroku. Pro lepší přehlednost jsou tyto dva kroky oddělené, avšak pro funkčnost to není nutné (počet větví IF – THEN není nijak omezený ve větvích CASE – OF)[5]. Na konci sekvenceru je čítač, pomocí kterého je řešené opakování celé sekvence. Počet cyklů se nastavuje v HMI a do výše tohoto čísla se postupně inkrementuje hodnota po každém dokončeném cyklu hlavní sekvence. Pokud je čítač ve své maximální přednastavené hodnotě, tzn. uběhl daný počet cyklů sekvence, skočí program do nulového kroku a tím je celý cyklus u konce. Akce pneumotoru B jsou stejným způsobem zacyklené od 40. až do 60. kroku. Počet lisovacích akcí pneumotorem B je možné v HMI nastavit stejným způsobem, jako počet cyklů celé sekvence. Parametry pro cyklus lisování pneumotorem B jsou také parametrizovatelné, ovšem jsou stejné pro celý cyklus.

```

1  POU( );
2  // časovače, čítače a default stavy
3  Casovac (IN:= Cas_ON, PT:= REAL_TO_TIME(Cas_kroku*1000),
4  Q=> Cas_ubehl , ET=> );
5  Citac_Cyklu(
6  CU:= (Cyklus_sekvence),
7  RESET:=(Reset_citac_sekvence) ,
8  PV:= Cyklus_CNT ,
9  Q => Sekvence_vykonana ,
10 CV=> HMI_CNT_Cyklu_aktualni);
11 Citac_B(
12 CU:= (Cyklus_b_dokoncen),
13 RESET:=(Reset_citac_b) ,
14 PV:= B_POCETCYKLU_CNT ,
15 Q => B_vykonano ,
16 CV=> HMI_CNT_B_aktualni);
17
18 IF spinindex <> spinbuffer THEN
19 load :=1;
20 spinbuffer := spinindex;
21 END_IF;
22
23 IF load = 1 THEN
24 GVL.HMI_RYCHLOST := GVL.HMI_CONTROL[SpinIndex].Rychlost;
25 GVL.HMI_CAS := GVL.HMI_CONTROL[SpinIndex].Cas;
26 GVL.HMI_TLAK := GVL.HMI_CONTROL[SpinIndex].Tlak;
27 load := 0;
28 END_IF;
29
30 IF hmi_save = 1 THEN
31 GVL.HMI_CONTROL[SpinIndex].Tlak := GVL.HMI_TLAK;
32 GVL.HMI_CONTROL[SpinIndex].Cas := GVL.HMI_CAS;
33 GVL.HMI_CONTROL[SpinIndex].Rychlost := GVL.HMI_RYCHLOST;
34 END_IF;
35
36 IF ON_OFF THEN
37 Viditelnost_HMI := TRUE;
38 GVL.V1_Enable := TRUE;
39 GVL.V2_Enable := TRUE;
40 GVL.V3_Enable := TRUE;
41 ELSE
42 GVL.V1_Enable := FALSE ;
43 GVL.V2_Enable := FALSE ;
44 GVL.V3_Enable := FALSE ;
45 Viditelnost_HMI := FALSE;
46 INIT_DONE := 0;
47 MANUAL := FALSE;
48 AutoMod := FALSE;
49 END_IF
50
51 IF (GVL.V1_ActualValveMode = 61 OR GVL.V2_ActualValveMode = 61
52 OR GVL.V3_ActualValveMode = 61) THEN
53 GVL.xMan := TRUE;
54 END_IF
55
56 IF INIT_SET THEN
57 Cyklus_sekvence := 0 ; //Nulování vstupu CU na čítači
58 Cyklus_b_dokoncen := 0 ; //Nulování vstupu CU na čítači
59 GVL.V1_Mode := 8; //MotionApp 8
60 GVL.V2_Mode := 8;
61 GVL.V3_Mode := 8;
62 GVL.V1_AppC := 2; //Zasunutí motoru
63 GVL.V2_AppC := 2;
64 GVL.V3_AppC := 2;
65 Reset_citac_sekvence := 1;
66 Reset_citac_b := 1;
67 GVL.V1_Set1 := 10000 ;
68 GVL.V1_Set2 := 6000 ;
69 GVL.V2_Set1 := 10000 ;
70 GVL.V2_Set2 := 6000 ;
71 GVL.V3_Set1 := 10000 ;
72 GVL.V3_Set2 := 6000 ;
73 INIT_DONE := 1;
74
75 Cyklus_CNT := HMI_pocet_cyklu_sekvence;
76 B_POCETCYKLU_CNT := HMI_pocet_cyklu_b;
77 Krok := 0;
78 END_IF
79
80 IF AutoMod THEN
81 MANUAL := 0;
82 AutoMod_STATE := 1;
83 Krok := 0;
84 END_IF
85
86 IF AutoMod_STATE AND Start AND GVL.V1_Retracted AND GVL.V2_Retracted
87 AND GVL.V3_Retracted AND INIT_DONE THEN
88 Reset_citac_sekvence := 0;
89 Reset_citac_b := 0;
90 Krok := 10 ;
91 END_IF

```



```

91 10:
92     IF GVL.V1_Retracted THEN
93         Cas_ON := 0 ; //Reset časovače
94         Reset_citac_buchar := FALSE ; //Shození resetu čítače
95         GVL.V1_Set1 := GVL.HMI_CONTROL[1].Rychlost*100 ;
96 //Nastavení rychlosti pneumotoru V1 z HMI pro krok 1
97         GVL.V1_Set2 := GVL.HMI_CONTROL[1].Tlak ;
98 //Nastavení tlaku pro krok1
99         Cas_kroku := GVL.HMI_CONTROL[1].Cas ;
100 //Nastavení časovače z HMI po kroku 1
101         GVL.V1_AppC := 1 ; //Pneumotor 1 vysunout
102         Krok := 15 ;
103     END_IF
104 15:
105     IF GVL.V1_Advanced THEN
106         Cas_ON := 1 ;
107         Krok := 20 ;
108     END_IF
109 20:
110     IF (GVL.V1_Advanced AND Cas_ubehl) THEN
111         Cas_ON := 0 ; //Reset časovače
112         GVL.V1_Set1 := GVL.HMI_CONTROL[2].Rychlost*100
113 ; //Nastavení rychlosti pneumotoru V1 z HMI pro krok 2
114         GVL.V1_Set2 := GVL.HMI_CONTROL[2].Tlak ;
115 //Nastavení tlaku pro krok2
116         Cas_kroku := GVL.HMI_CONTROL[2].Cas ;
117 //Nastavení časovače z HMI po kroku 2
118         GVL.V1_AppC := 4 ; //Pneumotor 1 zasunout
119         Krok := 25 ;
120     END_IF
121 25:
122     IF GVL.V1_Retracted THEN
123         Cas_ON := 1 ;
124         Krok := 30 ;
125     END_IF
126 30:
127     IF (GVL.V1_Retracted AND Cas_ubehl) THEN
128         Cas_ON := 0 ; //Reset časovače
129         GVL.V1_Set1 := GVL.HMI_CONTROL[3].Rychlost*100 ;
130 //Nastavení rychlosti pneumotoru V1 z HMI pro krok 3
131         GVL.V1_Set2 := GVL.HMI_CONTROL[3].Tlak ;
132 //Nastavení tlaku pro krok3
133         Cas_kroku := GVL.HMI_CONTROL[3].Cas ;
134 //Nastavení časovače z HMI po kroku 3
135         GVL.V1_AppC := 1 ; //Pneumotor 1 vysunout
136         Krok := 35 ;
137     END_IF
138 35:
139     IF GVL.V1_Advanced THEN
140         Cas_ON := 1 ;
141         Krok := 40 ;
142     END_IF
143 40:
144     IF GVL.V2_Retracted AND Cas_ubehl THEN
145         Cas_ON := 0 ; //Reset časovače
146         Cyklus_b_dokoncen := 0 ; //Reset vstupu čítače b
147         GVL.V2_Set1 := GVL.HMI_CONTROL[4].Rychlost*100 ;
148 //Nastavení rychlosti pneumotoru V2 z HMI pro aktuální krok
149         GVL.V2_Set2 := GVL.HMI_CONTROL[4].Tlak ;
150 //Nastavení tlaku pro krok
151         Cas_kroku := GVL.HMI_CONTROL[4].Cas ;
152 //Nastavení časovače z HMI po kroku
153         GVL.V2_AppC := 1 ;
154         Krok := 45 ;
155     END_IF
156 45:
157     IF GVL.V2_Advanced THEN
158         Cas_ON := 1 ;
159         Krok := 50 ;
160     END_IF
161 50:
162     IF GVL.V2_Advanced AND Cas_ubehl THEN
163         Cas_ON := 0 ;
164         GVL.V2_Set1 := GVL.HMI_CONTROL[5].Rychlost*100 ;
165 //Nastavení rychlosti pneumotoru V2 z HMI pro aktuální krok
166         GVL.V2_Set2 := GVL.HMI_CONTROL[5].Tlak ;
167 //Nastavení tlaku pro krok
168         Cas_kroku := GVL.HMI_CONTROL[5].Cas ;
169 //Nastavení časovače z HMI po kroku
170         GVL.V2_AppC := 4 ;
171         Krok := 55 ;
172     END_IF
173 55:
174     IF GVL.V2_Retracted THEN
175         Cas_ON := 1 ;
176         Cyklus_b_dokoncen := 1 ;
177         Krok := 60 ;
178     END_IF
179 60:
180     IF GVL.V2_Retracted AND Cas_ubehl AND B_vykonano = 0 THEN
181         Krok := 40 ;
182     END_IF
183     IF GVL.V2_Retracted AND Cas_ubehl AND B_vykonano = 1 THEN
184 // 10. krok dle obecného krokového diagramu
185         Cas_ON := 0 ;
186         Reset_citac_b := TRUE ;
187         GVL.V1_Set1 := GVL.HMI_CONTROL[6].Rychlost*100 ;
188 //Nastavení rychlosti pneumotoru V1 z HMI pro aktuální krok
189         GVL.V1_Set2 := GVL.HMI_CONTROL[6].Tlak ;
190 //Nastavení tlaku pro aktuální krok
191         Cas_kroku := GVL.HMI_CONTROL[6].Cas ;
192 //Nastavení časovače z HMI po aktuálním kroku
193         GVL.V1_AppC := 4 ;
194         Krok := 105 ;
195     END_IF
196 105:
197     IF GVL.V1_Retracted THEN
198         Cas_ON := 1 ;
199         Krok := 110 ;
200     END_IF
201 110:
202     IF GVL.V1_Retracted AND Cas_ubehl THEN
203         Cas_ON := 0 ;
204         GVL.V3_Set1 := GVL.HMI_CONTROL[7].Rychlost*100 ;
205 //Nastavení rychlosti pneumotoru V3 z HMI pro aktuální krok
206         GVL.V3_Set2 := GVL.HMI_CONTROL[7].Tlak ;
207 //Nastavení tlaku pro aktuální krok
208         Cas_kroku := GVL.HMI_CONTROL[7].Cas ;
209 //Nastavení časovače z HMI po aktuálním kroku
210         GVL.V3_AppC := 1 ;
211         Krok := 115 ;
212     END_IF
213 115:
214     IF GVL.V3_Advanced THEN
215         Cas_ON := 1 ;
216         Krok := 120 ;
217     END_IF
218 120:
219     IF GVL.V3_Advanced AND Cas_ubehl THEN
220         Cas_ON := 0 ;
221         GVL.V3_Set1 := GVL.HMI_CONTROL[8].Rychlost*100 ;
222 //Nastavení rychlosti pneumotoru V3 z HMI pro aktuální krok
223         GVL.V3_Set2 := GVL.HMI_CONTROL[8].Tlak ;
224 //Nastavení tlaku pro aktuální krok
225         Cas_kroku := GVL.HMI_CONTROL[8].Cas ;
226 //Nastavení časovače z HMI po aktuálním kroku
227         GVL.V3_AppC := 1 ;
228         Krok := 125 ;
229     END_IF
230 125:
231     IF GVL.V3_Retracted THEN
232         Cas_ON := 1 ;
233         Cyklus_sekvence := 0 ;
234         Cyklus_sekvence := 1 ;
235         Krok := 130 ;
236     END_IF

```

```

237 130:
238 IF GVL.V3_Retracted AND Cas_ubehl AND Sekvence_vykonana = 0 THEN
239     Krok := 10 ;
240 END_IF
241 IF GVL.V3_Retracted AND Cas_ubehl AND Sekvence_vykonana = 1 THEN
242     HMI_sekvence_dokoncena := Sekvence_vykonana;
243     Cas_ON := 0 ;
244     Reset_citac_sekvence := TRUE ;
245     Krok := 0 ;
246 END_IF
247 END_CASE

```

Obr. 6 – Třetí část ST programu

3. Závěr

Byl vytvořen projekt v prostředí CoDeSys pro ovládání 3 pneumotorů pomocí Festo Motion Terminálu VTEM - VEVN ventilů zahrnující: volbu aplikace funkcí „Motion Apps“ č.8 s názvem „volitelná úroveň tlaku“, která dovoluje parametrizovat úroveň tlaku v milibarech a otevření větrání ventilu v procentech - pro řízení rychlosti pístů pneumotorů. Dále je v projektu vybrán řídicí modul CPC-CEC-S1-V3 a programovací jazyk ST (strukturovaný text). Byla přidána knihovna „Festo_VTEM_DevCon“ pro následné programování zmíněného hardwaru.

V rámci projektu byla vytvořena programová organizační jednotka, ve které se nachází funkční bloky přidané ze zmíněné knihovny. Pro ovládání jednoho VEVN ventilu, v rámci aplikace funkcí č.8, byly přidány vždy dva funkční bloky (tzn. celkově 6 bloků), pracující s globálními proměnnými, které jsou následně přímo ovlivňovány programem a ovládáním z HMI. Byl vytvořen návrh řídicího algoritmu VEVN ventilů pro ovládání 3 pneumotorů. Toho bylo docíleno pomocí větvení CASE – OF a podmínek IF – THEN. Jednotlivé kroky sekvence lze parametrizovat (rychlost pístu pneumotoru, tlak a časová prodleva po vykonání kroku). Celou sekvenci je možné cyklit pomocí čítače. Akce pneumotorem B je možné uvnitř každého jednoho cyklu také cyklit pomocí druhého čítače. Všechny tyto parametry, tedy rychlost, tlak, čas a oba čítače, lze měnit ve vizualizaci HMI.

Byl tak vytvořen projekt řízení lisovacího přípravku s využitím instrumentace pomocí technologie Festo Motion Terminal s možností zadávat různé výrobní sekvence, různé parametry výrobních sekvencí, různé výrobní dávky

Na tento projekt navazuje vytvoření vizualizace. Pomocí vizualizace HMI by bylo možné ovládat inicializaci, start sekvence a měnit zmíněné parametry v jednotlivých krocích. V takové vizualizaci by také bylo možné sledovat a ukládat provozní data pro jejich vyhodnocení.

ZDROJE

- [1] *ZADANI_RA_E_V2.pdf* [online]. [vid. 2021-07-26]. Dostupné z: https://moodle-vyuka.cvut.cz/pluginfile.php/153425/mod_resource/content/1/ZADANI_RA_E_V2.pdf
- [2] *Koubová - DIDAKTICKÉ ÚLOHY PRO DIGITÁLNÍ PNEUMATIKU S VYUŽITÍM.pdf* [online]. [vid. 2021-07-11]. Dostupné z: https://dspace.cvut.cz/bitstream/handle/10467/90334/F2-DP-2020-Koubova-Jana-DP_Koubova_Jana_2020.pdf?sequence=-1&isAllowed=y
- [3] *Digitalizovaný pneumatický systém | Festo Motion Terminal* [online]. [vid. 2020-12-27]. Dostupné z: <https://www.festo.com/vtem/cs/cms/10169.htm>
- [4] *F3-BP-2017-Hanak-Miroslav-Advanced PLC Programming Methods.pdf* [online]. [vid. 2021-08-16]. Dostupné z: https://dspace.cvut.cz/bitstream/handle/10467/68419/F3-BP-2017-Hanak-Miroslav-Advanced_PLC_Programming_Methods.pdf
- [5] *Structured Text - an overview | ScienceDirect Topics* [online]. [vid. 2021-08-16]. Dostupné z: <https://www.sciencedirect.com/topics/computer-science/structured-text>

MOŽNOSTI REALIZACE INTERFERENČNÍHO SPEKTROMETRU

Pavel Blažek

¹ *Ústav přístrojové a řídicí techniky, ČVUT v Praze, pavel.blazek@fs.cvut.cz*

Abstrakt: V článku je popsáno pokračování na návrhu interferenčního spektrometru s Fourierovou transformací s ohledem na hysteretní chování piezoposuvu pohyblivého zrcátka v Michelsonově interferometru. Práce začíná stručným teoretickým úvodem do problematiky interferenčního spektrometru s Fourierovou transformací a rekapitulací výsledků, kterých bylo dosaženo v předešlém projektu. Následují kapitoly praktické části, která začíná popisem korekčního algoritmu kompenzující vliv hysteretního chování piezoposuvu pohyblivého zrcátka. Vyřešení tohoto problému bylo nezbytné pro správnou funkci navrhovaného spektrometru. Po vyzkoušení funkčnosti algoritmu následuje popis návrhu spektrometru a jeho finální realizace. Na závěr je ověřena funkčnost spektrometru a zhodnocení dosažených výsledků.

Klíčová slova: Interferenční spektrometr, Fourierova transformace, Michelsonův interferometr, hysterese, piezoaktuátor, autokorelační funkce

Abstract: The article describes the continuation of the design of Fourier Transform Spectrometer with respect to the behavior of the piezo mirror-shifter in Michelson interferometer. The work begins with a brief theoretical introduction to the issue of interference spectrometer and recapitulation of the results that were in the previous project. The chapters of the practical part follow, which begins with a description of the correction algorithm, which compensates for the influence of the piezo mirror-shifter. The solution of this problem was necessary for the correct function of the proposed spectrometer. After testing the functionality of the algorithm, a description of the spectrometer design and its final implementation follows. Finally, the functionality of the spectrometer and the evaluation of the achieved results are verified.

Keywords: Interference spectrometer, Fourier transform, Michelson interferometer, hysteresis, piezoactuator, autocorrelation function

1 Úvod

Tato práce je věnována problematice spektrometru, který spojuje poznatky z optiky i matematiky. Interferenční spektrometr s Fourierovou transformací vznikl aplikací Fourierovy transformace, která je považována za jeden z nejdůležitějších algoritmů v oblasti zpracování signálu, na interferometrii. Vzniklo tak zařízení, jenž našlo své uplatnění zejména v oblasti chemie, optiky a astronomie a v řadě vlastností jako je například rychlost a rozlišení předčí konvenční disperzní spektrometry. Dnešní plně automatizované interferenční spektrometry dokonce umožňují práci i lidem, kteří nemají v oblasti Fourierovy transformace a optiky žádné znalosti. Práce popisuje návrh interferenčního spektrometru s Fourierovou transformací, který by byl realizovatelný ve školní laboratoři. Velká část práce je věnována návrhu korekčního algoritmu, který kompenzuje vliv hysteretního chování piezoposuvu pohyblivého zrcátka v Michelsonově interferometru, který je součástí navrhovaného spektrometru.

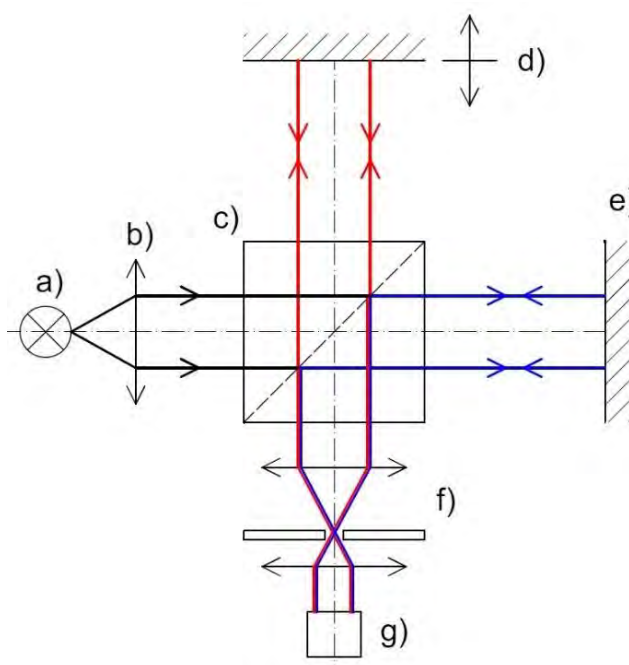
Tento článek popisuje práci v rámci školního projektu, nicméně toto téma se stalo i tématem méj diplomové práce [6] pod vedením doc. Ing. Jana Hoška, Ph.D. Článek je tedy primárně zkrácenou verzí části méj diplomové

práce.

2 Interferenční spektrometr s Fourierovou transformací

Interferenční spektrometr s Fourierovou transformací (zkráceně FTS z angl. Fourier transform spectrometry) je spektrometr, který dokáže analyzovat spektrum ve vysokém rozlišení a v širokém rozsahu. Jeho předností je také vysoká optická účinnost. Na jedno skenování pojme spektrální data z velmi širokého spektrálního rozsahu. Díky jeho přesnosti se používá ke kalibraci méně přesných typů spektrometrů [1]. Výhodou také je, že lze jeho parametry měnit. Jádrem interferenčního spektrometru je Michelsonův interferometr využívající jev zvaný interference.

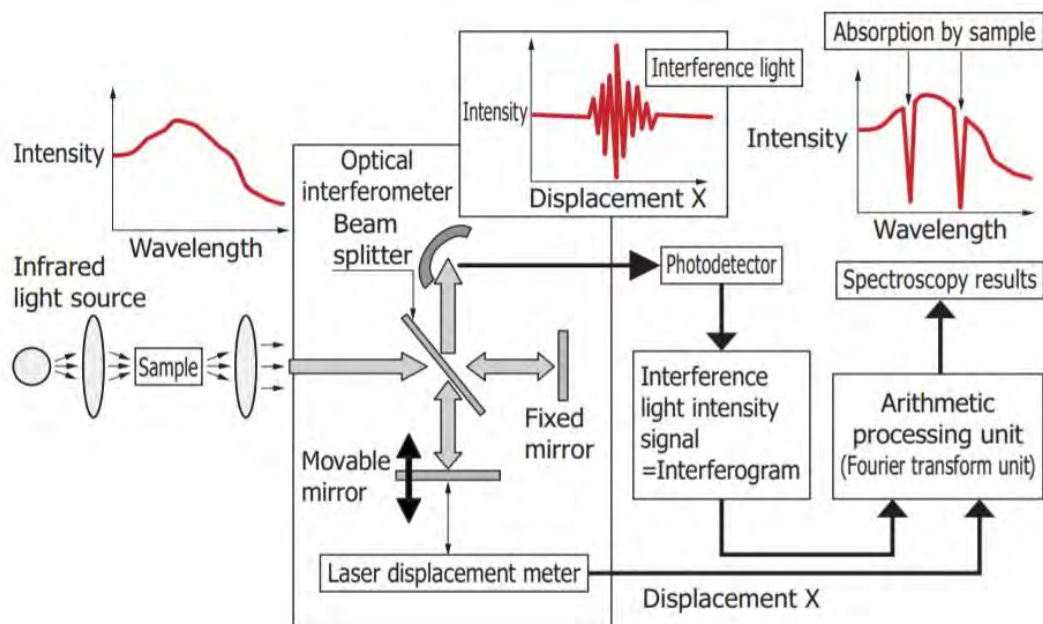
Interference světla je jev, který nastává v případě, kdy se dostanou do zákrytu dva nebo více světelných koherentních svazků. Výchylka složeného signálu je dána superpozicí jednotlivých svazků, čímž se myslí algebraický součet jejich okamžitých výchylek. Výsledná výchylka je závislá na fázovém posuvu jednotlivých skládaných vln [2]. Pokud je fázový posuv δ vůči druhému signálu nulový, tak dochází ke konstruktivní interferenci a získáme maximálně možné zesílení výsledné vlny. Pokud je naopak posuv roven půlperiodě, tak dochází k plně destruktivní interferenci a signál se zeslabí. V případě stejných amplitud obou signálů by měl výsledný signál při destruktivní interferenci nulovou amplitudu. Při posuvu menším než je půlperioda dochází k částečné interferenci.



Obr. 1 Michelsonův interferometr

Michelsonův interferometr měří rozdíl optických drah. Jeho schéma je na obr. 1. Kolimátor b) vytváří rovinnou vlnu ze záření zdroje a), která vstupuje do optického děliče c). Ten dělí paprsek amplitudově v poměru 50:50. Světelný modrý svazek odražený od referenčního (nehybného) zrcátka e) je označen jako referenční a červený svazek, odražený od pohyblivého zrcátka d) je označen jako měřicí. Oba svazky jsou děličem opět sloučeny a interferují spolu. Podle fáze, kterou realizuje pohyblivé zrcátko zaznamenáme na detektoru g) výslednou intenzitu složených svazků.

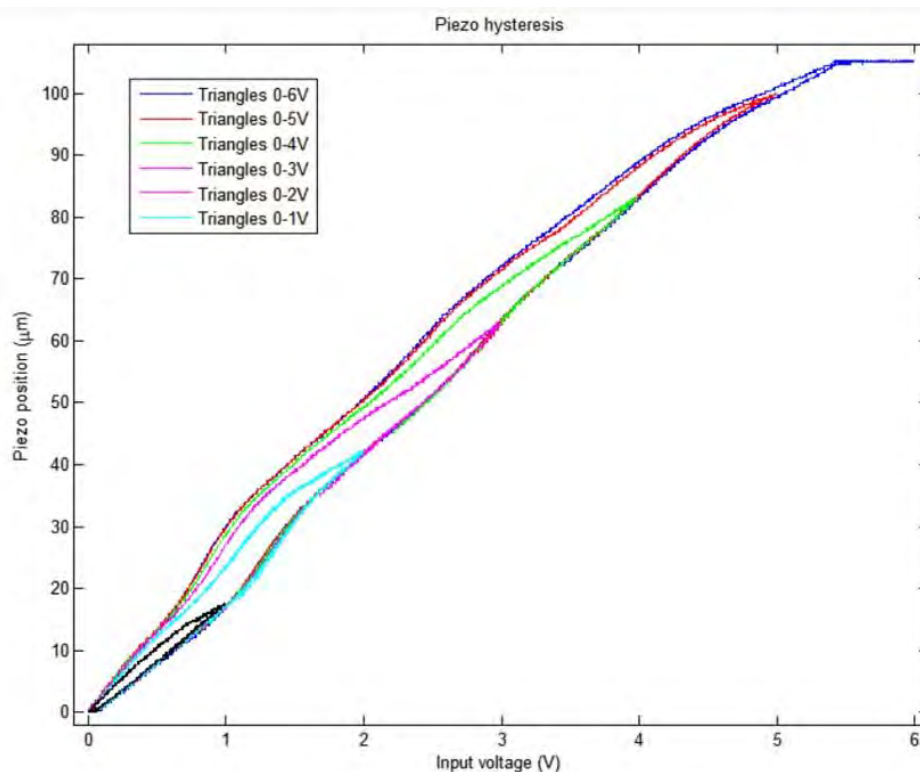
Interferenční spektrometr funguje tak, že přivedeme záření do Michelsonova interferometru a posouváme lineárně pohyblivým zrcátkem, čímž se mění fázový rozdíl referenční a měřicí světelné vlny. Na výstupu detektoru získáme tzv. autokorelační funkci (korelogram), což je průběh intenzity interferovaného záření v závislosti na fázovém rozdílu interferujících vln [3]. Tato autokorelační funkce v podobě diskretních dat ze senzoru se následně zpracuje algoritmem Fourierovy transformace (nejčastěji se používá Rychlá Fourierova transformace FFT), čímž získáme hledané spektrální rozdělení světla na vstupu interferometru.



Obr. 2 Princip funkce interferenčního spektrometru s Fourierovou transformací [4]

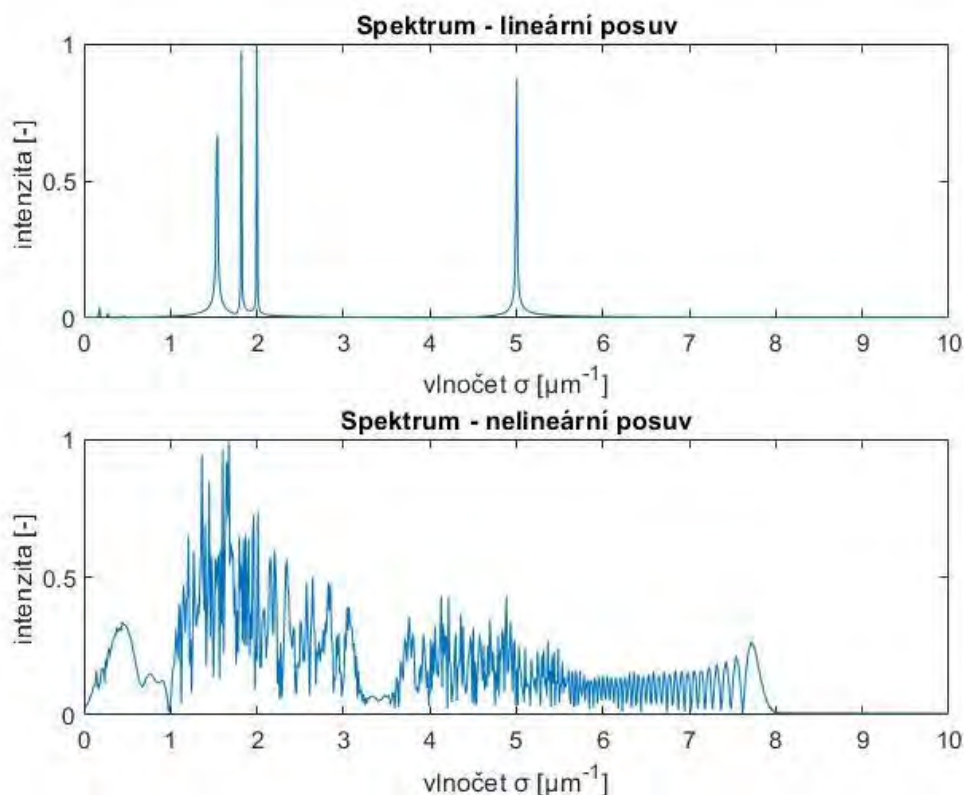
3 Rekapitulace výsledků dosažených v minulosti

V rámci projektu II v roce 2020 jsem započal práci na návrhu interferenčního spektrometru. Motivací bylo nalezení problému hysteretního chování piezoposuvu pohyblivého zrcátka v Michelsonově interferometru, který byl součástí spektrometru, který sestavil můj vedoucí práce. Na obr. 3 je znázorněno chování použitého piezostolku PX 100 od firmy piezosystem jena. Závislost posuvu na napájecím napětí je silně nelineární. Dalším problémem byl fakt, že hysteretní křivky se s časem měnily.



Obr. 3: Hysteretní chování piezostolku PX 100

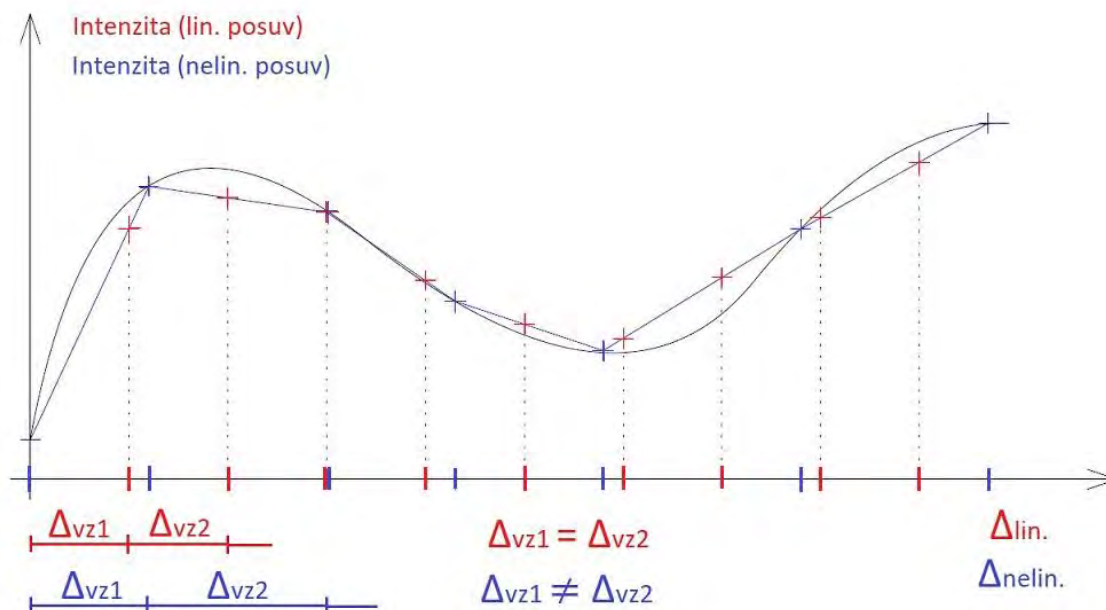
Mým úkolem bylo zjistit, do jaké míry toto chování ovlivňuje naměřené spektrum. Využil jsem prostředí MATLAB, kde jsem si vytvořil simulaci daného problému. Všiml jsem si, že data autokorelační funkce naměřená při lineárním posuvu a nelineárním posuvu se výrazně neliší. Data odpovídající nelineárnímu posuvu pouze získávám ve špatný okamžik. Tento předpoklad mi pomohl k realizaci funkčního algoritmu. I když se data moc neliší, tak vliv nelinearity na naměřené spektrum je fatální. Simulovaná spektra jsou na obr. 4.



Obr. 4 Simulovaná spektra čtyř monochromatických signálů – vliv hysterezního chování piezoposuvu

4 Korekční algoritmus

Jak už jsem zmínil, získaná data při nelineárním posuvu zrcadla nejsou nepoužitelná. Pouze je získávám během vzorkování ve špatný okamžik vlivem měnící se rychlosti zrcátka. Pokud bych tedy vynesl závislost intenzity záření na nelineárním posuvu ve formě diskretních dat a provedl lineární interpolaci, tak bych mohl zjistit hodnotu autokorelační funkce pro ideální, tedy lineární posuv. Tuto myšlenku ilustruje obr. 5. V grafu je modrými křížky vyjádřena naměřená hodnota intenzity záření v jednotlivých krocích zrcadla během nelineárního posuvu (velikost kroků je proměnná při konstantní frekvenci vzorkování). Po provedení lineární interpolace dat se zjistí hodnota intenzity odpovídající lineárnímu krokování zrcadla. Tato data jsou znázorněna červenými křížky a z nich se poté provede Fourierova transformace. Data se dále interpretují jako by se zrcadlo pohybovalo konstantní rychlostí, která lze dopočítat pokud znám délku uražené dráhy zrcadla a dobu trvání pohybu zrcadla. Počet kroků nelineárního i lineárního posuvu se však musí rovnat a to znamená, že se musí rovnat i počty dat autokorelační funkce.



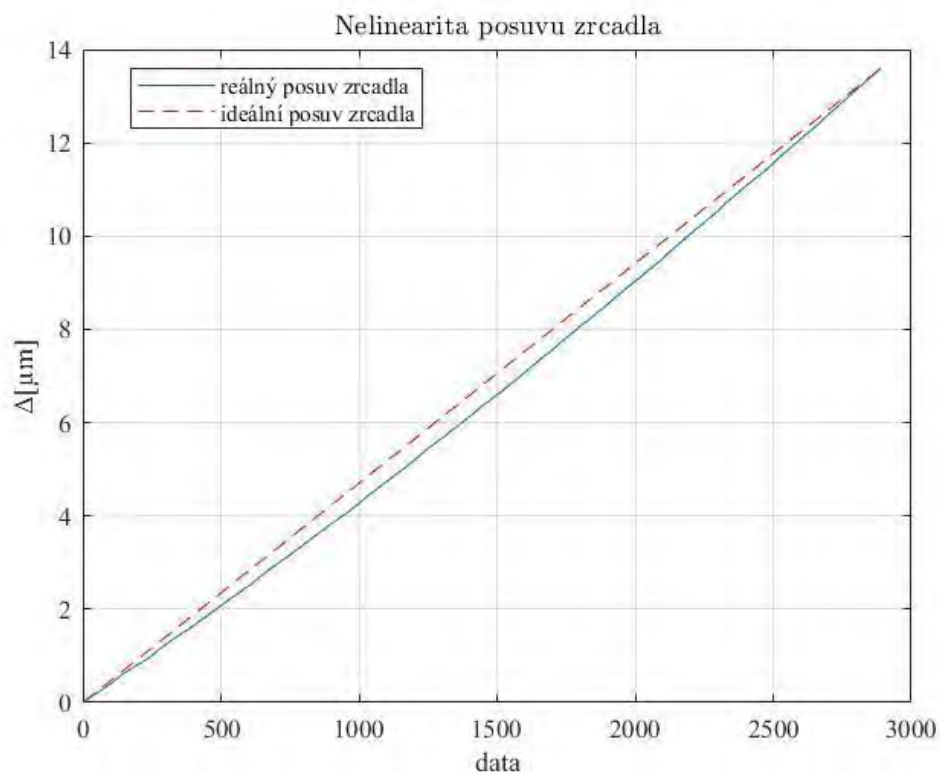
Obr. 5 Princip korekce autokorelační funkce

Jinými slovy, nezabývám se samotnou hysterezi piezoposuvu, ale opravuji až naměřená data. K realizaci jsem však potřeboval přesně měřit polohu pohyblivého zrcadla. Nabídlo se mi měření pomocí monochromatického laseru o vlnové délce 632,8 nm (He-Ne laser). Ten se připojí na vstup interferometru a nechá se interferovat současně s analyzovaným zařízením. Jeho výhodou je, že na výstupu získám autokorelační funkci, která má čistě kosinový průběh. Perioda vlny má délku 632,8 nm, což odpovídá uražené vzdálenosti zrcadla o délce 316,4 nm (polovina z vlnové délky laseru). Podle počtu vln této autokorelační funkce tak zjistím vzdálenost, kterou zrcadlo urazilo. Nelinearitu pohybu pak měřím z měnící se délky periody této kosinové funkce. Mám totiž k dispozici periodu vzorkovací frekvence, a tak mohu z rozložení korelogramu i zjistit, jak rychle se zrcátko pohybovalo.

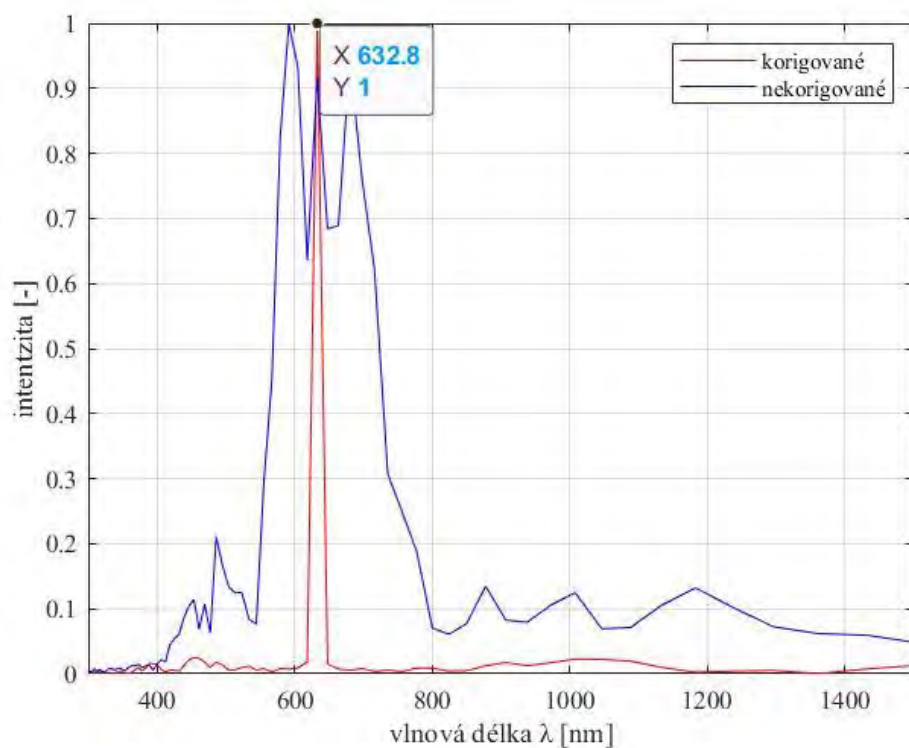
Algoritmus jsem implementoval v prostředí MATLAB a vyzkoušel ho na reálném zařízení ve škole, které sestavil můj vedoucí práce. Pro začátek jsem využil He-Ne laser jak pro měření zrcátka, tak i jako analyzovaný zdroj záření. Na obr. 7 je výsledek měření. Algoritmus byl tedy funkční a já tak mohl začít s návrhem vlastního spektrometru, kde bych tento algoritmus mohl aplikovat.

5 Návrh vlastního spektrometru

Na obr. 8 je schéma vlastního návrhu spektrometru, který byl úspěšně realizován a vyzkoušen. Červený svazek představuje záření z He-Ne laseru a). Do slučovače d) vstupuje záření ze zdroje j) jehož spektrum analyzuji. Záření obou zdrojů prochází prostorovým filtrem e) a kolimátor f) z nich vytváří dvě rovinné vlny. Ty vstupují do Michelsonova interferometru. Dělič i) rozdělí obě interferovaná záření amplitudově v poměru 50:50. Filtr k) propouští pouze záření He-Ne laseru a senzor l) zaznamenává jeho korelogram. Polarizátorem m) filtruji polarizaci He-Ne laseru a na senzoru n) zaznamenávám autokorelační funkci analyzovaného záření.

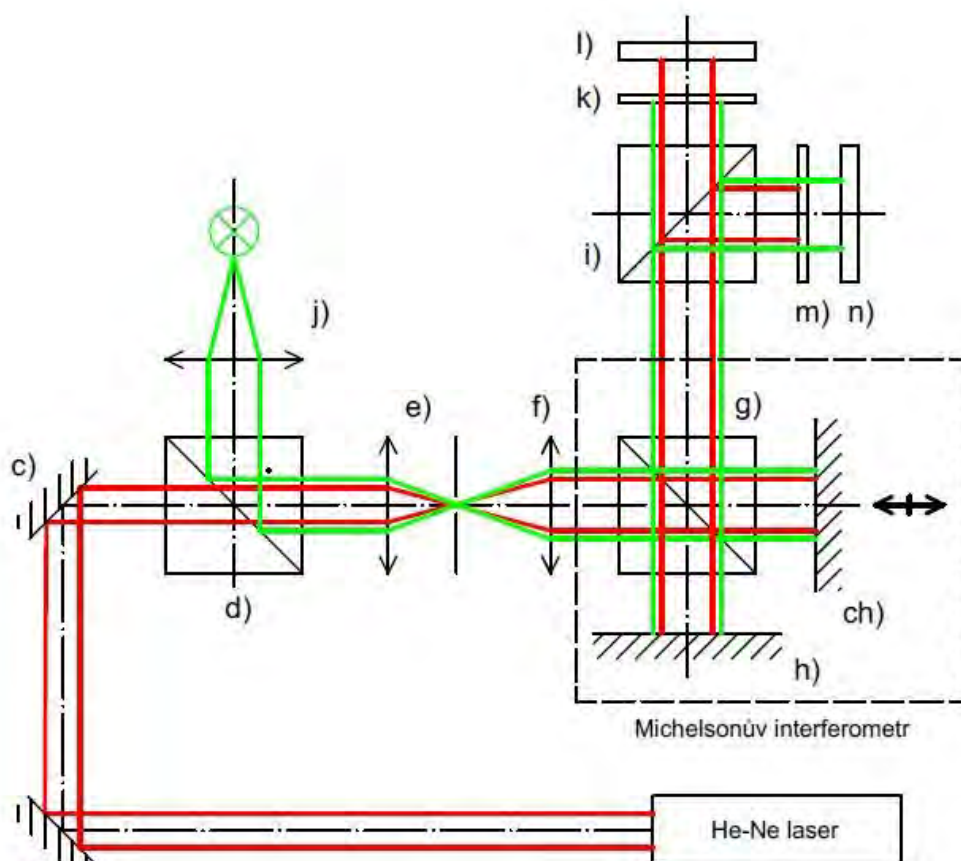


Obr. 6 Namodelovaná nelinearita posuvu piezostolku z rozložení korelogramu He-Ne laseru



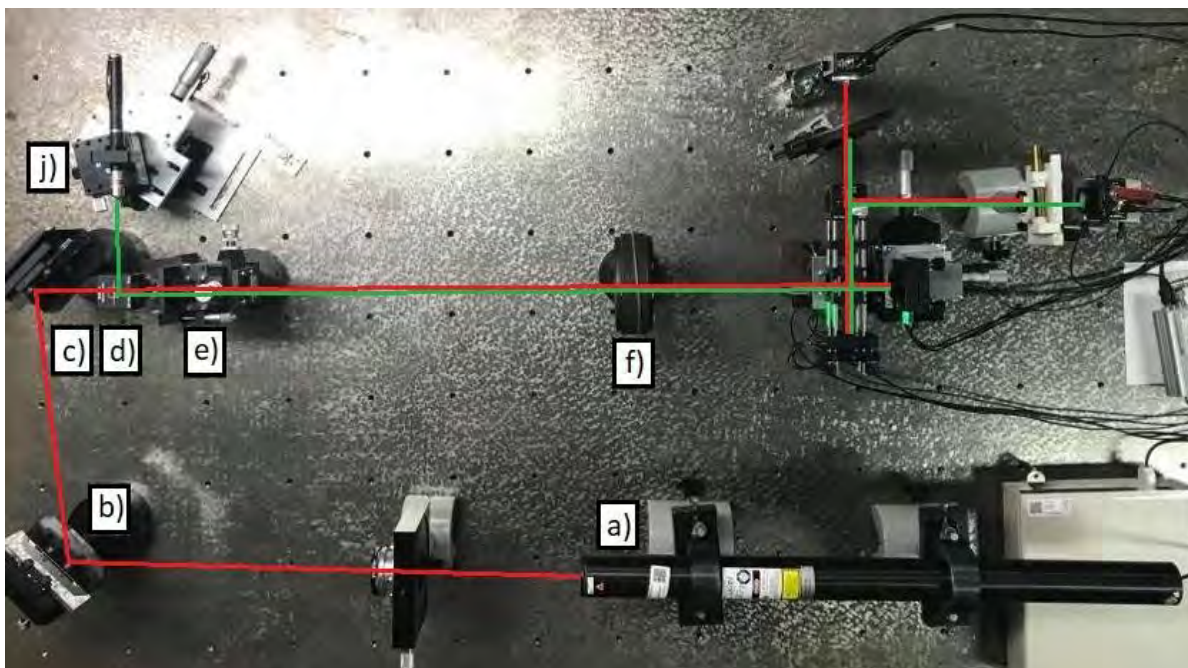
Obr. 7: Naměřená spektrální čára He-Ne laseru – modré spektrum je získané z neopravených dat

Realizace navrženého spektrometru se však neobešla bez potíží. Významným problémem byly například mechanické vibrace. Celá konstrukce stála původně na systému z přesných posuvných a otočných stolků, které umožňovaly naklápět a posouvat interferometr vůči vstupující rovinné vlně tak, aby vlna vstupovala do systému kolmo. Konstrukce však nebyla pravděpodobně dost tuhá, což se projevilo na nestabilním interferenčním obrazci na výstupu spektrometru. Takové chování však negativně ovlivnilo funkci korekčního algoritmu, který vyžadoval hladký průběh zejména He-Ne korelogramu, použitého k měření polohy zrcadla. Vibrace jsem řešil maximálním zjednodušením konstrukce a postupným přidáváním optických prvků.

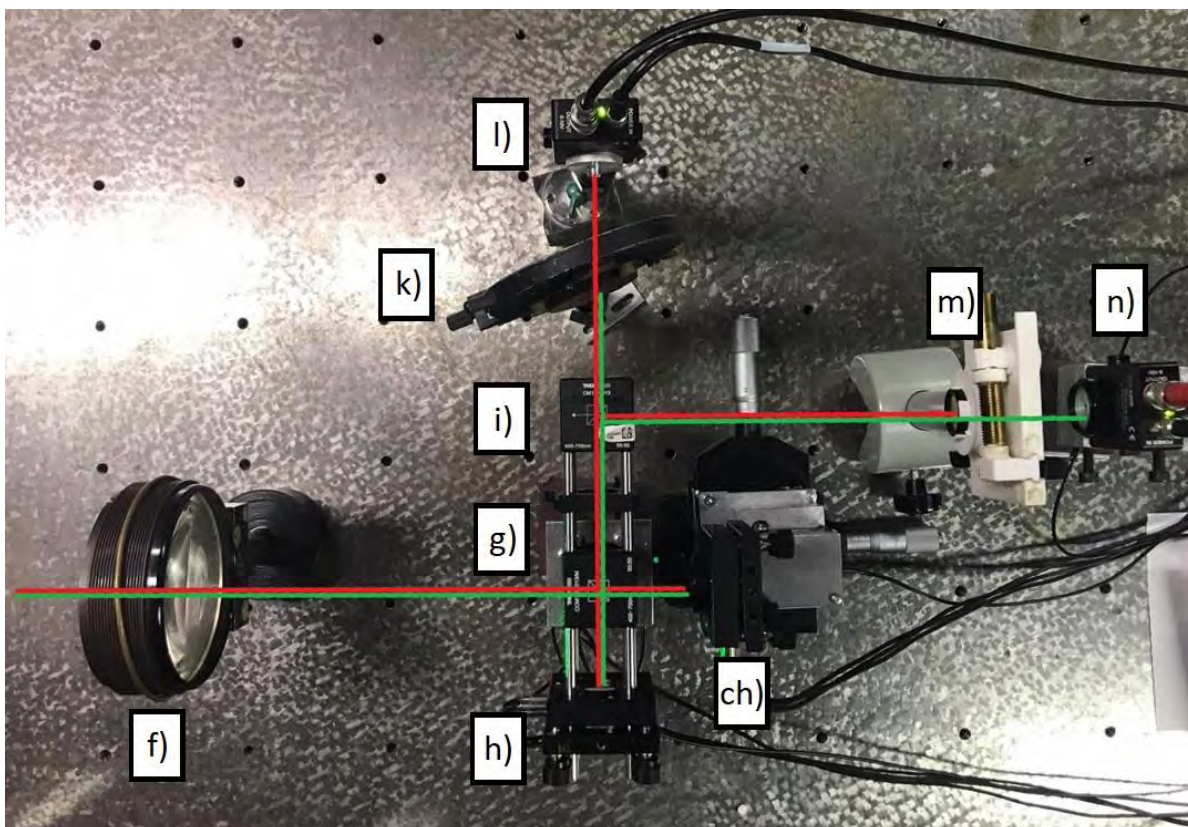


Obr. 8 Schéma navrženého spektrometru

Dalším problémem byl záznam dat na senzoru n). Nepodařilo se mi opticky odfiltrovat interferované záření He-Ne laseru z analyzovaného záření. Ve finálním spektru se proto objevovala spektrální čára měřicího laseru. Problém jsem řešil softwarovou filtrací dat (v mém případě pásmovou zádrží). V souvislosti s tím jsem preventivně filtroval i šum z celého datového souboru. Použité filtry jsem navrhl tak, aby propouštěly s maximálním zesílením vlnové délky od 400 – 700 nm, pro které byl spektrometr navržen.



Obr. 9 Celková optická sestava. Popisky odpovídají schématu



Obr. 10 Michelsonův interferometr se senzoricou částí

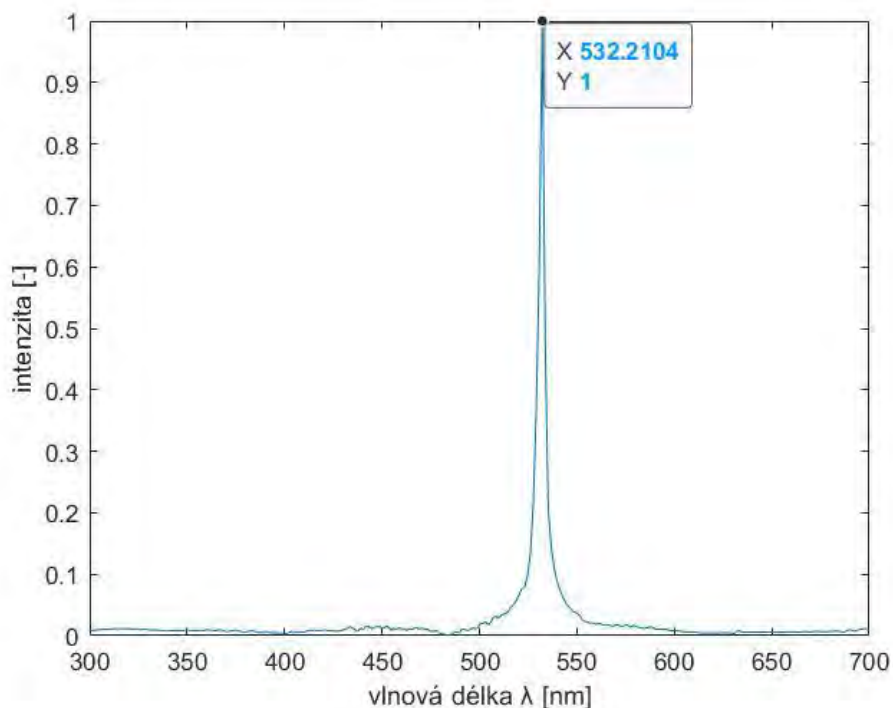
6 Ověření funkčnosti

Pro ověření funkčnosti spektrometru jsem použil zelené a fialové laserové ukazovátko, jejichž vlnovou délku jsem chtěl pomocí spektrometru změřit. Znal jsem katalogové hodnoty vlnových délek [5], nicméně měření jsem si ještě ověřil konvenčním disperzním spektrometrem. Na obr. 11 je tabulka s naměřenými vlnovými délkami.

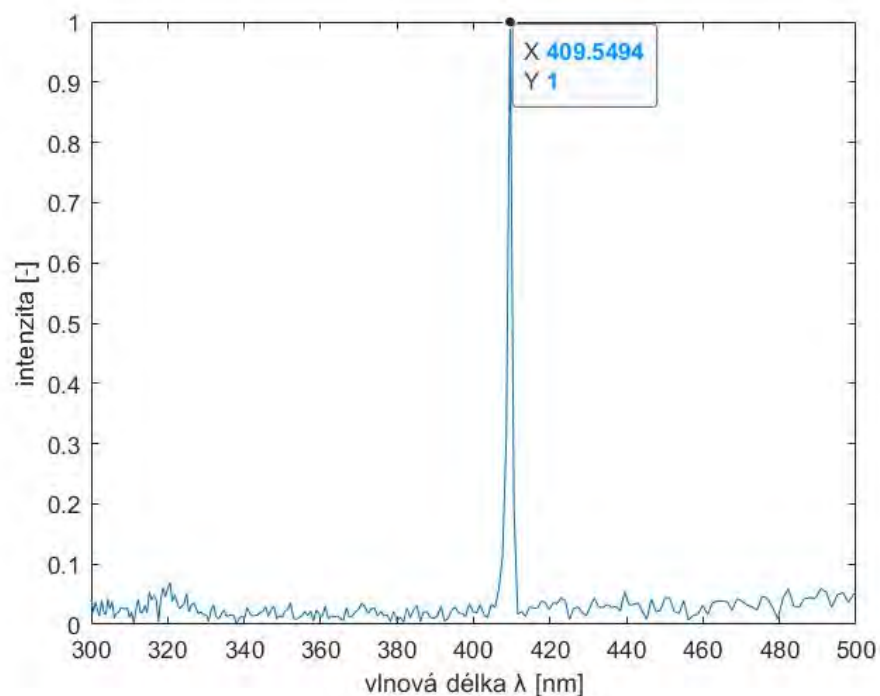
	<i>Katalogová hodnota</i>	<i>Disperzní spektrometr</i>	<i>Navržený spektrometr</i>
Zelený laser	532 nm	532 nm	532,2 nm
Fialový laser	405 nm	409 nm	409,5 nm

Obr. 11 Porovnání naměřených vlnových délek zeleného a fialového laseru

Na obr. 12 a 13 jsou naměřené spektrální čáry obou laserů. Vlnová délka zeleného laseru byla stanovena na 532,2 nm s rozlišením 1,7 nm. Vlnová délka fialového laseru je pak 409,5 nm a byla změřena s rozlišením 1 nm. Spektrum fialového laseru je poměrně silně zatíženo šumem. Signál byl velice slabý, a tak je poměr signál – šum malý. Nicméně vlnové délky naměřené navrženým spektrometrem se liší jen o desetiny nm oproti vlnovým délkám naměřeným disperzním spektrometrem. V rámci návrhu spektrometru jsem prováděl i analýzu možných chyb spektrometru, ale zjistil jsem, že veškeré chyby jsou zanedbatelné v porovnání s rozlišením spektrometru. Navržený spektrometr jsem tedy prohlásil za funkční.



Obr. 12 Spektrální čára zeleného laseru naměřená navrženým spektrometrem



Obr. 13 Spektrální čára fialového laseru naměřená navrženým spektrometrem

7 Závěr

V rámci projektu bylo cílem navrhnout a postavit interferenční spektrometr s Fourierovou transformací a vyřešit problém hysterezního chování piezoposuvu pohyblivého zrcátka. Jednalo se o komplexní úlohu, kde jsem si vyzkoušel práci s optickými prvky v kombinaci s programováním v prostředí MATLAB. Zároveň se toto téma stalo i tématem pro moji diplomovou práci.

Navazoval jsem na výsledky, ke kterým jsem dospěl v minulém projektu. Vytvářel jsem matematické simulace daného problému a navrhl funkční algoritmus korigující nelinearity již zmíněného posuvu. Praktické činnosti předcházelo studium odborné literatury a sepsání odborné rešerše.

V rámci tohoto projektu jsem navržený korekční program modifikoval tak, aby se dal ověřit na reálném zařízení ve školní laboratoři, kde byl sestaven spektrometr mým vedoucím diplomové práce. Algoritmus byl úspěšně vyzkoušen a já začal s návrhem vlastního spektrometru. Navržený spektrometr měří v rozsahu od 400 – 700 nm (viditelná oblast záření) s rozlišením 1,7 nm při vlnové délce 532 nm. Byl úspěšně realizován ověřovací experiment, kdy jsem zjišťoval pomocí spektrometru vlnovou délku zeleného a fialového laseru.

Spektrometr je funkční, ale stále se nabízí prostor pro různá vylepšení. Určitě by se dalo lépe využít světla ze světelného zdroje, jehož ztráty jsou poměrně veliké. Dále jsem omezen koherenční délkou zdroje záření, které do spektrometru vpouštím. Problém je v seřízení přístroje na nulový rozdíl optických drah v Michelsonově interferometru. Ručním mikrometrickým posuvem referenčního zrcadla s rozlišením 0,01 mm je přesné nalezení této pozice téměř nemožné, a proto bílé světlo o koherenční délce v jednotkách mikronů nebude nikdy interferovat. Řešením by tedy bylo, kdyby i referenční zrcadlo bylo posouváno piezoaktuátorem s výrazně vyšším rozlišením. Protože je spektrometr provozován v oblasti viditelného záření, bylo by vhodné navýšit rozlišení přístroje použitím posuvu s delším pracovním rozsahem než je pouhých 100 μm . Dosažené rozlišení 1,7 nm při vlnové délce 532 nm by bylo dostatečné pro provoz spektrometru v infračervené oblasti. Pro rozšíření spektrálního rozsahu spektrometru by bylo potřeba použít optické prvky navržené pro větší rozsah vlnových délek.

Práce mě obohatila o řadu nových zkušeností, zejména při práci s optickými prvky a jejich seřizování. Překonávání překážek mě naučilo, že při návrhu zařízení je nejlepší hledat co nejjednodušší cesty, využívat základních fyzikálních principů a především nabyt dostatečného množství informací, ze kterých poté mohu těžit.

Reference

- [1] DAVIS, Sumner P., Mark C. ABRAMS a James W. BRAULT. Fourier Transform spectrometry. San Diego: Academic Press, 2001. ISBN 0-12-042510-6.
- [2] NĚMCOVÁ, Šárka. Přednášky Vlnová optika [online]. Dostupné z: <https://moodle-vyuka.cvut.cz/course/view.php?id=1160>
- [3] PAVLÍČEK, Pavel a HÝBL Ondřej. Měření spektra svítící diody pomocí interferenční spektroskopie. Univerzita Palackého Olomouc, 2006.
- [4] FTIR engine/Technical note [online]. [cit. 2021-7-15]. Dostupné z: https://www.hamamatsu.com/resources/pdf/ssd/fir_engine_kacc9012e.pdf
- [5] Laser ukazovátka [online]. [cit. 2021-7-27]. Dostupné z: <https://laser-shop.cz/68-laserova-ukazovátka/>
- [6] BLAŽEK, Pavel. Možnosti realizace interferenčního spektrometru. 2021. Diplomová práce. ČVUT v Praze.

POUŽITÍ NAVÁDĚNÉHO EVOLUČNÍHO ALGORITMU K OPTIMALIZACI PARAMETRŮ PSD REGULÁTORŮ (USE OF GUIDING EVOLUTIONARY ALGORITHM FOR OPTIMIZATION OF PSD REGULATOR PARAMETERS)

Petr Siblík¹

¹ Ústav přístrojové a řídicí techniky, fakulta strojní, ČVUT v Praze, siblpetr@gmail.com

Abstrakt: Článek nejprve představuje „naváděný evoluční algoritmus“ („guiding evolutionary algorithm“, GEA) určený pro řešení problémů globální optimalizace. Dále je představeno využití tohoto algoritmu pro optimalizaci parametrů diskrétních regulátorů. Následně je prostřednictvím simulovaných soustav otestována funkčnost a spolehlivost vytvořené optimalizační metody. V poslední části je algoritmus otestován také na reálné experimentální úloze.

Klíčová slova: naváděný evoluční algoritmus, GEA, Optimalizace řízení, PSD regulace

1. Úvod

V dnešní době je nemožné si představit výrobu bez automatizovaných procesů. Mezi stěžejní úseky automatizace patří převážně vhodné nastavení regulátoru příslušného řízeného procesu. V praxi je využívána široká škála nastavovacích metod, které každá skrývá jisté úskalí. Například experimentální metody mohou vést k nebezpečnému poškození soustavy, zatímco analytické zase potřebují přesnou matematicko-fyzikální analýzu systému, což může být pro některé systémy velmi složitý úkol. Velmi často se také využívají metody kombinující oba zmíněné přístupy, čímž eliminují jejich negativa.

Bezpečné seřízení regulátoru u neznámých (matematicko-fyzikálně nepopsaných) soustav je velice lákavá myšlenka. Velkou pomocí v této oblasti představují nejrůznější optimalizační algoritmy, které prostřednictvím simulovaných běhů regulované soustavy naleznou optimální nastavení regulátoru pro danou soustavu.

Cílem této práce je využití optimalizační metody „naváděný evoluční algoritmus (Guiding Evolutionary Algorithm, GEA) na hledání optimálních parametrů vybraných regulátorů. Schopnost nalezení optimálního nastavení regulátorů je nejprve testována na simulovaných soustavách. Optimalizační simulace jsou prováděny na aproximacích zmíněných soustav. Nepřesnost mezi původní soustavou a její aproximací simuluje reálné využití v praxi, kdy model soustavy, na kterém je prováděna optimalizace, také dokonale neodpovídá reálné soustavě. Na závěr je optimalizace otestována na reálné experimentální úloze.

2. Naváděný evoluční algoritmus

Guiding Evolutionary Algorithm with Greedy Strategy [1] (GEA) byla představena týmem Cao, Xu & Goodman v roce 2016 pro řešení globálních optimalizačních problémů. Jedná se o metaheuristický algoritmus založený na dvou dříve vytvořených optimalizačních metodách: Netopýří algoritmus (Bat Algorithm, Yang, 2010) [2] a Optimalizace hejnem částic (Particle Swarm Optimization, Kennedy & Eberhart, 1995) [3]. Kombinací těchto algoritmů získává GEA výhody každého z nich a zároveň ještě vylepšuje schopnost konvergence ke globálnímu optimu.

Stejně jako jiné evoluční algoritmy pracuje GEA s populací n jedinců, jenž reprezentují d optimalizovaných parametrů ve formě vektoru.

$$x_i (i = 1, 2, \dots, n) = (x_{i1}, x_{i2}, \dots, x_{id}) \quad (1)$$

Kvalita či vhodnost jedince je posuzována prostřednictvím kritériální funkce (2). Hledání globálního optima – řešení optimalizační úlohy pak probíhá skrze minimalizaci nebo maximalizaci této funkce.

$$f(x), x = (x_1, x_2, \dots, x_d) \quad (2)$$

Jedinec s nejlepší kritériální funkcí v populaci je označován jako nejlepší jedinec x^* .

Optimalizace, hledání nejlepšího řešení, probíhá postupnou aplikací tří po sobě následujících operací (křížení, mutace a lokální hledání), které modifikují optimalizované parametry jedinců.

2.1 Křížení

Křížení je operace zahajující modifikaci jedince. V rámci naváděného evolučního algoritmu zprostředkovává hlavní mechanismus optimalizace a je realizováno podle následujícího výrazu:

$$x_i^t = x_i^{t-1} + (x_* - x_i^{t-1}) \cdot \beta, \quad (3)$$

kde x_i^t je nový jedinec, x_i^{t-1} je modifikovaný jedinec x_i v předchozí generaci, x_* je aktuální nejlepší jedinec a β je délka kroku přírůstku parametrů – náhodná hodnota ze zadaného rozsahu s uniformním rozdělením. Aktuální nejlepší jedinec x^* je zkombinován s aktuálně modifikovaným jedincem x_i^{t-1} za vzniku nového jedince x_i^t . Efektivnost této operace spočívá v tom, že se jedinci v každé iteraci „pohybují“ směrem k aktuálnímu nejlepšímu jedinci, přičemž dlouhodobě společně putují ke globálnímu/lokálnímu optimu.

2.2 Mutace

Přestože je křížení velmi účinný optimalizační nástroj, jeho nevýhodou je jeho velká pravděpodobnost uvíznutí v lokálním optimu. Z toho důvodu je do algoritmu zařazena mutace daná vzorcem (4). Náhodná změna parametrů jedince zvyšuje rozmanitost populace a tím zprostředkovává únik z lokálního optima.

$$x_i^t = x_i^t + \epsilon M, \quad (4)$$

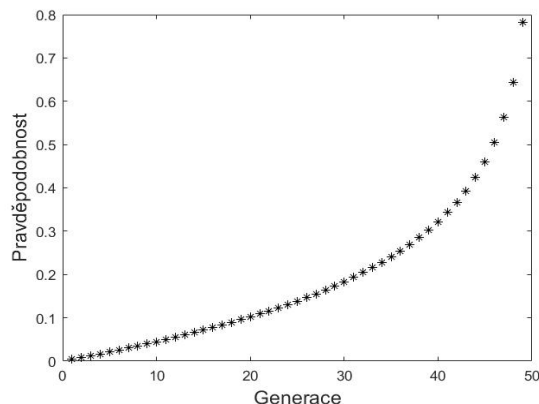
kde x_i^t je jedinec po křížení, ϵ je náhodné číslo s uniformním rozdělením $[-1,1]$ a M je vektor určující rozsah mutace. Za předpokladu, že rozsah j -té dimenze jedince x je $[a, b]$ pak:

$$M_j = \max(x_{ij}^t - a, b - x_{ij}^t), j = (1, 2, \dots, d) \quad (5)$$

Nebezpečí uvíznutí v lokálním optimu prostřednictvím křížení hrozí převážně v pozdějších generacích optimalizace. Proto je výhodné mutaci aplikovat až v těchto pozdějších iteracích algoritmu. Z toho důvodu je mutace jedince zatížena pravděpodobností p která se s přibývajícimi generacemi zvyšuje. Procentuální pravděpodobnost mutace p je dána vztahem:

$$p = c \cdot \ln\left(\frac{T_{max}}{T_{max}-t}\right) \quad (6)$$

kde c je limitující parametr, T_{max} je celkový počet generací, t je aktuální generace. Za předpokladu, že $T_{max} = 50$ a $c = 0.2$, pak se pravděpodobnost p zvyšuje v průběhu generací podle Obr. 1.



Obr. 1. Pravděpodobnost mutace

2.3 Lokální hledání

Funkce lokálního hledání spočívá v důkladném prozkoumání blízkého okolí nejlepšího jedince. Značně přínosná je v momentě, kdy je již nalezeno okolí globálního optima a pomáhá tak nalézt to skutečně nejlepší řešení. Z toho důvodu je opět vhodné aplikovat lokální hledání až v pozdějších generacích, a proto je také spjato s pravděpodobností p . Předpis lokálního hledání je definován rovnicí:

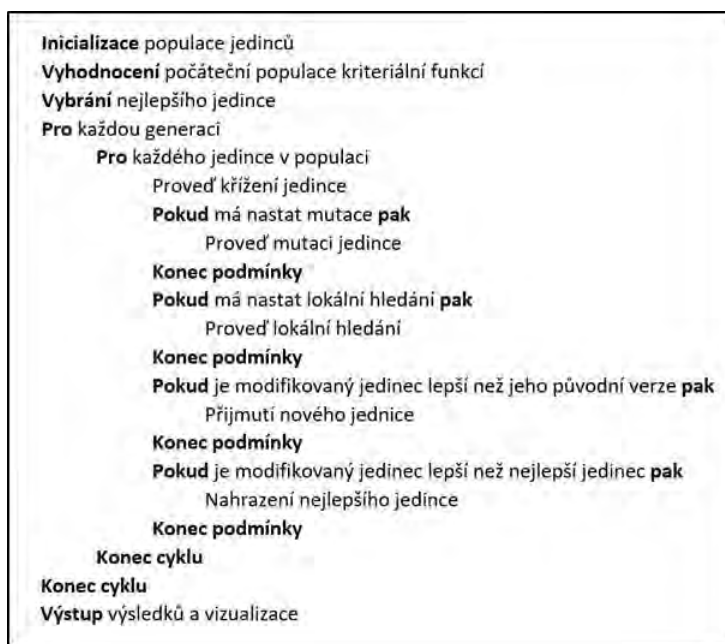
$$x_i^t = x_* + \epsilon L, \quad (7)$$

Kde x_* je aktuální nejlepší jedinec, ϵ je náhodné číslo s uniformním rozdělením $[-1,1]$ a L je vektor určující rozsah náhodného pohybu. Za stálého předpokladu, že rozsah j -té dimenze jedince x je $[a, b]$ pak:

$$L_j = l \cdot (b - a) \quad (8)$$

Kde l je koeficient rozsahu náhodného pohybu.

Následně dojde k výše zmíněnému ohodnocení modifikovaného jedince pomocí kritériální funkce $f(x)$. Nové parametry jsou přijmuty pouze v případě, že je modifikovaný jedinec, z hlediska kritériální funkce, v opačném případě jsou jeho parametry navráceny do původního stavu před modifikací. Je-li přijatý jedinec lepší než dosavadní nejlepší jedinec, převezme nový jedinec titul nejlepšího jedince. Následně začne naváděný evoluční algoritmus upravovat dalšího jedince v pořadí, dokud tímto způsobem neprojde celou populací čímž vytvoří novou generaci. Popsaný optimalizační cyklus se iterativně opakuje do té doby, dokud nedojde splnění některé z ukončovacích podmínek (např. dosažení požadované kvality řešení, počet proběhlých generací nebo nalezení nejlepšího možného jedince). Pseudokód naváděného evolučního algoritmu (Guiding Evolutionary Algorithm with Greedy Strategy) je vidět na obrázku 2.



Obr. 2. Pseudokód naváděného evolučního algoritmu

3. Aplikace naváděného evolučního algoritmu na optimalizaci diskretních regulátorů

Cílem této práce byla optimalizace parametrů diskretních regulátorů. Konkrétně normálního PSD regulátoru a dvou jeho modifikací PS-D a S-PD, které jsou blíže popsány v [4] (kapitola 4.3). Naváděný evoluční algoritmus byl vytvořen v prostředí programu Matlab. Jedinec x v této konkrétní aplikaci reprezentuje vektor jednotlivých nastavitelných parametrů regulátoru r_0 , r_i a r_d .

$$x = (r_p, r_i, r_d) \quad (9)$$

Hodnocení jedinců pomocí kritériální funkce je realizováno v prostředí programu Matlab – Simulink. Při hodnocení je provedena simulace reakce regulačního obvodu na jednotkový skok akční veličiny. Kritériální funkce pak hodnotí jak „vhodný“ byl získaný regulační pochod. V této práci bylo využito modifikované kritérium ITAE doplněné o členy minimalizující akční zásah a postihující velké změny akčních veličiny. Předpis této použité kritériální funkce je dán rovnicí (10).

$$f(x) = \int_0^t (C_e \cdot t \cdot |e(t)| + C_u \cdot u(t)^2 + C_{du} \cdot \Delta u) dt, \quad (10)$$

kde $e(t)$ je regulační odchylka, $u(t)$ je akční zásah, Δu je diference akčního zásahu, t je čas simulace, C_e je váhový koeficient regulační odchylky, C_u je váhový koeficient akčního zásahu a C_{du} je váhový koeficient diference akčního zásahu. Správné nastavení koeficientů C_e , C_u a C_{du} je pro optimalizaci klíčové, jelikož značně ovlivňují výslednou kritériální funkci. Pro obdržení optimálního nastavení regulátoru tak, aby výsledný regulační obvod vyhovoval technologickým požadavkům, je nutné nalézt mezi těmito váhovými koeficienty správnou rovnováhu.

4. Testy na simulovaných soustavách

Testování bylo provedeno na 5 soustavách různých vlastností běžně používaných v praxi pro testování regulátorů. Pro tyto soustavy byla získána aproximace modelem druhého řádu s dopravním zpožděním (SOPTD). Pro získané aproximace bylo prostřednictvím naváděného evolučního algoritmu získáno optimální nastavení parametrů regulátorů. Tyto regulátory byly následně aplikovány na původní systémy. Nepřesnosti mezi původní soustavou a její aproximací simulují reálné využití algoritmu, kdy model využitý pro optimalizaci také nebude dokonale reprezentovat reálnou soustavu.

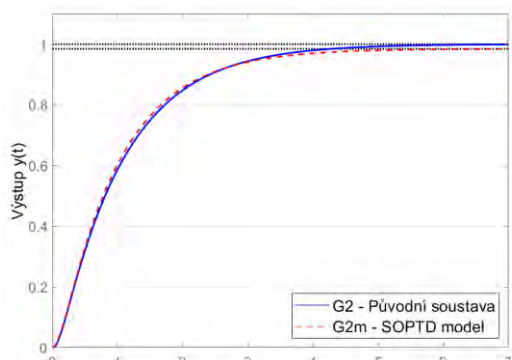
V tomto článku je uveden jeden příklad využití simulované soustavy.

4.1 Systém s dominantním náběhem

Pro soustavu s přenosem (11) byla získána aproximace SOPTD modelem ve tvaru (12). Přechodová charakteristika obou soustav je vidět na obrázku (3). Pro aproximační model G_{1m} bylo pro každý regulátor provedeno deset optimalizačních běhů. Průměrné hodnoty obdržených parametrů a jejich směrodatné odchylky jsou zaznamenány v tabulce (1).

$$G_1(s) = \frac{1}{(s+1) \cdot (0.1s+1) \cdot (0.01s+1) \cdot (0.001s+1)} \quad (11)$$

$$G_{1m}(s) = \frac{0.9843}{0.1095s^2 + 1.038s + 1} \quad (12)$$

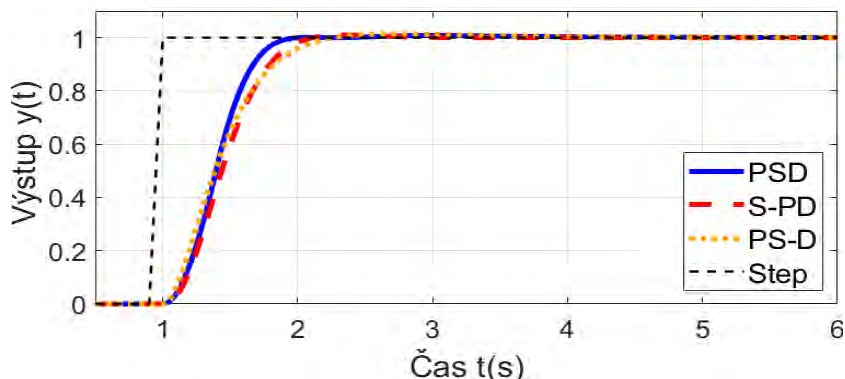
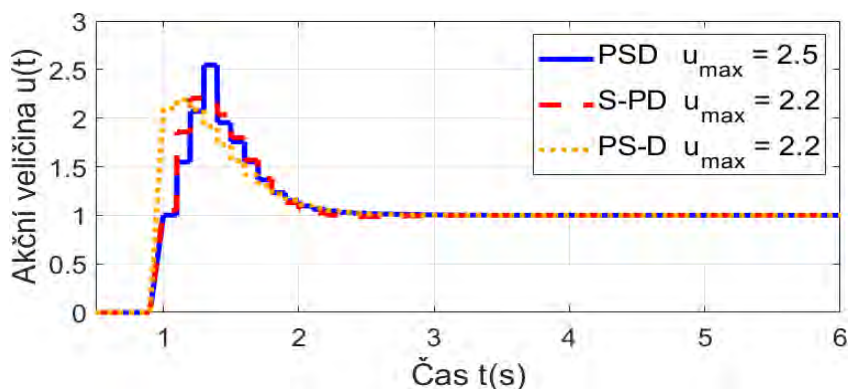


Obr. 3 Porovnání přechodových charakteristik G_1 a G_{1m}

Regulátor	r_0	r_i	r_d	f_{min}
PSD	2.08 ± 0.17	2.00 ± 0.14	0.20 ± 0.03	1.926 ± 0.017
PS-D	1.90 ± 0.03	2.17 ± 0.03	0.00 ± 0.00	1.785 ± 0.001
S-PD	5.54 ± 0.71	13.27 ± 1.64	0.47 ± 0.14	1.829 ± 0.003

Tab. 1: Optimalizované parametry regulátorů PSD, PS-D a S-PD pro soustavu G_1

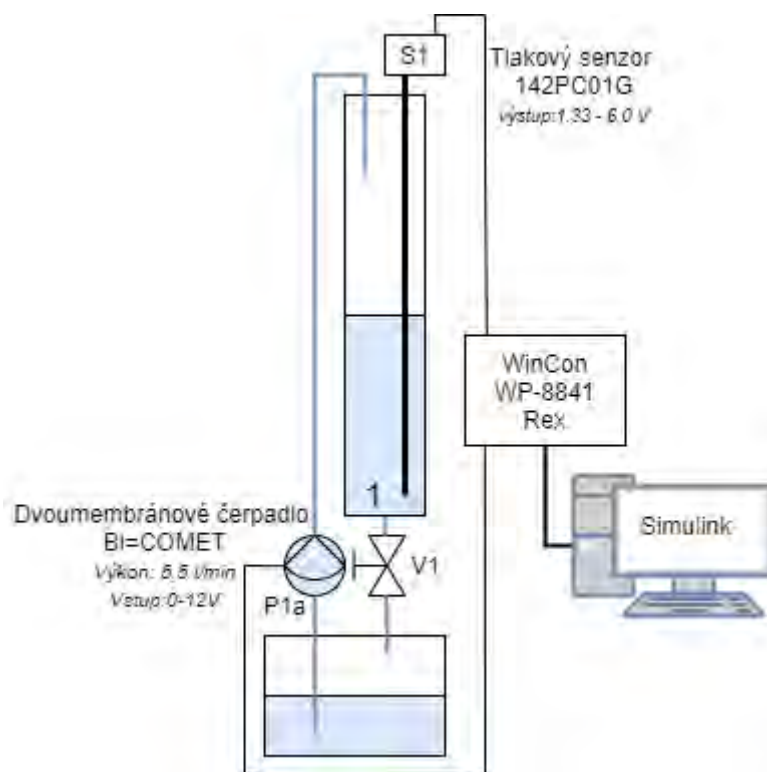
Z hodnot v tabulce (1) je patrné, že optimalizační běhy vedly u každého z regulátorů ke stabilním výsledkům, což je patrné z velmi malých směrodatných odchylek. Pro každý regulátor (PSD, PS-D, S-PD) bylo vybráno nejlepší nastavení parametrů regulátorů a tento regulátor byl použit pro regulaci původní soustavy G_1 . Na obrázku (4) je vidět reakce soustavy na jednotkový skok a na obrázku (5) jsou znázorněny akční zásahy jednotlivých regulátorů.

Obr. 4 Reakce regulované soustavy G_2 na jednotkový skok

Obr. 5 Porovnání akčních zásahů pro systém s dominantním náběhem

5. Testy na reálné experimentální úloze

Pro experimentální ověření funkčnosti optimalizačního algoritmu na reálné soustavě byla použita úloha jejíž schéma je na obrázku (6). Cílem byla regulace výšky hladiny pomocí čerpadla P1a. Výška hladiny byla snímána tlakovým senzorem a posílána prostřednictvím řídicí jednotky WinCon do počítače, kde byly v prostředí programu Matlab – Simulink realizované jednotlivé regulátory využité pro regulaci výšky hladiny. Pro reálnou soustavu byla opět získána aproximace modelem druhého řádu s dopravním zpožděním (13), který byl následně využit pro optimalizační simulace.



Obr. 6. Schéma experimentální úlohy

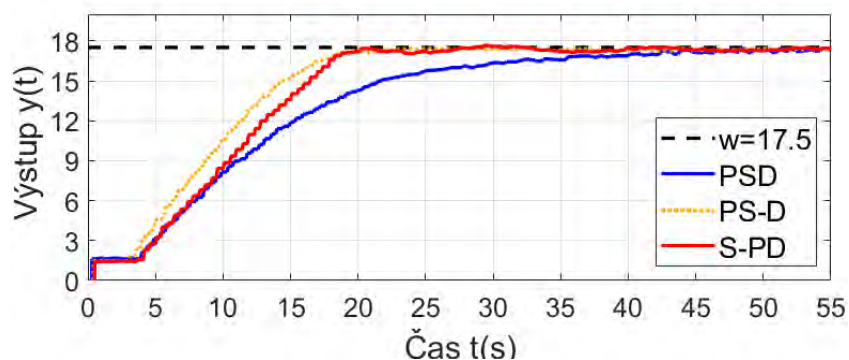
$$G_m(s) = \frac{15,1304}{28,7667s^2 + 119,6823s + 1} \cdot e^{-2,0847s} \quad (13)$$

V tabulce (2) jsou uvedeny průměrné hodnoty a jejich směrodatné odchylky optimalizovaných parametrů získaných v rámci deseti optimalizačních běhů pro každý regulátor. Podle vysoké směrodatné odchylky u parametrů regulátoru i u kritériální funkce jde usoudit, že optimalizační běhy pro PSD a PS-D regulátory podávali nekonzistentní, a tedy neuspokojivé výsledky. To může být způsobeno použitím nevhodných parametrů kritériální funkce pro danou kombinaci regulované soustavy a regulátoru. Oproti tomu výsledky obdržené pro S-PD regulátor se vykazují velikou konzistencí.

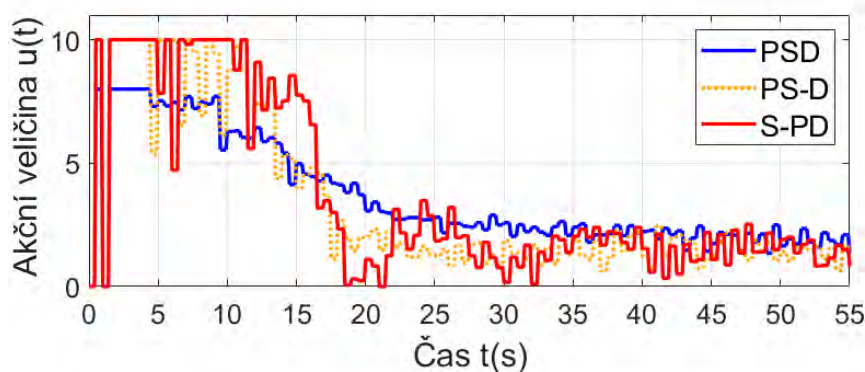
Regulátor	r_0	r_i	r_d	f_{min}
PSD	$1,57 \pm 0,16$	$0,11 \pm 0,039$	$1,52 \pm 0,66$	$48,853 \pm 21,915$
PS-D	$1,94 \pm 0,32$	$0,10 \pm 0,07$	$2,02 \pm 0,1,59$	$47,208 \pm 22,309$
S-PD	$3,34 \pm 0,05$	$0,53 \pm 0,01$	$2,91 \pm 0,02$	$31,183 \pm 0,215$

Tab. 2: Optimalizované parametry regulátorů PSD, PS-D a S-PD pro aproximační soustavu G_m

Pro každý regulátor bylo vybráno nejlepší optimalizované nastavení parametrů a vzniklé regulátory byly použity k regulaci reální experimentální úlohy. Na obrázku (7) jsou vidět obdržené regulační pochody. Na obrázku (8) jsou pak vyobrazeny příslušné průběhy akční veličiny. Přechodové pochody regulátorů PS-D a S-PD jsou si velmi podobné, zatímco regulátor PSD reguluje soustavu nejpomaleji.



Obr. 7. Regule experimentální soustavy pomocí optimalizovaných regulátorů PSD, PS-D a S-PD



Obr. 8. Porovnání akčních zásahů pro regulaci experimentální úlohy

6. Závěr

V rámci této práce byl v prostředí MATLAB vytvořen naváděný evoluční algoritmus pro optimalizaci parametrů diskrétních regulátorů (PSD, PS-D a S-PD). Taktéž byla vytvořena modifikovatelná kritériální funkce, umožňující respektující technologické požadavky na regulovanou soustavu. Ze simulačních testů této optimalizační metody lze usoudit, že podává velmi konzistentní a stabilní výsledky. Z experimentálního testování je patrné, že optimalizace parametrů PSD a PS-D regulátorů podávala neuspokojivě a nekonzistentní výsledky. Tato skutečnost byla nejspíše způsobena nevodným nastavením váhových koeficientů kritériální funkce. Oproti tomu pro S-PD regulátor byly výsledky velmi konzistentní.

Z výsledků provedených simulací a experimentu na reálné soustavě lze usoudit, že použití naváděného evolučního algoritmu k optimalizaci parametrů regulátorů je díky své snadné použitelnosti a adaptabilitě pro případné využití v praxi lukrativní.

Poděkování

Tato práce byla podpořena grantem Studentské grantové soutěže ČVUT č. SGS19/158/OHK2/3T/12.

Literatura

- [1] CAO, Leilei, Lihong XU a Erik D. GOODMAN. *A Guiding Evolutionary Algorithm with Greedy Strategy for Global Optimization Problems*. Computational Intelligence and Neuroscience [online]. 2016, 2016, 1-10 [cit. 2021-9-14]. ISSN 1687-5265. Dostupné z: <https://www.hindawi.com/journals/cin/2016/2565809/>; doi:10.1155/2016/2565809
- [2] EBERHART, R. a J. KENNEDY. *A new optimizer using particle swarm theory*. In: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science [online]. Nagoya, Japan: IEEE, 1995, s. 39-43 [cit. 2021-4-25]. ISBN 0-7803-2676-8. Dostupné z: <https://ieeexplore.ieee.org/document/494215> . doi:10.1109/MHS.1995.494215

- [3] YANG, Xin-She. *A New Metaheuristic Bat-Inspired Algorithm*. GONZÁLEZ, Juan R., David Alejandro PELTA, Carlos CRUZ, Germán TERRAZAS a Natalio KRASNOGOR, ed. *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, 2010, s. 65-74 [cit. 2021-4-25]. *Studies in Computational Intelligence*. ISBN 978-3-642-12537-9. Dostupné z: https://www.researchgate.net/publication/45913690_A_New_Metaheuristic_Bat-Inspired_Algorithm. doi:10.1007/978-3-642-12538-6_6
- [4] SIBLÍK, Petr. *OPTIMALIZACE PARAMETRŮ VYBRANÝCH REGULÁTORŮ POUŽITÍM GEA*. Praha, 2021. Diplomová práce. ČVUT v Praze, Strojní fakulta. Vedoucí práce Hofreiter Milan.
- [5] SALDANHA, Adrian. *Relay feedback identification using GEA for PID control*. Praha, 2020. Diplomová práce. České vysoké učení technické Praha, Fakulta strojní. Vedoucí práce Hofreiter M.
- [6] HOFREITER, Milan. *Základy automatického řízení*. V Praze: České vysoké učení technické, 2012. ISBN 978-80-01-05007-1.
- [7] CHUROVÁ, Vendula. *Optimalizační algoritmy inspirované živou přírodou* [online]. Brno, 2015 [cit. 2021-08-12]. Dostupné z: <https://theses.cz/id/7zc4m1/>. Bakalářská práce. Masarykova univerzita, Přírodovědecká fakulta. Vedoucí práce Mgr. et Mgr. Petr Dluhoš.

DIGITÁLNÍ DVOJČE PRO DISTANČNÍ VÝUKU PROGRAMOVATELNÝCH AUTOMATŮ

*Jakub Jura, Pavel Trnka, Martin Cahyna, Martin Novák*¹

¹ CVUT in Prague, Faculty of Mechanical Engineering

Jakub.Jura@fs.cvut.cz, Pavel.Trnka@fs.cvut.cz, Martin.Cahyna@fs.cvut.cz, Martin.Novak3

Abstrakt: Příspěvek prezentuje využití konceptu digitálního dvojčete a Software (SiL) a Hardware (HiL) In the Loop v distanční výuce po dobu koronavirových LockDownů v české Republice. Popsány jsou dva systémy. Jeden založený na simulaci elektropneumatických prvků polní instrumentace v prostředí FluidSim a jejich připojení k reálnému řídicímu systému. Druhý je založený na simulátoru výrobního systému FactoryIO, který je připojen k SoftPLC pomocí komunikačního protokolu Modbus po fyzické vrstvě Ethernetu.

Klíčová slova: Digitální dvojče, virtuální zprovoznění, distanční výuka, SiL, HiL,

Abstract: The submission presents the use of the digital twin and software (SiL) and hardware (HiL) In the Loop concept in distance learning during coronavirus Lock-Downs in the Czech Republic. Two systems are described in this article. One is based on the simulation of electro pneumatic field devices in the FluidSim environment and their connection to a real control system (PLC). The second is based on the factory production simulator, which is connected to SoftPLC using the Modbus communication protocol over the Ethernet physical layer.

Keywords: Digital Twin, Virtually commission, distanční výuka, SiL, HiL,

1 Úvod

V posledním roce a půl, tedy po dobu distanční výuky, v semestrech jaro 2020, podzim 2020 a jaro 2021, jsme se při snaze distančně vyučovat kybernetické předměty potýkali s tím, jak studentům zpřístupnit zkušenost s programováním řídicího systému bez laboratorní výuky. Výuka je normálně koncipována tak, že studenti programují řídicí systém. S nařízením lockdownu najednou chybělo vývojové prostředí pro programování, na kterém by mohli studenti pracovat doma - problém byl především s licencemi komerčních softwarů. Chyběl samozřejmě řídicí systém - zde se nabízela možnost jeho simulace a posléze i nouzové řešení založené na využití osobních počítačů a vývojářských kitů RPi, Arduino, ESP. A (ne)posledním problémem byla absence řízeného systému. Krizová situace paradoxně akcelerovala rozvoj perspektivních konceptů jako je digitální dvojče, virtuální zprovoznění a hardware a software in the loop a jejich zavedení do výuky.

1.0.1 Digitální dvojče

Pojem digitální dvojče pochází od Michaela Grievese (přednáška na téma PLM, 2003 [1]) a je definováno jako Virtuální ekvivalent fyzického produktu. Nebo specifičtěji jako virtuální reprezentace produktu založená na informacích (virtual representation of a physical product containing information about said product, with its origins in the field of product life-cycle management [1] parafrázováno dle [2]). Grieves rozšiřuje popis o schéma obsahující:

- reálný produkt ve fyzickém prostoru,
- virtuální produkt (digitální dvojče) ve virtuálním prostoru a
- obousměrnou komunikaci mezi nimi, tedy, data z reálného produktu do digitálního dvojčete a informace z virtuální reprezentace pro fyzický systém.

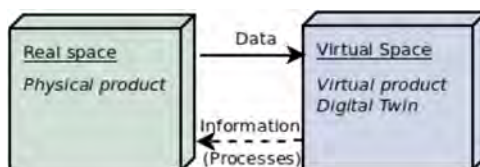


Fig. 1: Vztah zrcadlení virtuálního a fyzického prostoru, respektive virtuálního a fyzického produktu v něm

note: Grieves to definuje opačně, stejně jako na obrázku, tedy Real Space - Physical Product, ale v češtině dává lepší smysl opačný překlad reálný produkt - fyzický prostor.

Jak je zmíněno výše, digitální dvojče je zasazeno do kontextu životního cyklu produktu (potud ještě Grieves) a v tomto smyslu se digitální dvojče vyskytuje ve všech fázích životního cyklu a má různé podoby a využití - Návrh, prototyp, Instance. Za účelem návrhu, monitoringu, řízení, zlepšování, dokumentaci atp. V našem případě se jednalo o využití k návrhu + výuce.

David Jones a kolektiv ve svém rešeršním článku [2] nadefinovali zásadní pojmy týkající se digitálních dvojčat, jejichž vybrané podstatné položky jsou:

Tab. 1: Tabulka pojmů vztahujících se k digitálnímu dvojčeti. Výběr z [2]

Definiendum	Definiens
fyzická entita	Artefakt (raději prvek) reálného světa
virtuální entita	Počítačově generovaná reprezentace fyzického artefaktu
fyzické prostředí	reálný svět, v kterém je fyzická entita umístěna. Aspekty tohoto prostředí jsou měřeny a předávány do virtuálního prostředí
virtuální prostředí	Existuje v digitální doméně a je obrazem fyzického prostředí.
Fidelity	Věrnost, přesnost virtuální entity, vyjádřená například množstvím parametrů přenesených z reálné na virtuální entitu, jejich přesnost nebo úroveň abstrakce
stav	aktuální hodnoty všech parametrů (raději stavových proměnných)
Twinning	Akt synchronizace reálné a virtuální entity

Domény využití jsou téměř neomezené. Od digitálních dvojčat průmyslových výrobních systémů se přes další technické systémy jako jsou chytré domy nebo Smart Cities dostáváme k digitálním dvojčatům v rostlinné i živočišné výrobě [3].

Jako způsob využití digitálního dvojčete Jones [2] uvádí především (dle literární rešerše) optimalizaci, "data management" a monitoring, návrh, rekonfiguraci a "data driven design", "Model-based predictive control", sledování stavu strojů (možno přidat i prediktivní údržbu) a v nespolední řadě výuku. V souvislosti s tzv. distanční výukou v době koronavirové epidemie jsme byli postaveni před problém "jak učit programování průmyslových systémů bez průmyslových systémů". Tedy řízení bez řízeného objektu. Zatímco v jiných předmětech byla simulace běžná praxe, tak v Řízení programovatelnými automaty bylo v popředí právě připojení reálného systému a simulace nebyla využívána. Pro bezkontaktní výuku tedy bylo potřeba pokud možno zachovat práci s reálným kontrolérem a simulovat jen řízený systém. Tedy v podstatě použít digitální dvojče jako hardware in the loop. Vznikly z toho dva systémy. Jeden využívá simulaci elektropneumatického systému v prostředí FluidSim [4] a druhý simulaci výrobního systému v softwaru FactoryIO [5]. Předchůdcem těchto dvojčat byla bakalářská práce [6], kde simulace probíhala na RPi/Unipi, kteréžto bylo připojeno k PLC Siemens S7-1200. Pokud jsme chtěli využít softwarové simulace, museli jsme i zde mít nejen simulační program, ale k němu i interface pro připojení k reálnému PLC.

2 Digitální dvojče ve FluidSimu

FluidSim [4] je software pro simulaci pneumatických, elektropneumatických a hydraulických systémů. Studenti měli za úkol si namísto reálného zapojení úlohu "zapojit" pouze ve FluidSimové simulaci. K tomu byl připojen interface dodávaný výrobcem a nazvaný EasyPort. Ten z jedné strany komunikuje po USB s počítačem, na kterém FluidSim běží a z druhé strany poskytuje výstupy odpovídající 24 V průmyslovému systému - k čemuž je připojeno

PLC 3. Aby studenti mohli vyvíjet program doma, bylo zvoleno PLC Unipi Neuron [7], které je postavené na

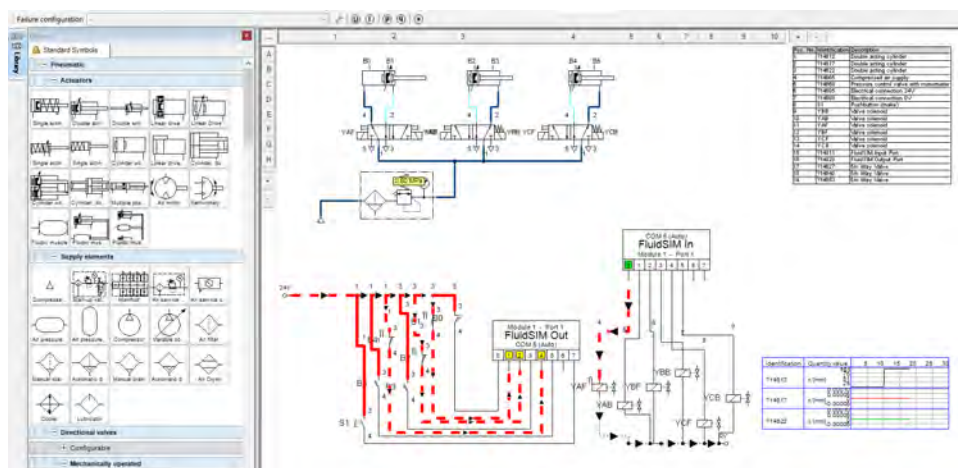


Fig. 2: Simulace elektropneumatického systému ve FluidSimu s připojeným konektorem fyzických výstupů

minipočítači RPi. Výhodou je SW otevřenost systému - v PLC je operační systém Linux a tudíž je možné do něj nahrát/nainstalovat runtime vlastního výběru. Zvolen byl OpenPLC [8], který téměř stoprocentně splňuje normu pro PLC IEC 61131 a disponuje všemi jazyky včetně SFC. Současně se jedná o svobodný a software. Vývojové prostředí je pro Linux i Windows. Studenti tedy mohli vyvíjet doma a testovat na digitálním dvojčeti, které bylo dostupné přes vzdálenou plochu Windows.

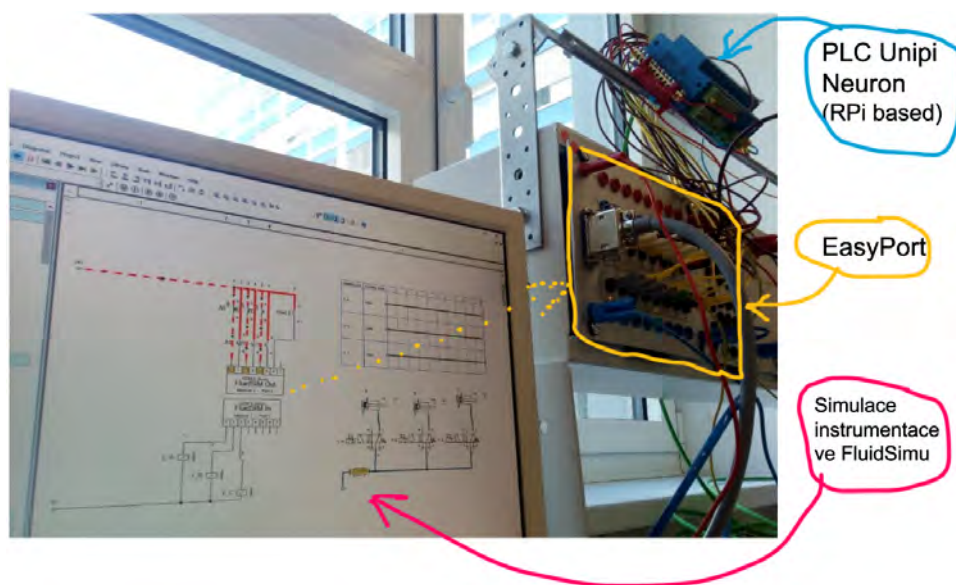


Fig. 3: Zapojení FluidSIMu jakožto digitálního dvojčete reálného elektropneumatického systému využitě pro virtuální zprovoznění úlohy v době distanční výuky

3 Digitální dvojče ve FactoryIO [5]

Digitální dvojče ve FactoryIO [5] mělo více variant využití. FactoryIO je simulátor výrobního procesu vytvořený herní společností Real Games. Má vlastní řídicí systém ControllIO a je možné k němu připojit externí řídicí systémy po fyzické vrstvě Ethernetu. Umí komunikovat po různých protokolech:

- Siemens: LOGO!, S7-200 / 200 SMART / 300 / 400, S7-1200 / 1500, S7-PLCSIM, WinSPS-S7
- Allen-Bradley: ControlLogix, CompactLogix, Micro800, MicroLogix 1100 1400, SLC 5/0X and SoftLogix

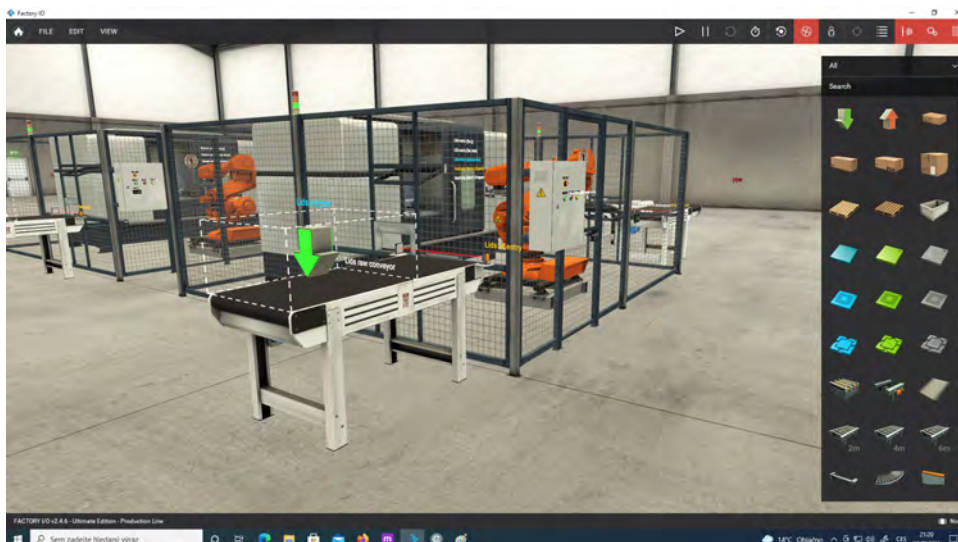


Fig. 4: Celá simulovaná scéna s robotizovanou buňkou s CNC strojem

- Modbus: Modbus TCP/IP Server and Client
- OPC: Client DA/UA
- Simulator Grafset-Studio (Automgen: Grafset and SysML)

Pro naše účely jsme používali komunikační řetězec: **OpenPLC - Modbus - FactoryIO**

4 Kyberpunkový závěr

„Konsenzuální halucinace prožívaná denně miliardami legitimních operátorů, v každém národě, dětmi, které se učí matematickým pojmům ... grafické zobrazení dat abstrahovaných z paměti každého počítače v lidské společnosti. Nepředstavitelná komplexita. Linie světla rozprostírající se v neprostoru mysli, klastry a konstelace dat. Jako světla velkoměsta, vzdalující se..." [9] Svobodný svět oprostěný od omezení fyzickou tělesností. Kyberkultura a kyberpunk jsou koncepty a hnutí, která pronikla do společenského prostoru nejdříve v době, kdy vlastnictví počítače bylo výsadou úzkého okruhu vyvolenců, které první počítače vtáhly do jejich světa a držely. Pohledem z vnějšku není jasné zda oni mistrně ovládali počítače ... nebo to bylo naopak. Typickým obrazem byl outsiderský vzhled, neatletická postava, mužské pohlaví a soustředění na to, jak počítač dokáže rozšířit naše mentální schopnosti (nebo naopak oni rozšiřovali schopnosti počítače?) a počítač, který jako paměťové médium používal kazetový pásek. Prostor vytvořený technologií nemusí být jen prostorem otevřeným simulaci a manipulaci - je i kreativním rozhraním, dovolujícím seberealizaci „reálně neuskutečnitelným" způsobem [10].

Stávají se z nás kyborgové, kombinace stroje a člověka. Naše technika se stále zmenšuje, je nám čím dál bližší a zanedlouho námi prostoupí.

- Bojuj s mocí.
- Pošli zpětnou vazbu zpátky do vládnoucího systému.
- Surfuj na Okrajích

BRANWYN, G. Kyberpunkové teze. cit dle [11]

Ale ona tato romantika exkluzivity končí v okamžiku, kdy je v tajném bratrstvu kyberpunkerů a počítačových mágů každý kdo chce. A po dobu online výuky tam do jisté míry museli být i ti, kdož ani nikdy nechtěli. A i ti si museli vyzkoušet zapojení se do online systému. Být neustále připojen k té "konsenzuální halucinaci". Téměř Matrix naživo. Literární kyberpunk se ve srovnání s online výukou minulých tří semestrů jeví jako pohádka pro malé děti.

Acknowledgement

The project was supported by grant SGS20/159/OHK2/3T/12.

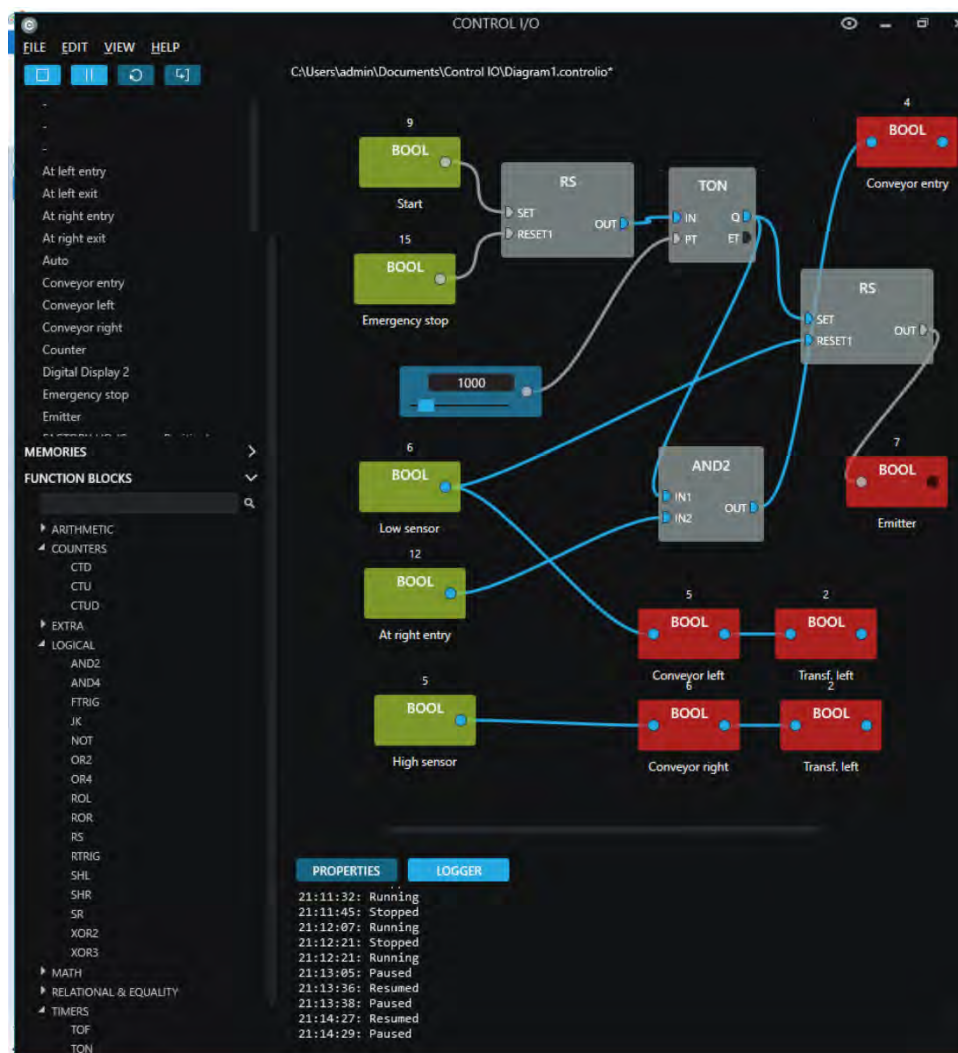


Fig. 5: Řídicí program v jazyce FBD napsaný v interním programovacím prostředí ControlIO

Literature

- [1] Michael Grieves. Digital twin: Manufacturing excellence through virtual factory replication. 2014.
- [2] David Jones, Chris Snider, Aydin Nassehi, Jason Yon, and Ben Hicks. Characterising the digital twin: A systematic literature review. 29:36–52.
- [3] Christos Pylaniadis, Sjoukje Osinga, and Ioannis N. Athanasiadis. Introducing digital twins to agriculture. 184:105942.
- [4] Fluidsim 5.
- [5] Real Games. Factoryio.
- [6] Kirill Rassudikhin. *Simulátor výrobního zařízení založený na RPi+UniPi*. 2020.
- [7] Faster CZ spol. s r.o. Unipi neuron, 2014-2021.
- [8] Thiago Alves. Openplc, April 17, 2019.
- [9] William Gibson. *Neuromancer*. Laser. OCLC: 42030019.
- [10] Jakub Macek. Defining cyberculture [koncept rané kyberkultury]. Publisher: Unpublished.
- [11] Kateřina STOJASPALOVÁ. *Průkopníci“ kyberkultury: Kyberpunk a Digerati*. Masarykova univerzita, Filozofická fakulta, Ústav české literatury a knihovnictví, Kabinet knihovnictví.

MODULAR STATION CP LAB

Jan Mohelník¹, Pavel Trnka²

¹ *České vysoké učení technické v Praze, Fakulta strojní, Jan.Mohelnik @fs.cvut.cz*

² *České vysoké učení technické v Praze, Fakulta strojní, Pavel.Trnka @fs.cvut.cz*

Abstrakt: Tato práce pojednává o třech laboratorních úlohách, které byly vytvořeny na modulární stanici CP Lab. Využíváme tak technologii, jenž se běžně vyskytuje v praxi, na aplikování naučené teorie z předmětu Automatické řízení v bakalářském studiu. Studenti zde mají možnost vidět jak hardwarové zpracování stanice, tak se podílet na jejím programování a propojit si teorii řízení s praxí.

Klíčová slova: CP Lab; Automatické řízení; Laboratorní úloha

Abstract: This work deals with three laboratory tasks that were created on a modular station CP Lab. We use the technology that commonly occurs in practice to apply the theory learned from the subject of Automatic Control in the bachelor's degree. Students have the opportunity to see both the hardware processing of the station and participate in its programming and connect control theory with practice.

Keywords: CP Lab; Automatic control; Laboratory task

1 Úvod

V laboratoři jsem měl na výběr ze dvou možností. Vrtací stanice, nebo horkovzdušný tunel. Každá z nich byla připevněna k paletovému dopravníku s ovládacím panelem. Já jsem si zvolil horkovzdušný tunel, jelikož mi přišel jako ideální kandidát pro všechny tři úlohy. Samotné stanice již byly fyzicky sestaveny a naprogramovány. Bohužel však dodaný program nesplňoval potřebné požadavky. Sloužil spíše, jako ukázka funkčnosti stanice, než pomůcka pro studenty v jejich rozvoji v oblasti automatizace. Bylo tedy zapotřebí stanici celou přeprogramovat, vymyslet způsob, jakým budou moci studenti na ní pracovat a zachovat původní program, jako ukázkou původního řešení.

1.1. CP Lab Heat Tunnel

Horkovzdušný tunel – Tato stanice má za úkol simulovat průmyslové řešení sušení nátěru (barvy) při průjezdu tunelem. Jako akční člen zde slouží ventilátor se zabudovanou výhřevnou spirálou, kterou lze regulovat na výkonu, pro měření teploty zde používáme odporový teploměr PT100 a jako možnost simulace vnějšího rušení procesu sušení, kovový rošt s dírami, jenž je přidělán na peci a ovládán skrze pohybový šroub přes plastové kolečko. Jako ukazatele zapnuté pece, tedy probíhajícího procesu schnutí, zde máme červenou LED lampu, která nám symbolizuje svou červenou barvou rozžhavenou pec. Senzor nám dovoluje měřit až do teplot okolo 200°C. Jelikož se ale jedná o školicí stanici, tak z důvodu bezpečnosti skutečná maximální dosažitelná teplota, po kterou můžeme pec rozežhát je 75~80 °C. [1]



Obr. 1. Heat Tunnel[1]

1.2. CP Lab Pallet Transfer Line

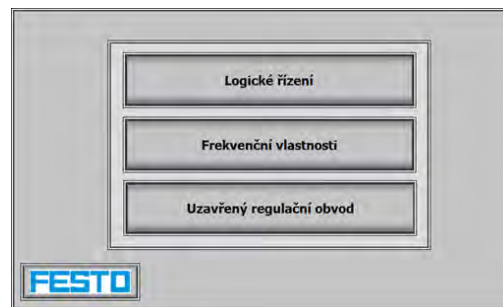
Dopravníkový pás - Slouží jako určitý základní modul pro všechny ostatní aplikační moduly (např. pec, vrtačka, kamerová kontrola a mnoho dalších). Každá dopravníková stanice je vybavena na svých stranách fotoelektrickými senzory, které zajišťují celkové propojení s více stanicemi sériově za sebou. Máme zde velké zastoupení indukčních a kapacitních senzorů, sloužících pro sledování pozice vozíku na páse a RFID senzor, který zpracovává data, jenž vozík se sebou přenáší v datové minci. Nejdůležitější část je zde rozvodná skříň, která v sobě ukrývá veškerou řídicí elektroniku. Velice důležitou součástí pro studenta, je ovládací dotykový panel, na němž probíhá veškeré měření a nastavování laboratorních úloh. [2]



Obr. 2. Pallet Transfer Line[2]

2. Samotné laboratorní úlohy

Při výuce automatizace v laboratořích přistupujeme ke třem základním typům úloh: Logické řízení, Frekvenční vlastnosti a Uzavřený regulační obvod.



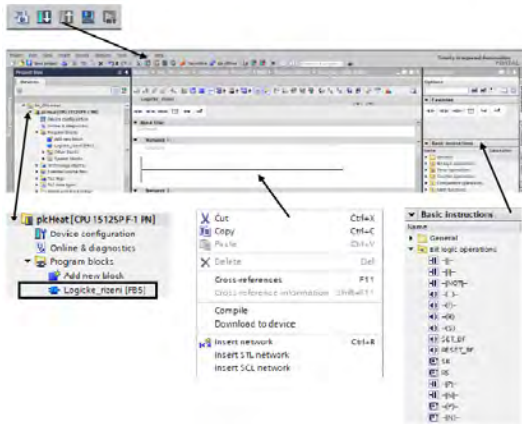
Obr. 3. Menu – Jednotlivé úlohy

2.1. Logické řízení

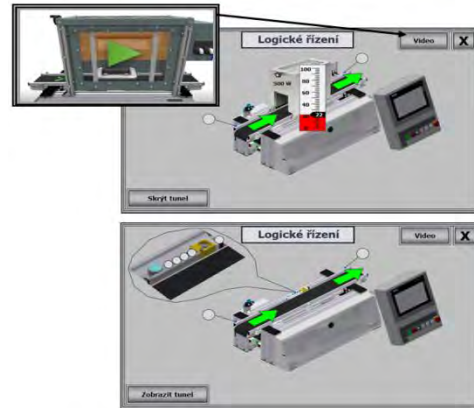
Pro úlohu mám připravenou obrazovku s modelem tunelu a vyznačenými čidly. Obrázek stanice na obrazovkách jsem použil přímo od výrobce stanice[1,2]. Jednotlivé puntíky nám zobrazují, či je čidlo „senzor“ v log.1 (zelené „●“), nebo v log.0 (šedé „●“). Jako ukázkou principu fungující naprogramované stanice jsem zde přidal tlačítko „Video“, které spustí na obrazovce animaci funkční linky, kterou jsem stáhl od výrobců této sestavy. Směr otáčení dopravního pásu je znázorněn blikajícími zelenými šipkami ve směru jízdy „←→“. V levém dolním rohu pak máme možnost zobrazení, nebo skrytí aplikační stanice horkovzdušného tunelu. Na něm je ukázána aktuální teplota a sepnutý výkon tunelu. Program, v němž budou studenti tuto úlohu programovat, se nazývá TIA Portal. Jedná se o platformu firmy SIEMENS. Prostředí tohoto softwaru je vcelku obsáhlé, a proto jsem ve své práci popsal jen části, které využijí studenti při svém programování. Ostatní věci si jde samozřejmě vyhledat na internetu, kde jsou volně dostupné návody od autorů programu. Pro programování jsem vytvořil funkční blok „**Logické řízení [FBS]**“, v kterém budou moct studenti tvořit svůj algoritmus horkovzdušného tunelu. Způsob programování je již na studentech. Dá se říct, že s jakýmkoliv (pro PLC dostupným) jazykem dokážeme naprogramovat funkční stanici, je ale otázka, jaký způsob je pro nás nejpříjemnější. Jsou k dispozici jazyky dle normy IEC 61131-3[8]. Co se týče přehlednosti, tak bych určitě doporučil LAD jazyk, jelikož se dá velice dobře pozorovat v jednotlivých reléových schématech funkčnosti programu. Co se týče pracování s daty tak SCL. Velice jednoduchá metoda pomocí základních znalostí programování, které vám ušetří spoustu starostí, například při práci s nepřímou adresací dat v LAD. Hlavní výhoda je, že LAD, SCL a STL, se dají používat a kombinovat v jednom bloku naráz. Co se týče FBD a GRAPH, tak pro jejich použití musíme vytvořit bloky s patřičným jazykem zvlášť. [7]

2.1.1. Postup studenta

Studenti tak na začátku dostanou seznam dostupných PLC adres, na které jsou připojena jednotlivé zařízení a dle zadání mají za úkol zprovoznit chod stanice v automatickém režimu.



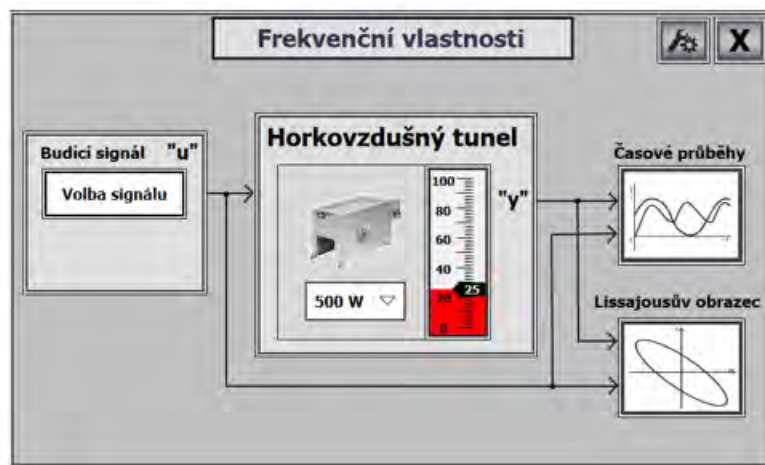
Obr. 4. TIA Portal prostředí



Obr. 5. Obrazovka – Logické řízení

2.2. Frekvenční vlastnosti

Tato úloha se zabývá metodami pro analýzu chování neznámých soustav, tedy v našem případě „**horkovzdušného modelu**“. To provádíme tak, že za pomoci budícího harmonického signálu $u = A \cdot \sin(\omega t)$ působíme na soustavu a sledujeme její chování skrze vynucený výstup $y = A \cdot \sin(\omega t + \varphi)$. Vynucené kmity bývají opožděny za budícím signálem a to právě skrze průběhy pozorujeme a měříme. [4]



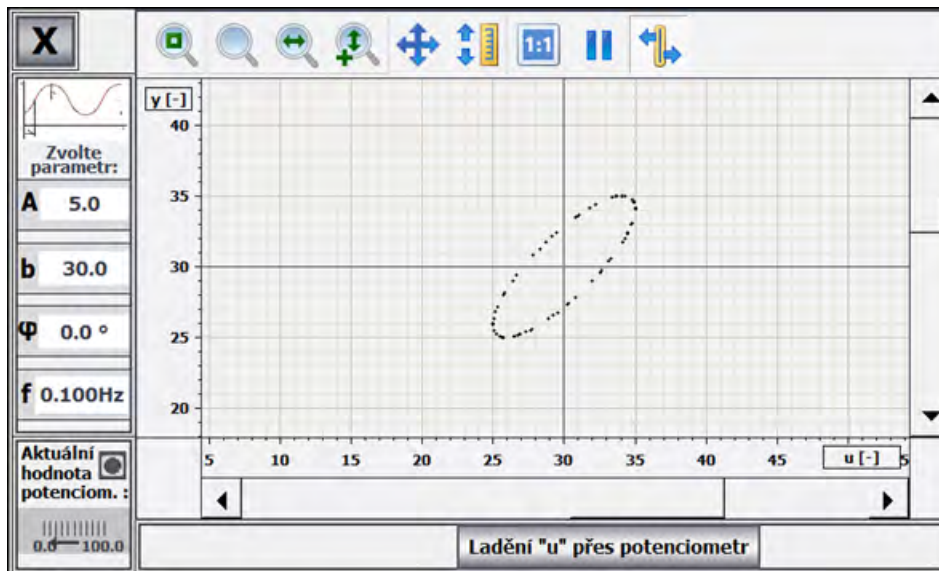
Obr. 6. Obrazovka – Frekvenční vlastnosti

2.2.1. Lissajousův obrazec

Lissajousova křivka „**obrazec**“ je vytvořená dvěma harmonickými signály, promítanými na osách x, y , kde: $x = A_1 \cdot \sin(\omega t + \varphi_1) + b_1$; $y = A_2 \cdot \sin(\omega t + \varphi_2) + b_2$ (1.1)

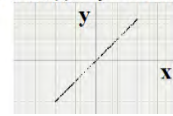
Co se týče laboratorní úlohy pro studenty, tak na ose „ x “ je zobrazován „ u “ budící harmonický signál z generátoru, který si sami nadefinujeme a na ose „ y “ je zobrazena hodnota teploty soustavy, která je právě vynucená měnicími se požadavky budícího signálu a jejíž chování je pro nás neznámé.

Vykreslování křivky funguje skrze funkci „**f(x)** (TrendView)“, kde nám **jeden bod jezdí po vypočtených souřadnicích křivky „ x, y “** a přibližně co 150 ms zaznamenává do grafu černým puntíkem svou dráhu. Po uplynutí určité doby tedy dostáváme z těchto bodů celkový graf „**obrazec**“. [4, 5]

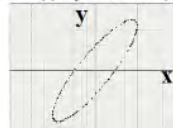


Obr.7. Obrazovka – Vykreslování křivek

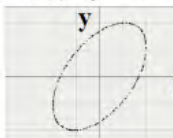
$$x = 10 \cdot \sin(t) \quad y = 10 \cdot \sin(t)$$



$$x = 10 \cdot \sin(t) \quad y = 10 \cdot \sin(t+0,5)$$



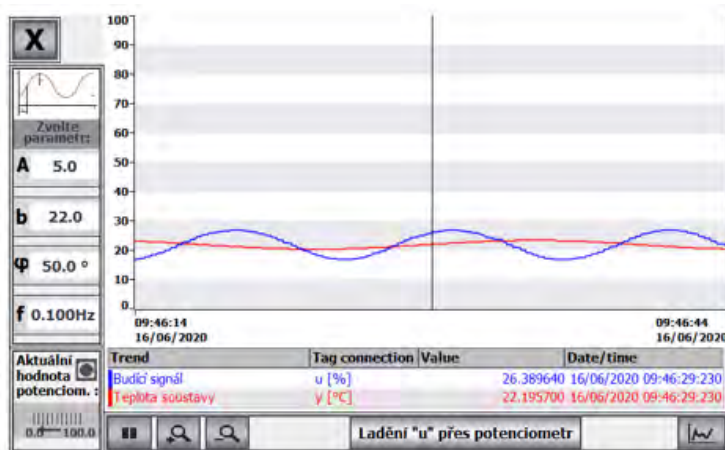
$$x = 10 \cdot \sin(t) \quad y = 10 \cdot \sin(t+1)$$



2.2.2. Časové průběhy

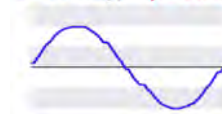
Stejně jako u předchozí metody, tak na ose „x“ je zobrazován „u“ budící harmonický signál z generátoru a na ose „y“ je zobrazena hodnota výstupu soustavy. Rozdíl oproti předchozí metodě je tedy pouze v tom, že tyto rovnice zobrazujeme společně na jedné časové ose.

Vykreslování křivky funguje skrze funkci „TrendView“, kde se nám v závislosti na přibývajícím čase (který řídí PLC) zobrazují co 100ms nové hodnoty v grafu ve formě bodů, které jsou interpolovány „spojeny“ s předchozími, aby tak křivka vypadala spojitě a hladce. Máme zde možnost pomocí posuvné osy měřit hodnoty jednotlivých křivek a změřené hodnoty se nám pak zobrazují společně s časem ve spodní tabulce grafu. Další z možností, je zde pak pomocí potenciometru na panelu ovlivňovat „ladit“ parametry budícího signálu. [4]

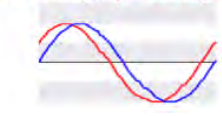


Obr.8. Obrazovka – Vykreslování křivek

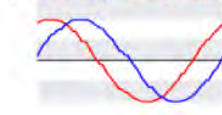
$$x = 10 \cdot \sin(t) \quad y = 10 \cdot \sin(t)$$



$$x = 10 \cdot \sin(t) \quad y = 10 \cdot \sin(t+0,5)$$



$$x = 10 \cdot \sin(t) \quad y = 10 \cdot \sin(t+1)$$

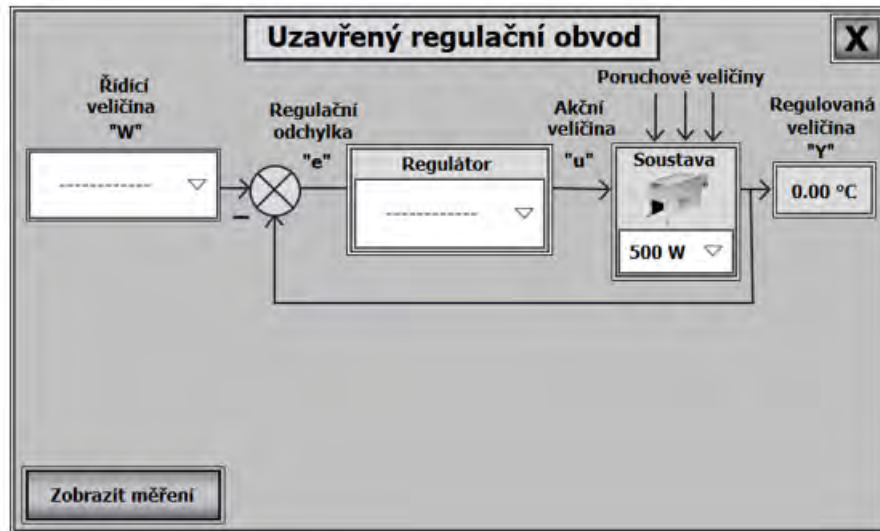


2.2.3. Postup studenta

Student si tak na začátku zvolí řídicí „budící“ hodnotu teploty, a to z několika možností pomocí generátoru signálu a poté má za úkol zjistit odezvu neznámé soustavy. Pomocí Lissajuse, či časových průběhů (tedy grafických metod) pak vyhodnocuje chování horkovzdušného tunelu na měnící se požadovanou hodnotu. Grafy jsou ukládány do excelu, z kterého pak student zpracovává protokol. Pro pracování s grafy je popsán podrobný návod v manuálu pro tuto laboratorní úlohu.

2.3. Uzavřený regulační obvod

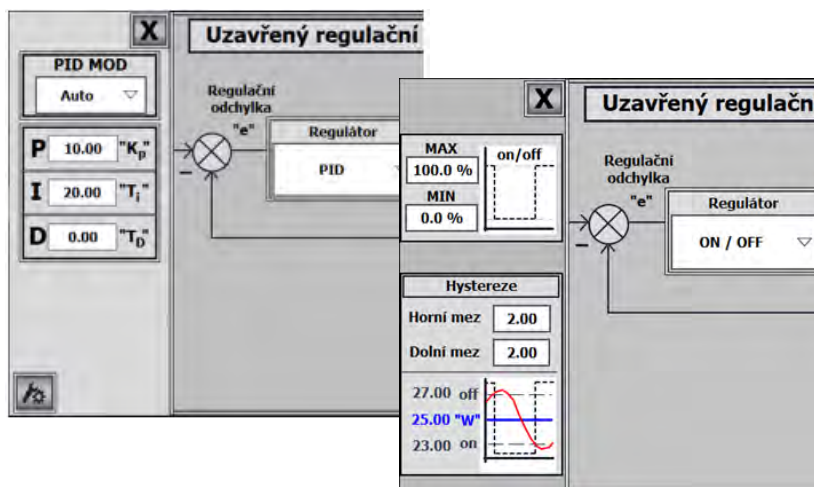
Jedním ze základních znalostí, jakožto studenta automatického řízení, je nastavování a rozpoznání různých typů regulátorů. A o tomto taky tato úloha bude. Úkolem studenta, je zde seřadit nastavování teploty horkovzdušného tunelu dle žádané hodnoty.



Obr.9. Obrazovka – Uzavřený regulační obvod

2.3.1. Možnosti volby regulátoru

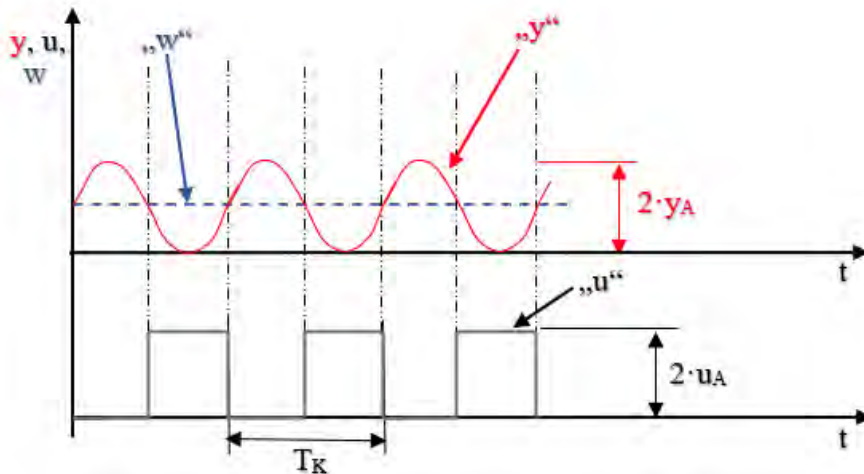
Jako prvotní krok je důležité zvolit správný regulátor. S tím pak studenti pracují a správně jej parametrizují pomocí různých metod. S největším zastoupením v praxi se dá považovat PID regulátor (či jeho modifikace PI, PD, kdy nevyužíváme všechny tři složky). Sice se již nepoužívá jako v dřívějších letech ve formě hardwaru, tedy určitého zařízení, ale dnes se vyskytuje spíše ve formě programové a to tak, že je jeho algoritmus naprogramován v PLC. Velice často používaný regulátor je také typu ON/OFF. Jedná se o primitivnější metodu dvupolohové regulace, avšak často velmi dostačující (např. pro ohřev vody v bojleru, napouštění a vypouštění nádrží). Oba tyto regulátory zde máme a pro laboratorní úlohu využíváme. [6]



Obr.10. Volba regulátoru

2.3.2. Typy řízení a seřizování regulátoru

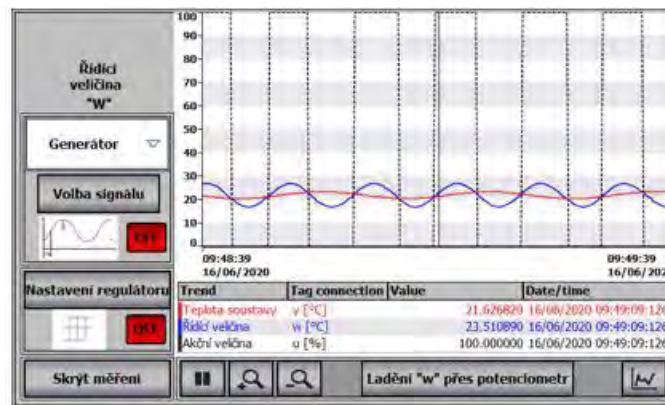
V práci jsou také osvětleny možné přístupy k řízení výstupu daného regulátoru (PWM, či analogový způsob). A to jejich výhody i nevýhody. Nejdůležitějším je zde však metodika nastavování regulátoru. Těchto metod je obrovské množství, a proto jsem popsal pouze pět možných přístupů. Každá metoda je popsána krok, po kroku i s patřičnými grafy a tabulkami. Doporučenou cestu pro studenty jsem však zvolil **Metodu relé Åström-Hägglund**. Tímto způsobem mají možnost využít obou typů regulátoru a pro naši úlohu, je to jedna z nejlepších možností nastavení správného řízení teploty. [3, 6]



Obr.11. Metodika seřizování[3]

2.3.3. Postup studenta

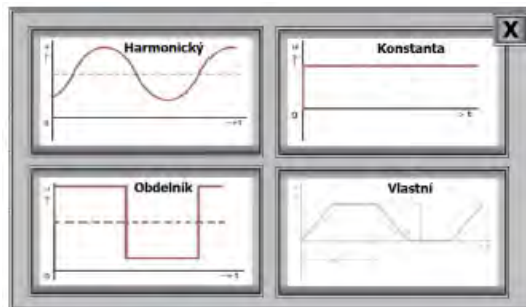
Úloha je dělána tak, aby byla co nejvíce intuitivní. Student si tak na začátku zvolí řídicí „požadovanou“ hodnotu teploty, a to z několika možností pomocí generátoru signálu. Poté přidá patřičný regulátor a může zvolit výkon pece. Pomocí TrendView „Zobrazit měření“ pak vyhodnocuje, jestli PLC nastavuje teplotu dobře, nebo je třeba něco upravit. Graf je přitom ukládán do excelu, z kterého pak student zpracovává protokol. Pro zvědavější studenty jsem zde připravil možnost auto-tuningu a rozsáhlejšího nastavení regulátoru (jenž je nad rámec požadovaných znalostí). Hodnoty řídicí veličiny jdou navíc měnit i po-mocí potenciometru, který je fyzicky přítomen na ovládacím panelu. Díky vykreslování na operátorském panelu tak můžou studenti v reálném čase vidět, jak jejich nastavování mění chování stroje.



Obr.12. Probíhající měření

3. Generátor signálu

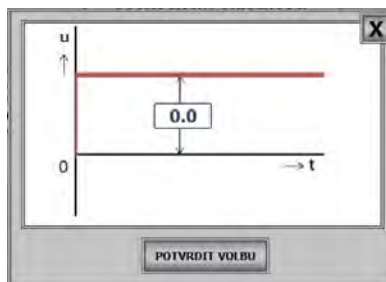
V druhé a třetí laboratorní úloze bylo třeba generovat určitý signál. V případě frekvenčních vlastností to byl „signál budící“ a pro uzavřený regulační obvod zase „žádaná hodnota“. Z tohoto důvodu jsem se rozhodl vytvořit generátor, který bude mít na výběr ze čtyř plně nastavitelných signálů. Jedná se vlastně o funkční blok FB1, jenž je volán v hlavním organizačním bloku OB1. Jeho vnitřní struktura je napsaná v LAD a SCL jazyce a je ovládán skrze grafické rozhraní HMI ovládacího panelu. Podrobný popis naprogramování je pak vysvětlen v bakalářské práci. Specificky pak pro signál Vlastní, bylo třeba vymyslet vykreslování dané křivky již přímo v editoru, aby uživatel viděl, co přesně do PLC posílá. Hlavní snahou byla jednoduchost zadávání hodnot a jejich přehlednost.



Obr.13. Menu - Generátor signálu

3.1. Konstanta

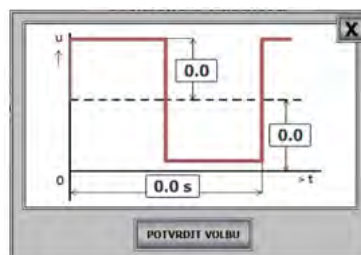
Jedná se o nejjednodušší druh signálu, pouze zde přečteme zadanou hodnotu v HMI, která se však nahraje na výstup generátoru až po stlačení tlačítka „POTVRDIT VOLBU“.



Obr.14. Generátor signálu – Konstanta

3.2. Obdélník

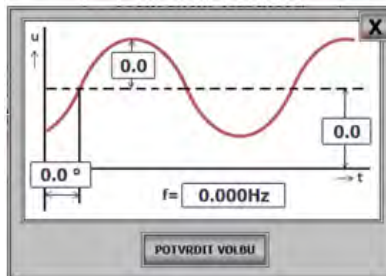
Tento signál je tvořen třemi volenými parametry v HMI (Amplitudou, Offsetem a Periodou). Generátor poté co dostane „POTVRDIT VOLBU“, periodu rozpůlí a v první polovině hodnotu amplitudy k offsetu přičítá a v druhé polovině zase odečítá, přičemž na konci periody časovač vynuluje a jede znovu.



Obr.15. Generátor signálu – Obdélník

3.3. Harmonický

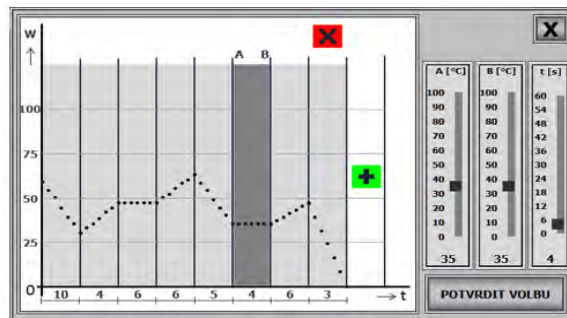
Zde máme sinus, který pracuje přes matematický blok s parametry (Amplituda, Frekvence, Posuv a Offset). Ty spolu tvoří rovnici: $Y = A \cdot \sin(2 \cdot \pi \cdot f \cdot t + f_0) + b$
 Veškeré parametry jsou zde vloženy skrze HMI a následným „POTVRDIT VOLBU“ nahrány do generátoru. Fázový posuv je zde přepočten, aby šel zadávat ve stupních.



Obr.16. Generátor signálu – Harmonický

3.4. Vlastní

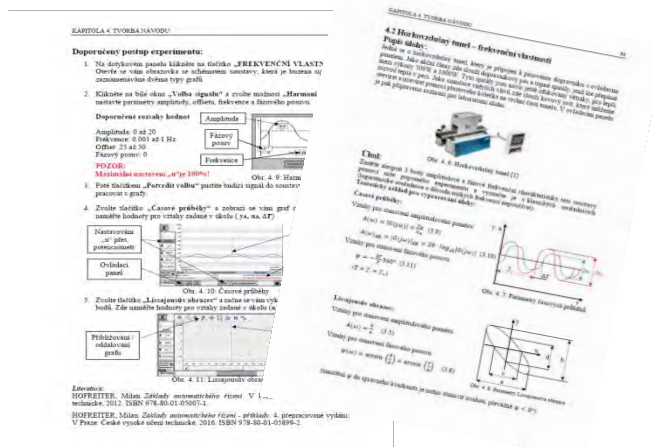
Generátor může mít celkově až devět nastavitelných polí, kde každé pole má maximální periodu trvání 60ti sekund. Tvar signálu je zde tvořen pomocí dvou posuvných jezdců „A, B“, které mají maximální nastavení 100°C). Periodu „t“ také nastavujeme pomocí jezdců, ale je zde pouze v číselné reprezentaci a na tvar křivky signálu v tomto editoru nemá žádný vliv.



Obr.17. Generátor signálu – Vlastní

4. Tvorba návodu

Pro každou úlohu je pak vytvořen podrobný návod postupu v laboratoři. Na každou jsou vymyšleny tři úkoly, z kterých se pak vypracovává protokol. Kromě zadání a popisu, je pak přesně popsáno co, a jak se zadává, nebo měří, aby měli studenti v poměrně krátkém časovém úseku laboratorního cvičení vůbec šanci vše stihnout.



Obr.18. Návody k úlohám

5. Závěrečné informace

Pro každou úlohu jsem vytvořil jak samotný program, tak grafické rozhraní v HMI, společně s podrobný návodem do laboratoře. Jelikož samotné FESTO stanice již měly určitý algoritmus v sobě, tak jsem ho zachoval a pouze vše zablokoval. Je zde tak kdykoliv možnost přejít do původního „ukázkového programu“ chodu stanice přímo od výrobce, a to skrze tlačítko na hlavní obrazovce. Přidal jsem zde i další funkce, jakož jsou různé nastavení limitů zadávaných parametrů, simulace a další. Které však nejsou bez zadání hesla přístupny, a to jak z důvodu bezpečnosti, tak i proto, aby se studenti nepustili do něčeho, co po nich není v zadání požadováno. Jsou to tedy spíše funkce pro pokročilejší studenty, když by jim zbyl čas navíc.

Poděkování

Tato práce byla podpořena grantem Studentské grantové soutěže ČVUT č. SGS19/158/OHK2/3T/12.

Literatura

- [1] Heat Tunnel manual. In: *Festo didactic - CP Factory/Lab - Application modules* [online]. Denkendorf: Festo Didactic SE, c2019 [cit. 2019-02]. Dostupné z: <https://ip.festo-di-dactic.com/InfoPortal/CPFactoryLab/data/download.php?model=CP-AM-HEAT&file=manual&lang=en>
- [2] CP Lab Conveyor circuit diagrams. In: *Festo didactic - CP Factory/Lab - Base modules* [online]. Denkendorf: Festo Didactic SE, c2018 [cit. 2018-26-11]. Dostupné z: <https://ip.festo-didactic.com/InfoPortal/CPFactory-Lab/data/download.php?model=CP-L-CONV&file=cir-cuit-diagram&lang=en>
- [3] ÅSTRO□M, Karl J., Tore HA□GGLUND a Karl J. ÅSTRO□M. *PID controllers*. 2nd ed. Research Triangle Park, N.C.: International Society for Measurement and Control, c1995. ISBN 1-55617-516-7.
- [4] HOFREITER, M.: *Základy automatického řízení*. V Praze: České vysoké učení technické, 2012. ISBN 978-80-01-05007-1.
- [5] *Croline Teti*, Introduction to Lissajous Figures. In: *Acade-mia.edu* [online]. University fo California, c2018 [cit. 2018]. Dostupné z: https://www.acade-mia.edu/37261286/Introduction_to_Lissajous_Figures_Exploring_Variable_Effects_in_a_System_of_Pa-rametric_Equations
- [6] VÍTEČKOVÁ, M., VÍTEČEK, A.: *Vybrané metody seřizování regulátorů*. Ostrava: Vysoká škola báňská - Technická univerzita, 2011. ISBN 978-80-248-2503-8.
- [7] Programming guideline for S7-1500. In: <https://support.industry.siemens.com/> [online]. Siemens c2017 [cit. 2017-03]. Dostupné z: <https://support.industry.siemens.com/cs/document/90885040/programming-guide-line-for-s7-1200-s7-1500?dti=0&lc=en-CZ>
- [8] ČSN EN 61131-3. *Programovatelné řídicí jednotky – Část 3: Programovací jazyky*. 2. vyd. Praha: ÚNMZ, říjen 2013

PŘEDSTAVENÍ PROJEKTU PROCRAFT A SPOLUPRÁCE S LETECKÝM MUZEEM KBELY

Ivana Oswaldová¹, Jaromír Fišer², Tomáš Vyhliďal³

¹ Ústav přístrojové a řídicí techniky, fakulta strojní, ČVUT v Praze, ivana.oswaldova@fs.cvut.cz

² Ústav přístrojové a řídicí techniky, fakulta strojní, ČVUT v Praze, jaromir.fiser@fs.cvut.cz

³ Ústav přístrojové a řídicí techniky, fakulta strojní, ČVUT v Praze, tomas.vyhliďal@fs.cvut.cz

Abstrakt: Projekt PROCRAFT je realizován v rámci projektu JPICH Cultural Heritage. Je zaměřen především na ochranu leteckých artefaktů z druhé světové války, soustředí se především na konzervaci a restaurování slitin hliníku a vytvoření vhodných podmínek k jejich uchování. V rámci spolupráce na řešení projektu mezi ČVUT a Leteckým muzeem Kbely byly vybrány objekty vhodné k případné ochraně (např. bitevní letoun Iljušin Il-2m3) a budovy (hangáry), v nichž budou měřeny podmínky vnitřního prostředí a navrženo možné řešení řízení mikroklimatu.

Klíčová slova: PROCRAFT, řízení mikroklimatu, hangár, Culture Heritage, letecké artefakty

1 Úvod

Druhá světová válka znamenala velké lidské i materiální ztráty, přinesla však také obrovský pokrok ve vývoji především vojenské techniky. Značný vliv měla také na pokrok v rámci leteckého výzkumu a průmyslu. Projekt PROCRAFT (PROtection and Conservation of Heritage AirCRAFT) je zaměřen především na toto období. Letecké artefakty se nacházejí v různém stavu, různých podmínkách i velikostech, jsou vlastněny rozličnými subjekty. Úkolem projektu je vytvoření pokynů pro restaurování a konzervaci slitin hliníku leteckých artefaktů pro neprofesionální aktéry.

V rámci tohoto projektu vznikla úzká spolupráce mezi ČVUT a Leteckým muzeem Kbely, které nám poskytlo vhodné objekty pro další možnosti výzkumu a potřebné prostory k případnému návrhu vhodných mikroklimatických podmínek.

2 PROCRAFT (PROtection and Conservation of Heritage AirCRAFT)

Projekt je realizován v období 2. 11. 2020 – 1. 11. 2023 v rámci JPICH Cultural Heritage projektu, který získal dotační podporu z EU Horizon 2020 Research and innovation programme. Projekt je zaměřen na vytvoření inovativních postupů a řešení pro každý klíčový krok při konzervaci letadel. Zároveň propojuje více aktérů operačního řetězce jako jsou obnova, údržba a vytvoření vhodných okolních podmínek. Zasahuje do různých oborových oblastí přes archeologii, historii, hmotné dědictví, objekty, materiály, ochranu, preventivní konzervaci až po technologicko – vědecké procesy a management kulturního dědictví. Koordinátorem celého projektu je Arc'Antique laboratory (GPLA, Nantes, Francie) a hlavními partnery: CNRS - Centre d'élaboration de matériaux et d'études structurales (Toulouse, Francie), University of Bologna (UNIBO, Itálie) and University of Ferrara (UNIFE, Itálie) a ČVUT v Praze. [1]

2.1 Hlavní cíle projektu

Cílem projektu je především vytvořit dílčí postupy pro údržbu leteckých artefaktů a přizpůsobit tak restaurátorské – konzervační techniky, použití inteligentních nátěrů pro venkovní ochranu při respektování požadavků ochrany kulturního dědictví, povlakování (potažení či naválcování ochrannou vrstvou), inovativní řešení pro preventivní konzervaci v omezeném nebo částečně uzavřeném prostředí a poskytnutí návodu pro obnovu a konzervaci slitin hliníku pro neprofesionální subjekty. [1]

Tým při ČVUT si klade především za cíl vytvořit rozhodovací systém pro ochranu leteckých památek, využití matematických modelů (zejména redukovaných modelů CFD) při analýze dopadu prostředí na životnost leteckých komponent, dále pak použití senzorů a zpracování velkých dat. Dle předchozí analýzy navrhne energeticky účinný systém HVAC pro potenciální možnou instalaci.

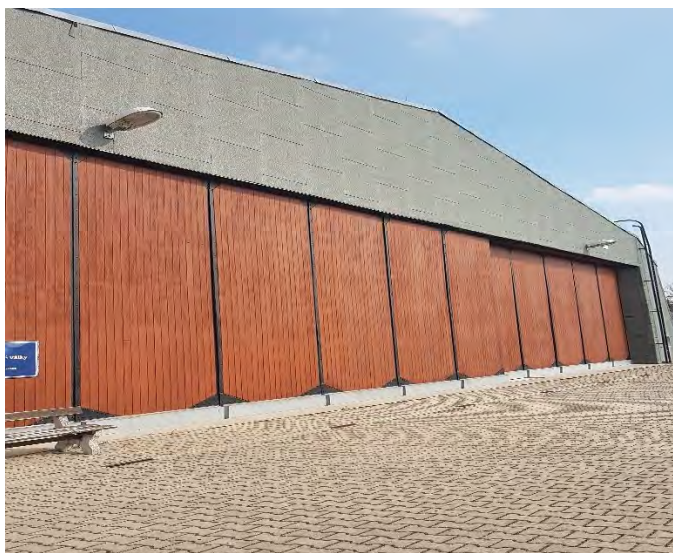
3 Letecké muzeum Kbely

V současné době Letecké muzeum Vojenského historického ústavu zahrnuje 6 hangárů se stálými expozicemi, sál pro konání příležitostných výstav a rozsáhlé venkovní prostory. Celkově je zde vystaveno 143 letadel, 50 motorů a desítky dalších částí letadel. [2]

S ohledem na vystavované objekty a možným návrhem vhodného mikroklimatu byly vybrány 4 typově různé hangáry z hlediska jejich konstrukce a materiálového složení.

3.1 Hangár gen. Karla Janouška (letectví 2. světové války)

Jedná se o hangár typu Wagner o rozměrech 30,8 x 32,0 m, který prošel v roce 2008 generální opravou. Tvoří jej ocelová nosná konstrukce opláštěná dřevem a lepenkou. Hangár není vytápěn ani jinak temperován, ventilace je zde zajištěna pouze 4 větracími otvory u stropu a otvory ve vratech hangáru a únikových dveřích. [2]



Obr. 1.: Hangár gen. Karla Janouška (letectví 2. světové války) – č. 87

Jsou zde vystaveny zejména letouny a další předměty vztahující se k činnosti československých letců v období 2. světové války. Jedná se především o sovětské letouny – bitevní Iljušin Il-2m3 a stíhací Lavočkin La-7, britské cvičné stroje De Havilland DH-82 Tiger Moth a Norduyn Harvard Mk.II., francouzský cvičný Morane Saulnier MS-230, německý proudový letoun Messerschmitt Me 262B a německá řízená puma Fritz X. [2]

Iljušin Il-2m3 Šturmovik

Tento sovětský bitevní letoun byl zařazen do projektu PROCRAFT rovněž díky jeho konstrukci a interakci různých materiálů, především kovů (Al, Fe) a dřeva. Byl navržen již v roce 1938 Sergejem Vladimirovičem Iljušinem, jeho sériová výroba započala v roce 1941. Jedná se o jednomotorový dvousedadlový dolnoplošník se zatahovacím podvozkem. Během 2. světové války patřil k základním typům bitevních jednotek sovětského letectva, včetně polských, československých a jugoslávských útvarů, které vznikly v SSSR. Českoslovenští piloti používali stroje Il-2 u 3. bitevního pluku v rámci 1. československé smíšené letecké divize v SSSR. [3]

Vystavovaný exponát používali u 3. československého bitevního leteckého pluku, který byl součástí 1. československé smíšené letecké divize v SSSR. Při Ostravské operaci 20. dubna 1945 byl sestřelen a jeho posádka byla nucena nouzově přistát mezi frontovými liniemi. Poté byl rozebrán a převezen zpět k pluku. Po skončení války sloužil v československém vojenském letectvu až do roku 1948. Poté byl předán do Národního technického muzea. [3]

V roce 1966 byl nalezen v těžce poškozeném stavu na letišti Praha – Kbely. Prošel renovací ve vojenských opravárnách v Líních, v letech 2005 a 2006 pak proběhly další opravy u firmy BMZ Air Service, letoun byl opatřen autentickým zbarvením z doby služby u 3. československého bitevního pluku. [4]

Letoun je smíšené konstrukce. Přední část (motorové lože a prostor posádky) se skládá z pancéřových plátů, zadní část je dřevěno-ocelové konstrukce (trup potažen tvarovanou překližkou), ocasní plochy obsahují 2 nosníky a 10 žeber z duralu, jeho kovová konstrukce je potažena plátnem, křídlo tvoří dřevěno-ocelová konstrukce. Ocelové části byly kompletně mechanicky očištěny od koroze, natřeny základní akrylovou uretanovou barvou a ošetřeny akrylovou barvou. Dřevěné části jsou naimpregnovány epoxidovým lepidlem, poté natřeny proti dřevokazným škůdcům, ošetřeny základní akrylovou barvou na dřevo a povrchově upraveny akrylovou barvou. Elektronické součástky jsou ošetřeny kyselinou křemičitou, následně opatřeny základním nátěrem a povrchově upraveny barvou. Letoun je po opravě kompletní, nefunkční, ve velmi dobrém technickém stavu. [4], [5]



Obr. 2.: Iljušin Il-2m3 Šturmovik, Motor: AM-38F o výkonu 1 250 kW; rozpětí: 14,60 m; délka: 11,65 m; hmotnost prázdného letounu: 4 525 kg; vzletová hmotnost: 6 360 kg; maximální rychlost: 404 km/h; dostup: 3 500 m; dolet: 765 km [3]



Obr. 3.: Iljušin Il-2m3 Šturmovik v roce 1966, Letiště Kbely [4]

3.2 Proudové letouny 1. generace (2. světová válka)

Hangár typu Pícha je opatřen kovovou konstrukcí s plechovým opláštěním o rozměrech 30,8 x 32,0 m. Je to standardní hangár československé konstrukce ze 30. letech minulého století. Není opatřen okny, větrání probíhá pouze přirozeně vraty hangáru a 4 větracími otvory. [2]

Jsou zde vystaveny především proudové letouny 1. generace, z nichž některé vznikly již v letech 2. světové války a zbývající záhy po jejím skončení: letouny Avia S-92 (v Československu sestavený Messerschmitt Me 262A), britské stíhačky Gloster Meteor F Mk.8 a De Havilland Vampire FB Mk.6, sovětský stíhací Jakovlev Jak-17 a americký cvičný Lockheed T-33. [2]



Obr. 4.: 3.2 Hangár č. 17 – proudové letouny 1. generace

3.3 Hangár plukovníka Jaroslava Janečky (československé letectvo 1918–1924)

Jeden z prvních postavených hangárů ve Kbelích je hangár typu Hart, byl sem převezzen z Chebu v roce 1919. Během 2. světové války sloužil jako kinosál, na který ho předělala německá armáda, takto byl používán až do 80. let 20. století. V roce 2003 prošel nákladnou rekonstrukcí (podařilo se zachránit některé původní trámy a části hangáru). Jeho konstrukce je dřevěná s dřevěným lepenkovým opláštěním se 2 větracími otvory. [2]

K nejvýznamnějším exponátům patří francouzský stíhací letoun SPAD VII z období 1. světové války a první vojenský letoun československé konstrukce Letov Š-2. Expozice je doplněna ukázkou letecké dílny včetně dobového vybavení a součástí letadel. [2]



Obr. 5.: 3.2 Hangár č. 88 – československé letectvo 1918–1924

3.4 Československé letectvo po roce 1945

Na místě 4 bombardováním zničených hangárů byl po skončení 2. světové války postaven tento dosud největší expoziční hangár typu Pícha (rozměry: 86,15 x 30 m). Až do roku 1988 byl jediným expozičním hangárem muzea. Hangár se skládá z kovové a zděné konstrukce, uvnitř je zateplen sádrokartonem – mezi ním a zdí jsou umístěny průfuky v horní i spodní části. Vstupní vrata hangáru jsou po celé délce horní části opatřena okny (mezi nimi – uprostřed – jsou umístěny větrací žaluzie). [2]

Je zde vystaveno téměř 60 letounů a množství dalších sbírkových předmětů, dokumentujících vývoj československého vojenského, dopravního a sportovního letectví po r. 1945. K vystavovaným patří např.: stíhačí letouny Avia S a CS 199, dopravní Aero C-3A, bitevní Avia B-33 a některé další stroje, Jak-23, řada migů počínaje MiGem 15 a 17 přes MiG-19 až po MiG-21 a MiG-23, československé konstrukce Aero Ae-45, Let L-200 Morava, československé vrtulníky, známé stroje Zlin a vrchol československého leteckého průmyslu Aero L-29 Delfín a L-39 Albatros, meziválečné tradice československého letectví: akrobatický stroj Avia Ba-122, jediný obojživelný letoun ČSA – Saunders Roe Cloud, návratová kabina kosmické lodi Sojuz 28 spolu se skafandrem Vladimíra Remka. [2]



Obr. 6.: 3.2 Hangár č. 40 – československé letectvo po roce 1945

4 Závěr

Spolupráce na projektu s Leteckým muzeem Praha nadále probíhá a v současné době jsou v jednotlivých expozičních hangárech umístěny sensory pro monitorování vnitřního mikroklimatu. Současně jsou stahována a ukládána data z Českého hydrometeorologického ústavu monitorující podmínky vnějšího prostředí v hodinových intervalech během celého dne. Tato data budou dále zpracována a při analýze dopadu prostředí na životnost leteckých komponent bude využito matematických modelů. Dle předchozí analýzy bude navržen energeticky účinný systém HVAC.

Poděkování

Práce byla podpořena evropským projektem PROCRAFT

Reference

- [1] Joint Programming Initiative on Cultural Heritage Research Hub [online]. [cit. 15. 9. 2021] Dostupné z: <https://www.heritageresearch-hub.eu/project/procraft/>
- [2] KHOL, M., RAJLICH, J., SÝKORA, J., ŠÍMA, J.: Katalog Leteckého muzea VHÚ Praha. Ministerstvo obrany ČR – VHÚ Praha, 2020, 2. doplněné vydání.
- [3] RASTREMIN, O.: Iljušin Il-2. JAKAB Publishing
- [4] SÝKORA, J.: První fáze renovace letounu Iljušin Il-2. Historie a vojenství 2/2006, s. 120-124
- [5] OVČÁČÍK, M., SUSÁ, K.: Ilyushin Il-2 Shturmovik. 4+ Publishing Co., Praha 2006

DOKTORANDSKÁ SEKCE

Data-driven model discovery using Sparse Identification of Nonlinear Dynamics

Kryštof Bystřický*¹

¹ *Czech Technical University in Prague, Faculty of Mechanical Engineering, Department of Instrumentation and Control Engineering, Technická 4, 166 07 Prague 6, Czech Republic, Krystof.Bystricky@fs.cvut.cz*

Abstract

The problem of obtaining models describing the dynamics of a certain system is current across many industries. The model can be created from the physical principles of the system, this approach however requires a developed theory of the underlying system. An alternative approach is model identification from measured data, which is often the only viable approach for complex systems without a developed theory. In this paper, I apply a method called Sparse Identification of Nonlinear Dynamics (SINDy), which utilizes both approaches, identification from data using at least some limited knowledge of the system. The method will be used to identify a nonlinear model of a pendulum-cart system from data with additive noise. The paper also describes methods for filtering and numerical differentiation of the measured signals and a new method for model selection for sparse models.

Key-words: Machine learning; system identification; sparse regression; dynamical systems; numerical differentiation; spectral methods

1. Introduction

The problem of discovering mathematical models describing real-world phenomena is relevant since the start of the scientific revolution. Having the ability to quantitatively describe reality opens new opportunities in both engineering and sciences. This paper focuses on dynamical models, which are mathematical models that describe the behaviour of a system in time.

The model discovery process is about identifying and describing patterns in the measurements. Traditionally, this relied heavily on expert knowledge and intuition. An expert had to notice the patterns, be able to reduce them into mathematical form and create the mathematical model describing the observations. The emergence of computers, increasing computational power, and advances in machine learning enable an alternative approach. Many machine learning methods, such as neural networks, are capable of describing dynamical systems purely from data. These models are however difficult to interpret (black box models), and they often don't respect physical constraints on the model dynamics. These issues justify the recent interest in physics-informed machine learning, which combines expert knowledge of physics and machine learning.

The identified model then has to be evaluated. According to the philosophy of Occam's razor, also known as the law of parsimony, a good model should be as simple as is necessary to accurately describe the observations. The mathematical model should also be able to generalize to data that haven't been seen during the model creation process. Physics-informed machine learning constrains the space of possible models, which prevents overfitting and therefore promotes the model's generalization capability.

2. Sparse Identification of Nonlinear Dynamics

Sparse Identification of Nonlinear Dynamics [1], or SINDy, is a method that utilizes both expert knowledge and machine learning. The expert's input is in reducing the space of possible models by picking a set of functions that might describe the real system dynamics. Sparsity-promoting regression is then used to create models from measurement data. By promoting sparsity, the functions that turn out to not be relevant in the underlying dynamics will be completely omitted

in the final model. Sparsity therefore means that the final model contains as few functions as possible. This follows the law of parsimony, which in the context of mathematical models means that we should value the simplicity of our models as well as their accuracy.

2.1. Regression for dynamical systems

2.1.1. General formulation

Dynamical systems are described by a system of ordinary differential equations (ODEs). Systems of ODEs are generally described by

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)). \quad (1)$$

The vector $\mathbf{x}(t) \in \mathbb{R}^d$, where d is the number of state variables, defines the state of the system at time t . The vector function \mathbf{f} describes the effect of natural dynamics at time t . The external input vector $\mathbf{u}(t) \in \mathbb{R}^b$, where b is the number of external inputs, and \mathbf{g} is the vector function describing the effect of external forcing on the system. Note that \mathbf{g} is a function of both state and input, as an input might have different effect on the system depending on its current state.

2.1.2. Matrix-vector formulation

For the SINDy [1] algorithm to be directly applicable, every single ODE \dot{x}_i from $\dot{\mathbf{x}} = [x_1, \dots, x_d]^T$ must be expressible as a linear combination of functions of \mathbf{x} and \mathbf{u}

$$\dot{x}_i(t) = \xi_1 \theta_1(\mathbf{x}(t), \mathbf{u}(t)) + \dots + \xi_m \theta_m(\mathbf{x}(t), \mathbf{u}(t)) \quad (2)$$

where ξ are scalar parameters and m is the total number of candidate functions. Simplifying the notation so that $\theta_i(t) = \theta_i(\mathbf{x}(t), \mathbf{u}(t))$, the equation (1) can be formulated in vector notation as

$$\dot{x}(t) = [\theta_1(t) \quad \dots \quad \theta_m(t)] \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_m \end{bmatrix} = \Theta \boldsymbol{\xi} \quad (3)$$

where the matrix Θ is the set of candidate functions and the vector $\boldsymbol{\xi}$ is the vector of coefficients of those functions. The regression task is therefore to find a vector of coefficients $\boldsymbol{\xi}$ that represents \dot{x}_i using functions from Θ .

Because the continuous functions $\theta(t)$ are only an abstraction that cannot be worked with numerically, we must approximate them with a finite number of measurements $\theta[k]$, where k is the time-sample index. Note that $\theta[k]$ are vectors of *measurements*, so they're far from a perfect substitute for the actual functions $\theta(t)$. Whether measurements $\theta[k]$ represent the underlying function $\theta(t)$ well doesn't depend only on the sampling frequency or the number of samples, but also on the way they've been generated. If, for example, we had a function $\theta(t) = x_1 \sin(x_2) + 1$, but the state variable x_2 was kept constant at 0 during the experiment, then the set of measurements $\theta[k]$ would be a terrible representative of $\theta(t)$, because it wouldn't describe the effects of any of its variables x_1 and x_2 .

The approximation of (3) therefore has the form

$$\dot{x}_i[k] = \begin{bmatrix} \theta_1[k] & \dots & \theta_m[k] \end{bmatrix} \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_m \end{bmatrix} = \Theta(\mathbf{X}, \mathbf{U}) \boldsymbol{\xi}. \quad (4)$$

$\Theta(\mathbf{X}, \mathbf{U}) \in \mathbb{R}^{N \times m}$ is called the *function library*, it has m columns representing the candidate functions, and N rows representing time-samples. The columns $\theta_i[k]$ representing the candidate functions are computed from state measurements \mathbf{X} and input measurements \mathbf{U} . The state derivative measurements $\dot{\mathbf{X}}$ must be computed from the state measurements \mathbf{X} numerically, using for example spectral differentiation described later in this paper.

The original paper [1] didn't mention identification of systems with external inputs \mathbf{u} as in the case above. Another paper was released shortly after, where an extension named SINDYc [2] included the control input \mathbf{u} in the function library. The extension essentially doesn't make a distinction between the inputs \mathbf{u} and states \mathbf{x} , they can be treated equally with no extra cost. The inputs \mathbf{u} will not appear in the equations later in the paper, since adding them is trivial.

2.2. Sparse regression

2.2.1. Description

The equation (4) represents the general regression problem

$$\mathbf{Ax} = \mathbf{b}. \quad (5)$$

The objective is to find a solution vector \mathbf{x} (not to be confused with the state vector \mathbf{x}), or $\boldsymbol{\xi}$ in the ODE formulation. There are many algorithms for solving this problem, many of them put some constraints on the solution \mathbf{x} . Following the law of parsimony, we want to find a solution \mathbf{x} that is *sparse*, meaning it has as few non-zero elements as possible. Note that when a solution coefficient ξ_i is 0, its respective candidate function $\theta_i(t)$ isn't present in the final model.

2.2.2. Sequentially thresholded least squares

The sequentially thresholded least squares (STLS) algorithm used in the original SINDy [1] paper uses standard least-squares regression

$$\arg \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|^2 \quad (6)$$

and then sets all elements $x \in \mathbf{x}$ that are below a defined hyperparameter threshold λ to 0. These two steps are then repeated multiple times until the solution \mathbf{x} no longer changes between iterations. In the ODE formulation (4), the algorithm sequentially reduces the number of considered candidate functions θ . On the first iteration, it works with the full function library $\Theta \in \mathbb{R}^{N \times m}$ and produces a non-sparse solution $\boldsymbol{\xi}$, whose elements ξ_i which are below the threshold λ are set to 0. If an element ξ_i is set to 0, its respective candidate function θ_i is dropped from Θ and won't be considered in the next iterations. The number of candidate functions usually drops significantly after the first iteration. The condition number κ , representing the well-posedness of the regression problem, is defined as a ratio of the biggest and the smallest singular value

$$\kappa = \frac{\max \sigma(\Theta)}{\min \sigma(\Theta)} \quad (7)$$

The condition number κ , is very high at the first iteration, meaning the problem is ill-posed and the solution isn't reliable. In the later iterations, as columns of Θ are dropped, κ also usually decreases to acceptable values. The STLS method produces sparse solutions \mathbf{x} while preserving the numerical robustness and low time complexity of least-squares algorithms.

2.2.3. Sequentially energy thresholded least squares

The STLS algorithm sequentially drops candidate functions θ_i based on their respective parameter values ξ_i under the assumption that low values of ξ_i indicate that the respective candidate function θ_i has a small effect on the target variable. However, when working with unnormalized measurements $\theta[k]$, this isn't necessarily true. Interpreting the measurements $\theta[k]$ as signals, a high energy signal $\theta_i[k]$ with a low coefficient ξ_i might have a higher effect than another signal $\theta_j[k]$ with a relatively higher coefficient ξ_j . To deal with this issue, I modified the STLS algorithm so that the thresholding is done based on the signal's total implied energy, defined for every candidate function and parameter vector $(\theta[k], \xi)$ pair as

$$E(\xi, \theta[k]) = \sum_{k=0}^{N-1} (\xi \theta_k)^2. \quad (8)$$

This energy E is calculated for every candidate function present in the function library, the highest energy $\max\{E(\xi, \theta[k])\}$ then becomes a baseline from which the threshold λ is calculated as

$$\lambda = \max\{E(\xi, \theta[k])\} \cdot \lambda_R \quad (9)$$

where λ_R is a hyperparameter setting the relative energy ratio between the lowest acceptable energy signal and the maximum energy signal. The candidate functions θ whose implied energies are lower than λ are then dropped from Θ and their respective coefficients ξ set to 0.

2.3. Implicit SINDy

According to the equation (1), the standard SINDy method cannot be used if the dynamical system is described by a rational function in the general form

$$\dot{\mathbf{x}}(t) = \frac{\mathbf{f}(\mathbf{x}(t))}{\mathbf{h}(\mathbf{x}(t))}. \quad (10)$$

In this case, the dynamics cannot be reduced into a linear combination of functions as before in (2). An extension called implicit-SINDy [3] can deal with this problem. The system of ODEs is transformed into an implicit form by multiplying both sides by the function \mathbf{h}

$$\dot{\mathbf{x}}(t) \mathbf{h}(\mathbf{x}(t)) = \mathbf{f}(\mathbf{x}(t)) \quad (11a)$$

$$0 = \mathbf{f}(\mathbf{x}(t)) - \dot{\mathbf{x}}(t) \mathbf{h}(\mathbf{x}(t)) \quad (11b)$$

For a single ODE \dot{x}_i from $\dot{\mathbf{x}}$, the equation (10) can again be simplified by using a single symbol θ for all right-hand side functions f , \dot{x} and h .

$$0(t) = \xi_1 \theta_1(\mathbf{x}(t)) + \dots + \xi_m \theta_m(\mathbf{x}(t)) \quad (12)$$

In this formulation, the equation can be again transformed into matrix-vector notation as

$$0(t) = [\theta_1(t) \quad \dots \quad \theta_m(t)] \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_m \end{bmatrix} = \Theta(\mathbf{X}, \dot{\mathbf{X}}) \boldsymbol{\xi} \quad (13)$$

One issue now is that our target function \dot{x} , which we want to model, now has to be embedded within the function library $\Theta(\mathbf{X}, \dot{\mathbf{X}})$. This means that even if we had the solution $\boldsymbol{\xi}$, it wouldn't be as trivial to reconstruct \dot{x} as before, where we simply had to multiply each candidate function by its coefficient and sum the functions. The reconstruction step essentially requires doing the step equations (10), (11a) and (11b) in reverse and the specific operations depend on the chosen structure of the candidate functions. In practice, this can be automated using symbolic math software, in the case of this paper using the SymPy [4] package for Python. Another problem is that the standard regression problem $\mathbf{Ax} = \mathbf{b}$ changed into the homogeneous problem

$$\mathbf{Ax} = \mathbf{0}. \quad (14)$$

The algorithms for solving (14) aren't as robust as for the more general problem (5). The existing methods are very sensitive to noise, making this extension by itself unpractical for real-world applications, where measurement noise is unavoidable.

2.4. Parallel implicit SINDy

Another extension, called SINDy-PI [5], deals with the high sensitivity of implicit-SINDy by picking one of the columns θ_i from $\Theta(\mathbf{X}, \dot{\mathbf{X}})$ and moving it to the left-hand side, effectively transforming the problem back into the more general $\mathbf{Ax} = \mathbf{b}$ formulation. If the chosen candidate function θ_i is in the real target dynamics, the solution $\boldsymbol{\xi}$ will be sparse and the model it generates will likely be accurate. The problem (13) is transformed, using a guess θ_i , into

$$\theta_i[k] = \begin{bmatrix} \theta_1[k] \\ \vdots \\ \theta_{i-1}[k] \\ \theta_{i+1}[k] \\ \vdots \\ \theta_m[k] \end{bmatrix}^T \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_{i-1} \\ \xi_{i+1} \\ \vdots \\ \xi_m \end{bmatrix} = \Theta^i(\mathbf{X}, \dot{\mathbf{X}}) \boldsymbol{\xi}^i \quad (15)$$

where $\Theta^i(\mathbf{X}, \dot{\mathbf{X}})$ is the function library without the i -th function (column) and $\boldsymbol{\xi}^i$ is the solution vector without the i -th coefficient.

The problem (15) is solved for many different left-hand side guess functions θ_i , with different hyperparameter values λ_R for each run. This generates a very large number of models, and requires techniques to sort through them to pick the best ones. When selecting the models, I utilize the fact that when the real model contains some different candidate functions $\theta_i, \theta_j, \theta_k$, then it will likely be identified whenever one of those functions acts as the left-hand side guess. A model that appears consistently therefore is likely to be correct.

Because the method relies on guessing and has to generate a lot of models, it's significantly more computationally difficult. However, the identification method is highly parallelizable, since each model can be identified separately. This high parallelization capability is why the method is called *Parallel Implicit*.

3. Data preprocessing

The SINDy method requires measurement data to identify the dynamics. The dynamics of every generalized coordinate in a mechanical dynamical system are, as a consequence of Newton's second law, described by second-order ODEs. When described in the state-space representation, this leads to two state-space variables for generalized coordinate, one for its current value, one for its rate of change. In this paper, I'll assume that we're measuring only the value of the generalized coordinate. The goal of modeling is to find a model for the acceleration. In other words, the acceleration is the target variable. To train the models, we need to estimate these target variables from data. Both velocities and accelerations can be obtained from position measurements using numerical differentiation.

3.1. Spectral differentiation

Spectral differentiation uses the properties of Fourier transforms to compute derivative estimates from measured data [6]. Because we're working with discrete signals and Fourier transforms are just a mathematical abstraction, we must use Discrete Fourier Transforms (DFTs) instead. DFTs are usually computed using the Fast Fourier Transform (FFT) algorithm, because of its low time complexity.

Assuming a measured discrete signal $x[k]$, its representation in the frequency domain is

$$\hat{x}[v] = \text{DFT}\{x[k]\}, \quad (16)$$

where v is known as the *wavenumber*. The frequency (in $\frac{\text{rad}}{\text{s}}$) ω can be computed from v as

$$\omega[v] = \frac{2\pi v}{\Delta t} \quad (17)$$

where Δt is the sampling period. The convenient property of Fourier transforms is that

$$\mathcal{F}\{\dot{x}(t)\} = i\omega\hat{x}(\omega), \quad (18)$$

meaning the Fourier transform of the function derivative in the time domain is equal to the function itself in the frequency domain multiplied by $i\omega$. Using this property, we can estimate the time-derivative of $x[k]$ as

$$\text{DFT}\{\dot{x}[k]\} = i\omega\hat{x}[\omega], \quad (19)$$

which can be transformed back into the time domain using inverse-DFT

$$\dot{x}[k] = \text{iDFT}\{i\omega\hat{x}[\omega]\}, \quad (20)$$

where $\dot{x}[k]$ is the estimate of the function derivative.

Notice that the property 18 implies that the time-derivative of x is the function itself in the frequency domain passed through a high-pass filter defined by $i\omega$. When our signals are measurements of some physical process, the signal itself has most of its energy in the lower frequencies, while (white) noise has the same energy at all frequencies. This unfortunately means that numerical differentiation decreases the signal-to-noise ratio, which raises requirements on low sensor noise and filtering methods.

3.2. Spectral filtering

An ideal filter has a gain of 1 in the specified frequency range and a gain of 0 outside the range. Traditional filters cannot meet these demands. Spectral filtering techniques use the FFT to calculate the signal's representation in the frequency domain. When we have this representation, we can simply set all the high-frequency coefficients to 0 and then use inverse-FFT to reconstruct the signal in the time domain. Mathematically, setting frequencies outside some defined range is equivalent to multiplying the frequency-domain signal with a square function $f_{sq}(\omega)$.

$$\hat{x}[\omega] = \text{DFT}\{x[k]\} \quad (21a)$$

$$\hat{x}_f[\omega] = f_{sq}(\omega) \hat{x}[\omega] \quad f_{sq}(\omega) = \begin{cases} 1, & \text{if } \omega \leq \omega_{cutoff}. \\ 0, & \text{if } \omega > 0. \end{cases} \quad (21b)$$

$$x_f[x] = \text{iDFT}\{x_f[\omega]\} \quad (21c)$$

When designing filters, the most important design choice is the cutoff frequency. If it's set too low, then we filter out useful information from the signal, but if it's set too high, we don't get rid of the noise. In this thesis, I choose the cutoff frequency based on the signal's periodogram, which is an estimate of the power spectrum density (PSD).

Any signal can be interpreted as a sum of information and noise. Information is the part of the signal that is generated by the process we intend to actually measure. Noise is the part of the signal that's generated by other processes. Many processes of interest generate information signals that are mostly dispersed in the lower frequencies, or in other words, have a relatively small bandwidth. The noise, on the other hand, typically has a far bigger bandwidth. Noise is often (mathematically) modeled as a white noise. White noise, by definition, has infinite bandwidth and a constant power spectrum density. In practice, the noise isn't as "flat" in the power spectrum as an ideal white noise. Instead, it's distributed with some variance around some mean power value. I choose the cutoff frequency automatically from the periodogram, assuming the noise component of the signal is distributed evenly in the power spectrum. I calculate the noise ceiling, the power that should be higher than than noise's power at most frequencies, by calculating the mean power μ and standard deviation σ of the 50% highest frequency components in the signal. The noise ceiling is then defined as $\mu + z\sigma$, where z is a manually set parameter.

Then I smoothed the periodogram using a simple moving average filter and find the lowest frequency at which the measurement signal crosses below the previously calculated noise ceiling. If the signal is intended to be used for numerical differentiation, it should be "over-filtered", meaning the cutoff frequency should be set much lower than the frequency at which the noise ceiling is reached. By defining the cutoff frequency as the frequency at which the noise becomes more powerful than the signal, the high frequency components of the filtered signal would have a signal-to-noise ratio of roughly 1. For numerical differentiation, this isn't good, because it acts as a high-pass filter, and these low SNR high frequencies would therefore throw off the derivative estimates.

The filter settings from the periodograms are shown in the Figure 1. In the first one x_1 , the cutoff frequency is set as the frequency at which the periodogram crosses the threshold (noise ceiling). In the second one, the cutoff frequency is manually offset to the lower frequencies to over-filter the signal for numerical differentiation.

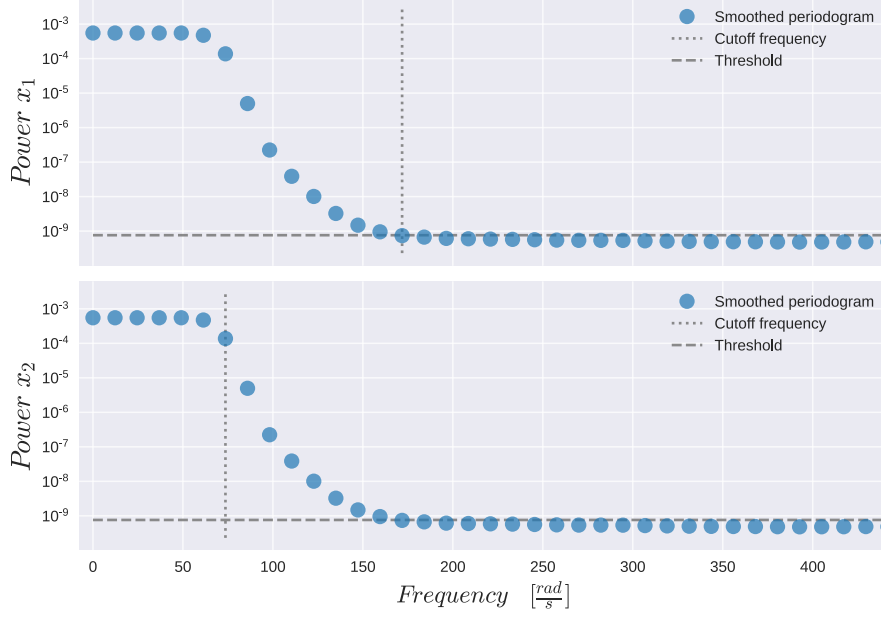


Fig. 1. Cutoff frequency set at the noise ceiling vs. cutoff frequency set before the noise ceiling - “overfiltering”. The threshold represents the noise ceiling, a power value that should be above all noise component power values.

4. Identification of a pendulum-cart model

4.1. Analytical model and simulation

I’m using the SINDy method to identify the nonlinear dynamical model describing the dynamics of a pendulum mounted on a moving cart. The input u into the system is a force acting on the cart. The position of the cart is given by the first state variable x_1 , and the pendulum angle is given by x_2 . The other two state variables, x_3 and x_4 , are the cart velocity and pendulum angular velocity respectively.

I derived an analytical model of the system using Euler-Lagrangian mechanics according to the paper [7]. The model also contains non-conservative friction forces for both the cart and the pendulum. The friction force is a function of velocity in both cases. The cart acceleration and the angular acceleration of the pendulum are given by \ddot{x}_3 and \ddot{x}_4 respectively. These are our target variables, which we want to predict using the state variables and inputs and functions derived from them. The analytical (reference) model is used for simulations that generate data, the objective is to reconstruct the reference model from that data. The two second order ODEs describing the dynamical system are

$$\dot{x}_3 = \frac{A(\mathbf{x}, u)}{B(\mathbf{x}, u)} \quad (22a)$$

$$\begin{aligned} A(\mathbf{x}, u) = & I_1 u(t) + a_1^2 m_1 u(t) - I_1 b_c x_3(t) + a_1^3 m_1^2 \sin(x_2(t)) x_4^2(t) + \\ & -a_1^2 b_c m_1 x_3(t) + a_1 b_1 m_1 \cos(x_2(t)) x_4(t) + a_1^2 g m_1^2 \cos(x_2(t)) \sin(x_2(t)) + \\ & + I_1 a_1 m_1 \sin(x_2(t)) x_4^2(t) \end{aligned} \quad (22b)$$

$$B(\mathbf{x}, u) = -a_1^2 m_1^2 \cos^2(x_2(t)) + a_1^2 m_1^2 + m_c a_1^2 m_1 + I_1 m_1 + I_1 m_c \quad (22c)$$

$$\dot{x}_4 = \frac{C(\mathbf{x}, u)}{D(\mathbf{x}, u)} \quad (23a)$$

$$\begin{aligned} C(\mathbf{x}, u) = & b_1 m_1 x_4(t) + b_1 m_c x_4(t) + a_1 g m_1^2 \sin(x_2(t)) + a_1 m_1 \cos(x_2(t)) u(t) + \\ & + a_1 g m_1 m_c \sin(x_2(t)) + a_1^2 m_1^2 \cos(x_2(t)) \sin(x_2(t)) x_4^2(t) + \\ & - a_1 b_c m_1 \cos(x_2(t)) x_3(t) \end{aligned} \quad (23b)$$

$$D(\mathbf{x}, u) = -a_1^2 m_1^2 \cos^2(x_2(t)) + a_1^2 m_1^2 + m_c a_1^2 m_1 + I_1 m_1 + I_1 m_c \quad (23c)$$

where I_1 are the pendulum's moment of inertia around its center of mass, a_1 is the distance from the pendulum's joint to its center of mass, b_c and b_1 are the viscous friction coefficients for the cart and pendulum respectively and m_c and m_1 are the masses of the cart and the pendulum. In the simulations, the physical parameters are defined according to the Table 1.

Table 1. The physical parameters of the simulated pendulum-cart system in base units.

Physical parameter	Meaning	Value
I_1	Pendulum's moment of inertia around its center of mass	0.0227kg m ²
a_1	Distance from the pendulum joint to its center of mass	0.18 m
b_c	Linear cart friction coefficient	10 $\frac{Ns}{m}$
b_1	Pendulum joint friction coefficient	0.15 $\frac{Ns}{m}$
m_c	Mass of the cart	0.8 kg
m_1	Mass of the pendulum	1 kg
g	Gravitational field intensity	9.81 $\frac{N}{kg}$

The state derivatives \dot{x}_1 and \dot{x}_2 are equal to the state variables x_3 and x_4 respectively. With the state derivative vector $\dot{\mathbf{x}}$ defined, the system can be numerically simulated using an ODE solver, in this case I used MATLAB's ODE45. During the simulation, the input U was defined as a band-limited random noise process, which was created by low-pass filtering a white noise signal. The result of the simulation are trajectories, or state measurements, \mathbf{X} , with sampling period $\Delta t = 0.001$ s and total duration 65 s. To simulate real measurements, I only take the first two variables, the cart position x_1 and the pendulum angle x_2 . To simulate noise, I add an additive white noise with zero mean and standard deviation $\sigma = 0.005$ m for the cart position measurements and $\sigma = 0.0025$ rad to the angle measurements.

4.2. Processing the measurements

The SINDy-PI method requires full state measurements \mathbf{X} , its derivatives $\dot{\mathbf{X}}$, and a function library $\Theta(\mathbf{X}, \dot{\mathbf{X}})$. The state variables x_3 and x_4 are equal to \dot{x}_1 and \dot{x}_2 respectively and represent the cart's linear velocity and the angle's angular velocity respectively. The state derivative variables \dot{x}_3 and \dot{x}_4 are the cart's linear acceleration and the pendulum's angular acceleration. Before creating the candidate function library, the velocities and accelerations must be estimated from the position measurements x_1 and x_2 . This can be done by spectral differentiation. Because the measurements contain noise, they must first be filtered. Since the data is intended for numerical differentiation, the cutoff frequency is set relatively low to over-filter the signal for reasons established before. The filtered signals are then numerically differentiated, generating the velocities $\dot{x}_1 = x_3$ and $\dot{x}_2 = x_4$, which are again differentiated generating \dot{x}_3 and \dot{x}_4 . The signals x_1 and x_2 are shown in Figure 2, and the accelerations \dot{x}_3 and \dot{x}_4 are shown in Figure 3.



Fig. 2. Comparison between the original noisy measurements of x_1 and x_2 and the filtered signals.

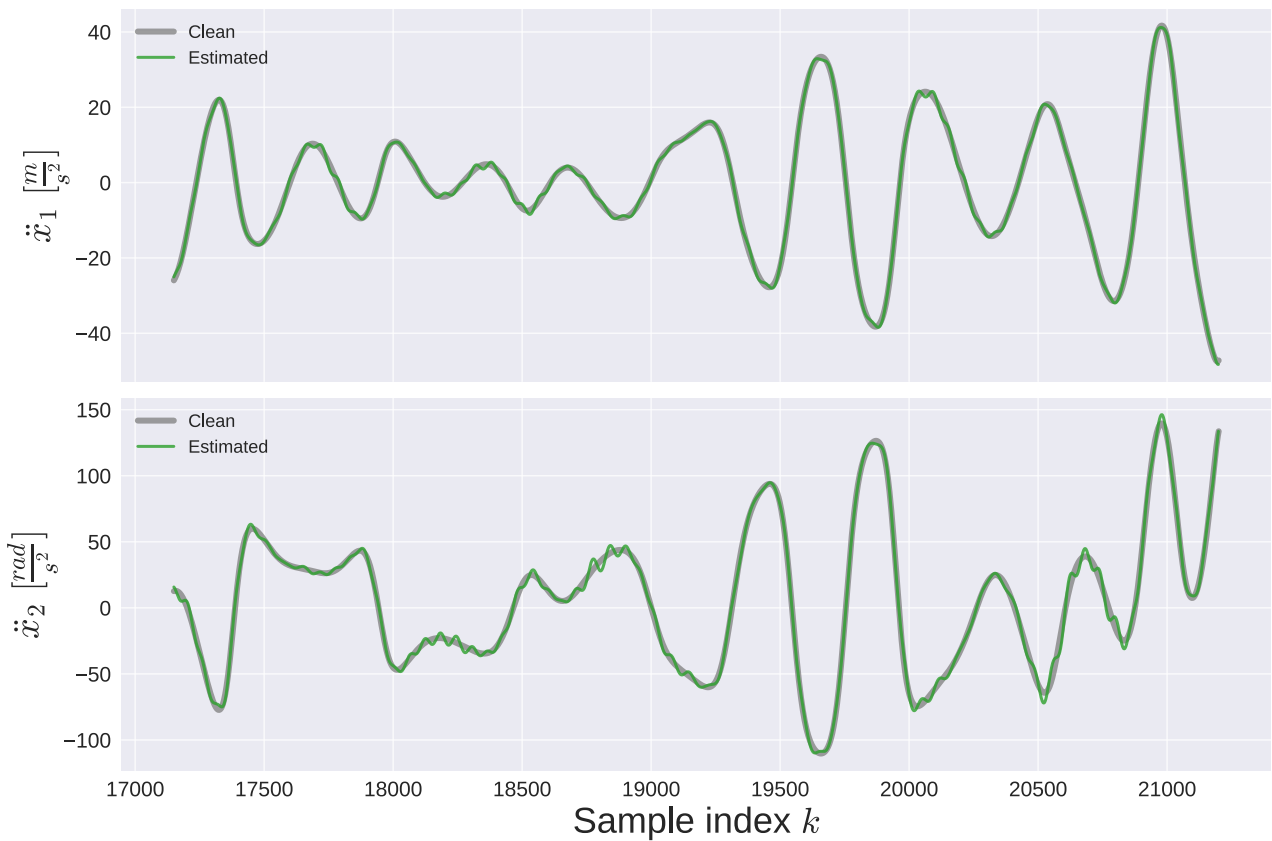


Fig. 3. Comparison between the computed accelerations \ddot{x}_3 and \ddot{x}_4 and the real accelerations.

4.2.1. Creating the function library

The function library $\Theta(\mathbf{X}, \dot{\mathbf{X}})$ is a matrix with columns representing the candidate functions θ . This step is where an expert's input is necessary, because the candidate functions must be created from the positions, velocities and accelerations so that all the terms that are in the real system dynamics are also in the function library. I create these functions by first creating a set of basis functions from which all the other candidate functions are generated. The state variable x_2 , the angle of the pendulum, implies rotation, so I added the functions $\sin x_2$ and $\cos x_2$. The positions of the cart x_1 and the angle of the pendulum x_2 don't appear by themselves in the real dynamics, so they're not included in the basis function set. The state derivative variables \dot{x}_1 and \dot{x}_2 are also omitted, because they're already represented by x_3 and x_4 respectively. The basis function set is therefore

$$\begin{aligned} \mathbf{Y} &= \{x_3, x_4, \sin(x_2), \cos(x_2), u, 1, \dot{x}_3, \dot{x}_4\} = \\ &= \{y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8\} \end{aligned} \quad (24)$$

A large number of candidate functions is then generated from \mathbf{Y} by creating all the possible 4-th order terms. Because I included the constant term $1(t)$ in the basis function set, all 4-th order terms of \mathbf{Y} will actually be all terms of the 1-st order up to the 4-th order. This generates 330 candidate functions, but most of them are dropped according to a set of manually defined rules, because they're not physically interpretable.

Creating the candidate functions manually is also possible, but it's much more time consuming than creating a large number of candidate functions and then defining a set of rules which are used to remove the bad ones. Including a candidate function that *isn't* in the real system dynamics is no big deal, because sparse regression will simply discard it when looking for a solution. But when a candidate function that is in the real system dynamics is *missing* from the function library, then the solution cannot be found. For this reason, it's safer to create more candidate functions than is necessary than to risk not including an important candidate function.

After dropping bad candidate functions, the function library Θ has 35 functions left. The reference model equations both have less than 10 unique terms in total, so the solution should pick less than 10 terms from the function library.

It's also important to look at the correlations between candidate functions, because high pair-wise correlations make the regression problem ill-posed. The correlation matrix for the function matrix Θ used for identifying the model for \dot{x}_3 is shown in Figure 4.

High pair-wise correlations themselves might give valuable insight about the actual system. If a time-derivative of some variable is perfectly correlated with another variable, that variable can be used as a model itself. In any case, the correlation matrix serves the expert as a guide when dropping candidate functions from the function library.

4.3. Creating candidate models

The SINDy-PI method described in subsection 2.4. relies on guesses that a function θ_i from Θ is active in the real dynamics. This guess is usually, for most columns, wrong. Because of this, many models must be created using different guess candidate functions. Furthermore, the hyperparameter λ_R used in the sequentially energy thresholded least-squares algorithm described in subsection 2.2.3. defines the sparsity of the solution and it also must be guessed. This means that for each candidate function guess, we must make another set of guesses of the hyperparameter value and create the respective models. The objective is to find the best model equation for each of the accelerations \dot{x}_3 and \dot{x}_4 .

4.4. Picking the best candidate models

For each of the target variables, more than 200 models is generated. Many of these candidate models have non-sparse solutions ξ and can therefore be dropped immediately. The rest of the models must be evaluated according to some accuracy and simplicity metric.

For equal comparison, I reorder the models defined by the equation (15) after creating the model back to the implicit form, by moving the guess function θ_i back into the function library Θ_i at the i -th column and adding an element into the i -th row of ξ_i with a value of -1 . This

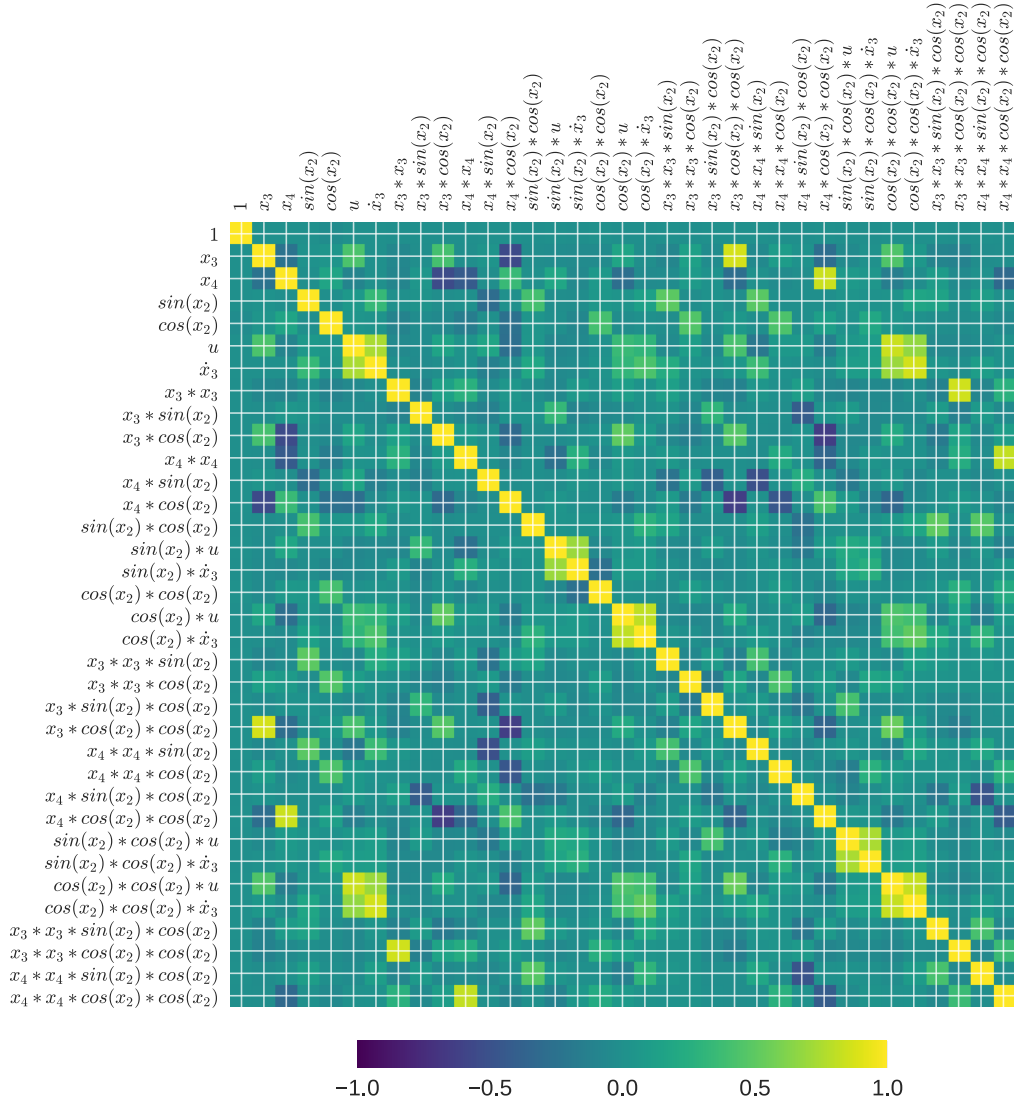


Fig. 4. Correlation matrix for the function library used to generate models for \dot{x}_3 . Shows the pair-wise correlation between candidate functions θ_i and θ_j , where i is the row index and j is the column index in the matrix.

results in an implicit model $\Theta(\mathbf{X}, \dot{\mathbf{X}})\boldsymbol{\xi} = \mathbf{0}$. Because the real model should be found whenever a correct candidate function guess θ_i is made, a model that appears consistently is likely to be correct. The solution vector $\boldsymbol{\xi}$ defining the models cannot however be compared directly, because there will always be a deviation in the parameters or the signs will be flipped. The objective is to find functions with the same active candidate functions. This can be done by picking the solutions $\boldsymbol{\xi}$ for each implicit model and creating another vector \mathbf{a} whose elements have value 1 whenever the respective element in ξ_i is non-zero and 0 everywhere else. This vector \mathbf{a} , which I call the term activation vector, is computed for every candidate model. The L^1 distance in \mathbf{a} between candidate models then simply says how many different active functions the models have, let's call this the *activation distance*. If two models have exactly the same active terms, then the activation distance is 0. After computing a distance activation matrix, which defines the activation distance between every pair of models, we can use clustering to find consistent models, which are likely to be the correct ones.

After clustering the models using their activation vectors, every model is given a cluster label. I only kept the models from clusters which contained 2 or more models. I then calculated the RMSE accuracy metric on those models. The implicit models for \dot{x}_3 and \dot{x}_4 are visualized in Figures 5. Each row represents a single model, with the respective y-axis tick describing the guess function θ_i which was used to generate the model and the model's training RMSE score. The columns represent the candidate functions. Candidate functions that didn't appear in any

of the models aren't visualized in the figures, because they'd make the plots too wide. The red and blue squares represent the signs of coefficients of ξ associated with the given candidate function, the square is red when the coefficient is positive, blue when it's negative and white when it's 0.

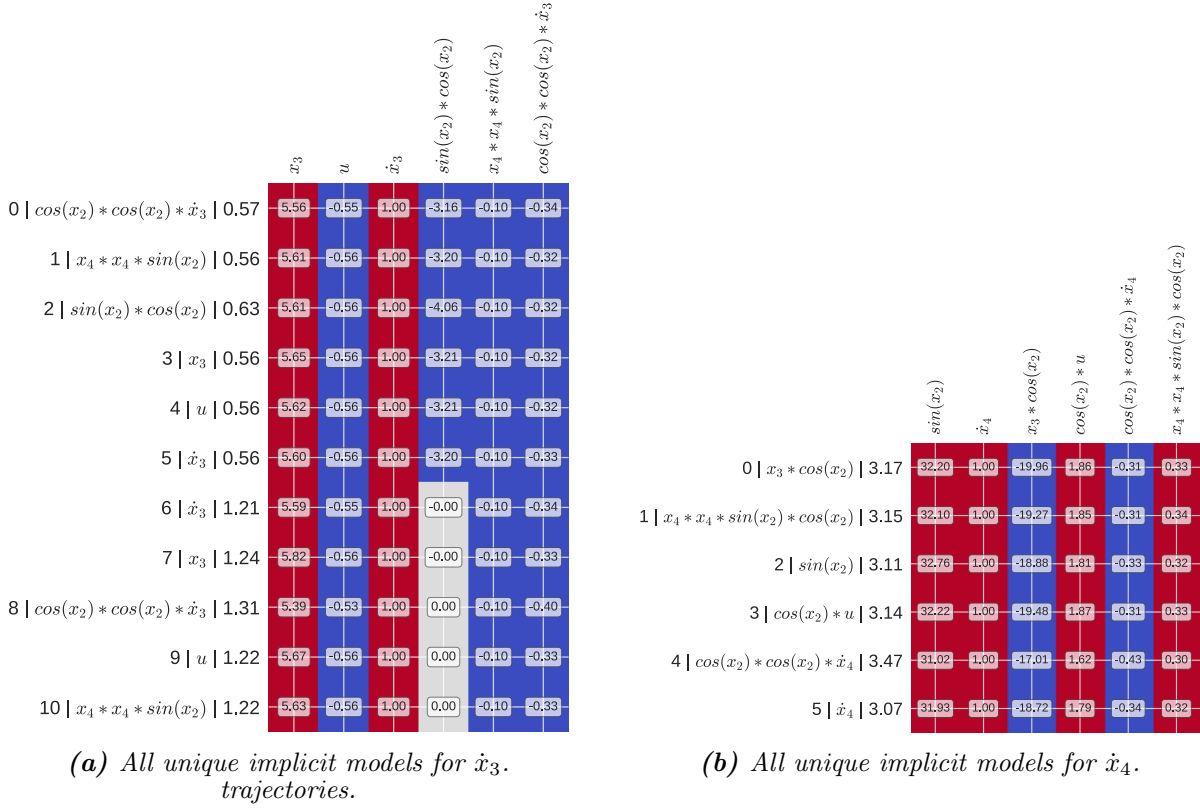


Fig. 5. The set of sparse, consistent and accurate identified implicit models. Each row corresponds to one model, with the y-axis labels specifying the model's index, guess function used to find the solution and its training RMSE metric.

The implicit models are then solved for the respective accelerations \dot{x}_3 and \dot{x}_4 , in this case using the Python's symbolic math package SymPy. The comparison of the reference model used for data generation and the best model identified from the data is in Table 2.

Table 2. Comparison of the reference and the identified models.

Cart acceleration model \dot{x}_1 [$\frac{m}{s^2}$]	
Reference	$\frac{-0.66407u + 6.64064x_3 - 0.11953x_4^2 \sin(x_2) - 0.02169x_4 \cos(x_2) - 1.91599 \sin(2.0x_2)}{0.19531 \cos(2.0x_2) - 1.0}$
SINDy	$\frac{-0.55672u + 5.61287x_3 - 0.10007x_4^2 \sin(x_2) - 1.61162 \sin(2.0x_2)}{0.32312 \cos^2(x_2) - 1.0}$
Pendulum angular acceleration model \dot{x}_2 [$\frac{rad}{s^2}$]	
Reference	$\frac{1.81554u \cos(x_2) - 18.15533x_3 \cos(x_2) + 0.1634x_4^2 \sin(2.0x_2) + 0.18156x_4 + 32.05886 \sin(x_2)}{0.3268 \cos^2(x_2) - 1.0}$
SINDy	$\frac{2.16696u \cos(x_2) - 22.63092x_3 \cos(x_2) + 0.19338x_4^2 \sin(2.0x_2) + 38.39076 \sin(x_2)}{0.19753 \cos(2.0x_2) - 1.0}$

4.5. Evaluating the acceleration models

Now that we have model equations for both accelerations \dot{x}_3 and \dot{x}_4 , we can take a state measurement matrix \mathbf{X} and external input measurement matrix \mathbf{U} , pass their rows \mathbf{x} and u into each equation and get the predicted accelerations. The predicted accelerations can then be compared to the real accelerations as computed by the reference model. The comparisons of derivative predictions are in Figures 6.

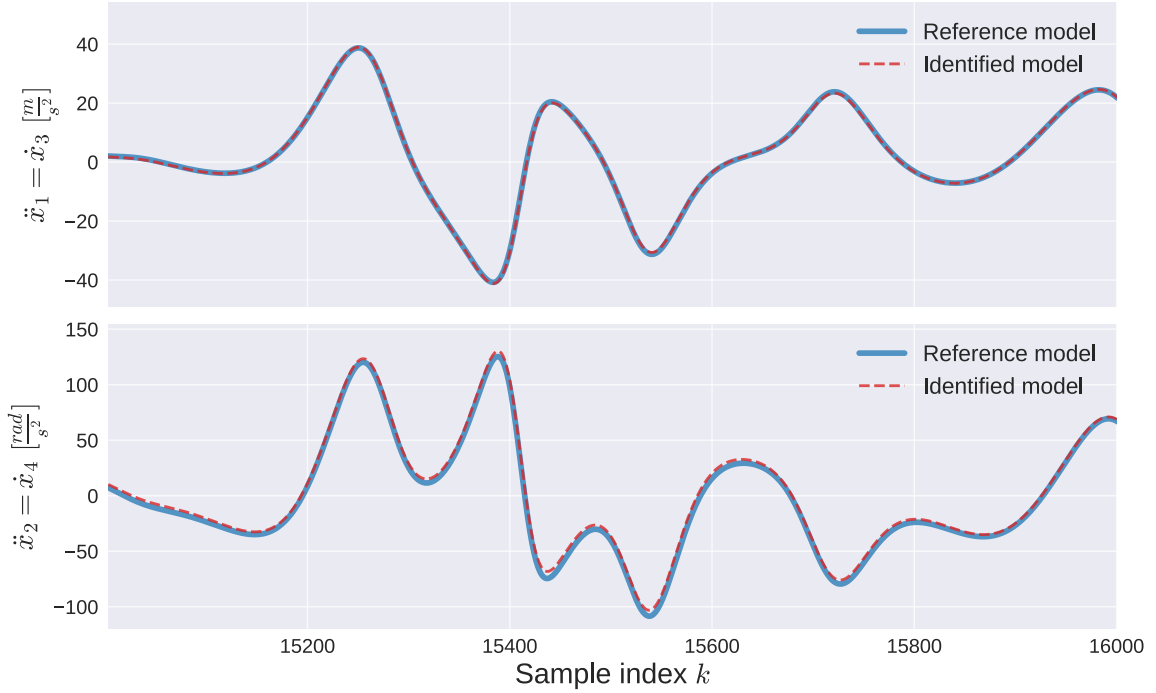


Fig. 6. Comparison of real accelerations to accelerations computed from the state using the identified model.

The model predictions are very close to the real accelerations, although there are some small deviations. Consolidating the acceleration models, $\dot{\mathbf{x}}$, into one state derivative vector equation creates a full state model. This way, the identified model can be simulated the same way as the analytically derived reference model. I simulated both from the same initial condition $\mathbf{x}[k=0]$ and using the same input sequence \mathbf{u} . The trajectories for each state variable are visualized in Figure 7

The trajectories start out very close to each other, but small deviations eventually add up and the trajectories separate into relative chaos. In practice, the identified model would be used for its predictive capability. The accumulation of errors (differences between the model prediction and the real system) would be prevented with an estimator, for example a Kalman filter, that uses sensor feedback to determine the real state by weighing both the model prediction and sensor measurements.

5. Conclusion

An accurate nonlinear, rational model of the pendulum-cart system was identified from simulated measurement data with additive white noise. The data was first filtered by transforming the measurements into the frequency domain and setting the frequency components above the cutoff frequency to zero. This data was then numerically differentiated leveraging the properties of Fourier transforms to get the first and second derivatives needed for identification. Using the SINDy-PI method, an implicit ordinary differential equation was identified for each of the two accelerations in the system. From these two implicit models, the explicit equations were created by symbolically solving for the respective acceleration. The identified acceleration equations were accurate at predicting the acceleration given the state vector and external input value. By combining both acceleration equations, a full state model of the pendulum-cart system was created, which was then numerically simulated and compared to the real, analytically derived model used for generating the training data. The trajectories generated by both the real and identified model overlapped for the first few seconds, but the small errors between the models accumulated and the trajectories eventually decoupled. However, the models are nevertheless qualitatively nearly identical, and in practice, the imperfectness of the identified model would be largely compensated for via sensor feedback and state estimation.

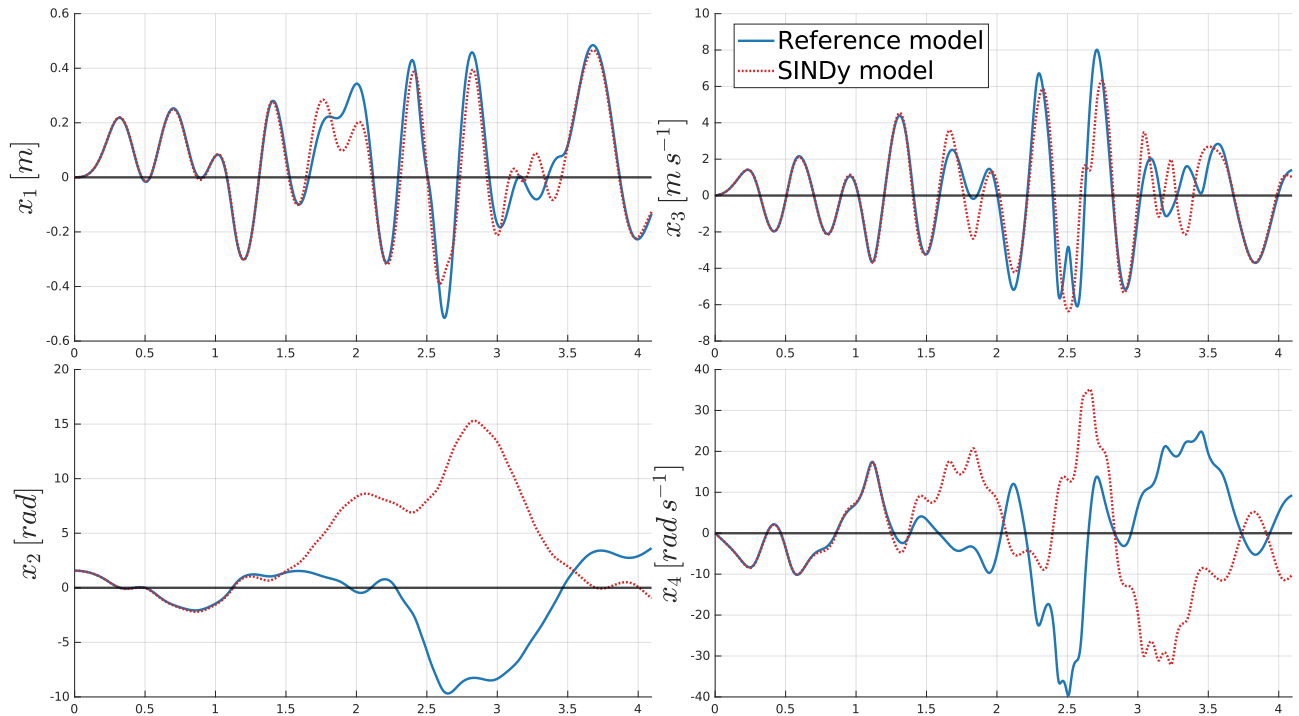


Fig. 7. Parallel simulation of the reference model and the identified model. Full animation at: <https://git.io/JBPXD>



Acknowledgement

This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS19/158/OHK2/3T/12.

Nomenclature

x_1	Cart position (m)
x_2	Pendulum angle (rad)
x_3, \dot{x}_1	Cart velocity (m s^{-1})
x_4, \dot{x}_2	Pendulum angular velocity (rad s^{-1})
\ddot{x}_1, \ddot{x}_3	Cart acceleration (m s^{-2})
\ddot{x}_2, \ddot{x}_4	Pendulum angular acceleration (rad s^{-2})
\mathbf{x}	State vector ()
$\dot{\mathbf{x}}$	State derivative vector ()
Θ	Function library ()
ξ	Vector of model parameters ()
\mathbf{a}	Activation vector ()
\mathbf{X}	Matrix of state measurements ()
$\dot{\mathbf{X}}$	Matrix of state derivative measurements ()
\mathbf{U}	Matrix of input measurements ()

References

- [1] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”. In: *Proceedings of the National Academy of Sciences* 113.15 (Apr. 12, 2016). Publisher: National Academy of Sciences Section: Physical Sciences, pp. 3932–3937. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1517384113. Available at: <https://www.pnas.org/content/113/15/3932> (visited on 03/31/2021).
- [2] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. “Sparse Identification of Nonlinear Dynamics with Control (SINDYc)”. In: *arXiv:1605.06682 [math]* (May 21, 2016). arXiv: 1605.06682. Available at: <http://arxiv.org/abs/1605.06682> (visited on 03/31/2021).
- [3] Niall M. Mangan et al. “Inferring biological networks by sparse identification of nonlinear dynamics”. In: *arXiv:1605.08368 [math]* (May 26, 2016). arXiv: 1605.08368. Available at: <http://arxiv.org/abs/1605.08368> (visited on 04/15/2021).
- [4] Aaron Meurer et al. “SymPy: symbolic computing in Python”. In: *PeerJ Computer Science* 3 (Jan. 2017), e103. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.103. Available at: <https://doi.org/10.7717/peerj-cs.103>.
- [5] Kadierdan Kaheman, J. Nathan Kutz, and Steven L. Brunton. “SINDy-PI: A Robust Algorithm for Parallel Implicit Sparse Identification of Nonlinear Dynamics”. In: *arXiv:2004.02322 [physics, stat]* (Sept. 29, 2020). arXiv: 2004.02322. Available at: <http://arxiv.org/abs/2004.02322> (visited on 03/31/2021).
- [6] Steven L. Brunton and J. Nathan Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. 1st ed. Cambridge University Press, Jan. 31, 2019. ISBN: 978-1-108-38069-0 978-1-108-42209-3. DOI: 10.1017/9781108380690. Available at: <https://www.cambridge.org/core/product/identifier/9781108380690/type/book> (visited on 03/31/2021).
- [7] Tobias Glück, Andreas Eder, and Andreas Kugi. “Swing-up control of a triple pendulum on a cart with experimental validation”. In: *Automatica* 49.3 (Mar. 2013), pp. 801–808. ISSN: 00051098. DOI: 10.1016/j.automatica.2012.12.006. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S000510981200605X> (visited on 04/11/2021).

Návrh nelineárního modelu 3D portálového jeřábu

Michal Pála

FS ČVUT

Přístrojová a řídicí technika

Praha, Česká republika

michal.pala@fs.cvut.cz

Abstrakt—Používání jeřábů v oblastech logistiky je dnes častým řešením přemísťování těžkých nákladů. Značnou nevýhodou jejich používání je přirozený kyv břemene, ke kterému při pohybu dochází. To mimo jiné prodlužuje časovou náročnost procesu či zvyšuje rizika nebezpečí. Tomu lze předejít navržením vhodného řídicího systému. Díky přibližné představě o chování, kterou nám matematické modely poskytují jsme schopni tyto řídicí systémy navrhnout bez použití reálné soustavy a až poté je aplikovat. Tím pak můžeme zamezit případným komplikacím v praxi. Článek je zaměřen na návrh nelineárního modelu portálového jeřábu pomocí Lagrangeových rovnic II. druhu. V závěru je chování modelu porovnáváno se skutečnou soustavou zn. INTECO.

Index Terms—portálový jeřáb, modelování, nelineární model, lagrange

I. ÚVOD

Jeřáby jsou dnes hojně používány k přemísťování velkých a těžkých nákladů v oblastech průmyslu a logistiky. Toto řešení s sebou ale nese rizika spojené s přirozeným kyvem těchto břemen při jejich přemísťování. Eliminací těchto kyvů můžeme dosáhnout větší bezpečnosti a efektivitě při práci s jeřáby. K tomu je potřeba mít na jeřábu vhodný řídicí systém, který se v dnešní době navrhuje s pomocí matematických modelů.

Díky přibližné představě o chování, kterou nám matematické modely poskytují jsme schopni tyto řídicí systémy navrhnout bez použití reálné soustavy a až poté je aplikovat. Tím se můžeme vyhnout zbytečným experimentům a předejít problémům v praxi. Matematické modely se tvoří na základě znalostí z fyziky a matematiky. Cílem je získat dynamiku nezávislých proměnných (druhé derivace) z hlediska působení vnějších sil. Ty jsou v tomto článku získány metodou Lagrangeových rovnic II. druhu. Z druhých derivací lze následně získat první derivace a poté samotné nezávislé proměnné. Spolu s vstupními parametry a počátečními podmínkami lze model odsimulovat a porovnávat s reálnou soustavou. Vstupní parametry je potřeba nějak získat (změřit, zvážit, vypočítat, atd.). Počáteční podmínky si můžeme určit.

S vyjádřenými druhými derivacemi nezávislých proměnných lze také získat lineární model pomocí stavového popisu (state space), díky němuž lze navrhnout řízení, např. stavové regulátory nebo tvarovače signálu (tzv. input shaping) sloužící k rozdělení vstupního signálu na více částí a tím eliminovat část kyvu – již není obsahem této práce.

Z hlediska řízení se soustava portálového jeřábu označuje jako silně nelineární systém typu MIMO (Multiple Input and

Multiple Output). V této práci je mezi vstupy zavedeno napětí ve směru u_x a u_y . Výstupem ze soustavy je poloha ve směru x a y včetně úhlů kyvu břemene α a β . Navíjení lana je zanebáno (vstup u_z nevyužit) a jeho délka je fixní.

Článek je zaměřen na návrh nelineárního modelu portálového jeřábu, popis zjištění jeho parametrů a následné porovnání chování s reálnou soustavou portálového jeřábu (INTECO).

Této úloze se již věnovalo více lidí předemnou. V práci [2] z Malajsie vytvořili obdobný model 3D portálového jeřábu, jako je model v tomto článku. Ve schématu mají rozdílně zakótované nezávislé úhly. Uvažují kinetickou energii nákladu včetně její rotace. Při mém pokusu o odsimulování nelineárního modelu s použitím stejných rovnic z [2] se model nechoval správně. Například při akčním zásahu F_x bychom čekali, že úhel θ se bude kývat (fungovalo správně) a že úhel ϕ dosáhne hodnoty 90° , což z počátečních podmínek 0° nedosáhl.

V práci [1] z talinské univerzity mají stejný laboratorní model jeřábu od výrobce INTECO. Využil jsem tedy citovaný článek k získání některých parametrů (viz. kapitola V). Úhly α a β jsou v této práci stejně zakótovány jako zde, jen jsou prohozeny. Potenciální energii mají definovanou jako $P = mgl$, což je zvláštní, protože se nemění v závislosti na úhlech α a β . Model tvoří také skrze Lagrangeovy rovnice II. druhu s pěti nezávislými proměnnými (uvažují i proměnnou délku lana L) a třetím vstupem (motor ovládající délku lana L). Na jejich práci navazoval návrh řízení pomocí Fuzzy regulátoru.

Kapitola I je samotný úvod. Kapitola II se zabývá popisem reálné soustavy jeřábu od polského výrobce INTECO. III. kapitola je zaměřena na popis a vysvětlení funkčnosti portálového jeřábu s pomocí zjednodušeného schématu, ze kterého se později určují první rovnice k vytvoření nelineárního modelu (kapitola IV). Kapitola V pojednává o parametrech modelu, které jsou nutné k odsimulování modelu a zjištění jeho chování v čase. V kapitole VI je chování modelu porovnáváno se skutečnou laboratorní soustavou. Z závěrečné kapitole VII je stručné shrnutí výsledků této práce.

II. POPIS PORTÁLOVÉHO JEŘÁBU INTECO

Skutečná soustava jeřábu, která je pro tuto úlohu k dispozici je 3D portálový jeřáb od polské firmy INTECO.

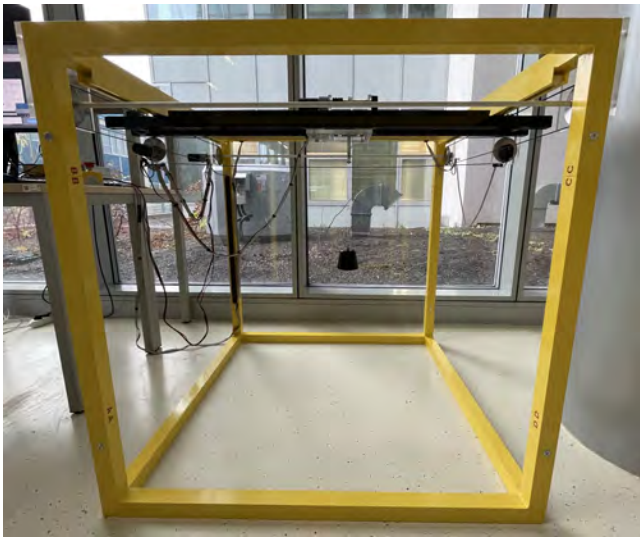
Jedná se o silně nelineární MIMO (Multiple Input and Multiple Output) elektromechanický systém. Jeho rozměry jsou $1 \times 1 \times 1 \text{ m}$.

Pohyby jsou zajištěny DC motory ve směru x , y a z . Ty jsou ovládány pomocí PWM (Pulse Width Modulation), čili v rozmezí hodnot od -1 až 1 .

Jeřáb je osazen pěti enkodéry, díky kterým lze zjistit polohy ve směru x , y , délku lana L , úhel ve směru osy x (značeno jako α) a úhel ve směru osy y (značeno jako β).

V rámci této práce není pohyb do směru z používán. Délka lana je zafixována.

Jeřáb je ovládán skrze počítač s prostředím Matlab Simulink.



Obrázek 1. Portálový jeřáb INTECO

III. SCHÉMA MODELU PORTÁLOVÉHO JEŘÁBU

Na obrázku Obr. 2 lze vidět zjednodušené schéma portálového jeřábu. Jeřáb se skládá s kolejnice (pohybující se ve směru x), jeřábové kočky (pohybující se ve směru x a y) a zavěšeného nákladu (pohybující se do směru x , y i z).

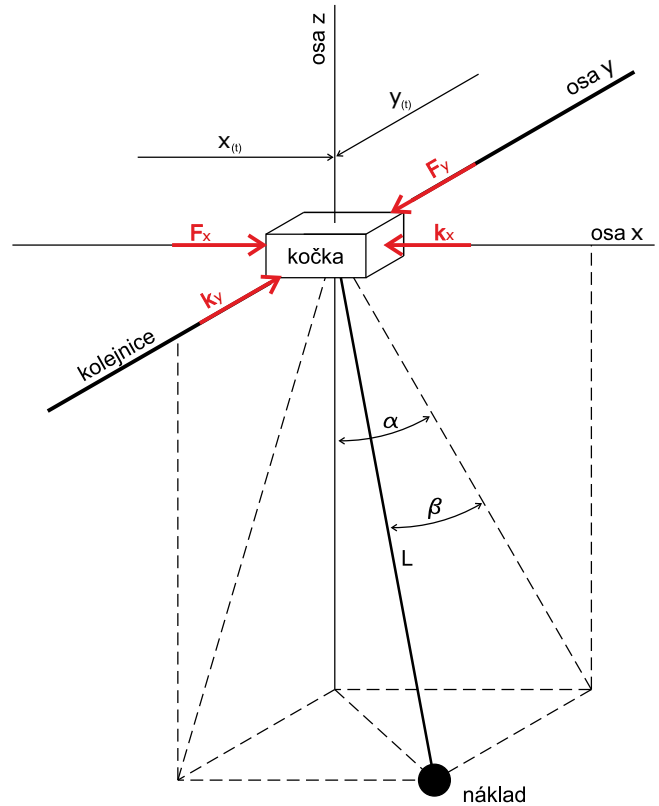
Vstupem do soustavy je napětí u_x (ve směru x) a napětí u_y (ve směru y). Napětí vstupující do soustavy je v pozdější fázi přepočítáváno na sílu F_x (ve směru x) a sílu F_y (ve směru y). Pro přepočet musí být experimentálně zjištěny koeficienty k_{ux} a k_{uy} .

Výstupem ze soustavy je poloha ve směru x , poloha ve směru y , úhel α (úhel mezi průmětem lana do roviny xz a osou z), úhel β (mezi lanem a průmětem lana do roviny xz).

Úhel β je ve skutečnosti měřen jako úhel od osy y . Pro zjednodušení při tvorbě modelu byl zaveden jako na obr. 2 (zaveden jako úhel mezi lanem a průmětem lana do roviny xz). Chyby, ke kterým s touto úpravou dochází jsou v blízkosti dolní úvratě zanedbatelné.

A. Nezávislé souřadnice

Nezávislé souřadnice jsou sepsány v tabulce tab. I. Mezi nezávislé souřadnice patří poloha kolejnice či kočky ve směru



Obrázek 2. Schéma portálového jeřábu

x , poloha kočky ve směru y a vychýlení úhlů α a β . Tyto souřadnice budou výstupem z nelineárního modelu.

Tabulka I
NEZÁVISLÉ SOUŘADNICE

Označení	Popis
x	poloha kolejnice a kočky ve směru x
y	poloha kočky ve směru y
α	úhel mezi průmětem lana do roviny xz a osou z
β	úhel mezi lanem a průmětem lana do roviny xz

IV. MODELOVÁNÍ

A. Poloha kolejnice, jeřábové kočky a závaží

Poloha kolejnice p_r (1), poloha kočky p_c (2) a poloha závaží p_p (3) jsou dány:

$$p_r = \begin{pmatrix} x(t) \\ 0 \\ 0 \end{pmatrix} \quad (1)$$

$$p_c = \begin{pmatrix} x(t) \\ y(t) \\ 0 \end{pmatrix} \quad (2)$$

$$p_p = \begin{pmatrix} x(t) + L \cdot \sin(\alpha(t)) \cdot \cos(\beta(t)) \\ y(t) + L \cdot \sin(\beta(t)) \\ -L \cdot \cos(\alpha(t)) \cdot \cos(\beta(t)) \end{pmatrix} \quad (3)$$

B. Kinetická energie

Energie kolejnice K_r (4) je dána vztahem

$$K_r = \frac{1}{2} \cdot m_r \cdot \dot{p}_r^2 \quad (4)$$

Energie kočky K_c (5) je dána vztahem

$$K_c = \frac{1}{2} \cdot m_c \cdot \dot{p}_c^2 \quad (5)$$

Energie nákladu K_p (6) je dána vztahem

$$K_p = \frac{1}{2} \cdot m_p \cdot \dot{p}_p^2 \quad (6)$$

Sečtením rovnic (4), (5) a (6) získáváme celkovou kinetickou energii K (7)

$$K = K_r + K_c + K_p \quad (7)$$

C. Potenciální energie

Potenciální energie P (8) pro tento případ

$$P = -m_p \cdot g \cdot L \cdot \cos(\alpha(t)) \cdot \cos(\beta(t)) \quad (8)$$

D. Euler-Lagrange

Lagrangián (9) je definován jako rozdíl kinetické (7) a potenciální (8) energie

$$\mathcal{L} = K - P \quad (9)$$

Lagrangerova rovnice (10) [2] je dána vztahem

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} = F_i, i = 1, 2, 3, 4 \quad (10)$$

kde q (11) je vektor nezávislých proměnných

$$q = \begin{pmatrix} x(t) \\ y(t) \\ \alpha(t) \\ \beta(t) \end{pmatrix} \quad (11)$$

a vnější síly F (12) jsou (včetně tření)

$$F = \begin{pmatrix} F_x(t) - k_x \cdot \dot{x}(t) \\ F_y(t) - k_y \cdot \dot{y}(t) \\ -k_\alpha \cdot \dot{\alpha}(t) \\ -k_\beta \cdot \dot{\beta}(t) \end{pmatrix} \quad (12)$$

Po vypočtení Lagrangerovy rovnice (10) dostáváme druhou derivaci všech nezávislých proměnných (zrychlení), ze kterých je možné vytvořit nelineární model v prostředí Matlab Simulink. Výsledné vyjádření proměnné nejsou přidány do tohoto textu, protože jsou příliš dlouhé.

V. PARAMETRY

Aby bylo možné model odsimulovat, je do něj potřeba znát všechny potřebné parametry. Parametry jsou k vidění v tab. II

Hmotnost nákladu (závaží) m_p jsem zvažil. Hmotnost kočky m_c a hmotnost kolejnice m_r jsou převzaty z [1], kde je použit identický laboratorní model jeřábu a nebylo kvůli tomu nutné jeřáb rozebírat na části.

Tření v čepu k_α a k_β bylo zanedbáno, protože v pozdějších fázích zjišťování parametrů tlumily kyv nákladu více, než bylo třeba. Toto tření je dohnáno skrze parametry k_x a k_y , skrze které dochází k tření při pohybu vozíku a tím i k tlumení kyvadla.

Přepočty k_{ux} a k_{uy} sloužící na převod napětí (F_x , F_y) na sílu (u_x , u_y) byly zjištěno experimentálně tak, aby se shodovaly směrnice a nájezd polohy ve směrech x a y (k vidění na obr. 3).

Hmotnost lana je zanedbána.

Všechny tyto parametry jsou použity jako vstup do nelineárního modelu.

Tabulka II
PARAMETRY MODELU

Ozn.	Popis
m_c	hmotnost kočky $m_c = 1,16$ kg, převzato z [1]
m_r	hmotnost kolejnice $m_r = 2,2$ kg, převzato z [1]
m_p	hmotnost nákladu $m_p = 0,58$ kg, zváženo
L	Délka lana $L = 0,413$ m, změřeno
g	gravitační zrychlení $g = 9,81$ m ² /s
k_α	tření v čepu $k_\alpha = 0$ [-], zanedbáno
k_β	tření v čepu $k_\beta = 0$ [-], zanedbáno
k_x	tření $k_x = 31$ [-], odhadnuto
k_y	tření $k_y = 26,5$ [-], odhadnuto
k_{ux}	přepočet napětí na sílu $k_{ux} = 13,15$ [-], odhadnuto
k_{uy}	přepočet napětí na sílu $k_{uy} = 12,48$ [-], odhadnuto

VI. POROVNÁNÍ NELINEÁRNÍHO MODELU S JEŘÁBEM INTECO

Pro porovnání bylo potřeba odhadnout parametry k_x a k_y spolu s přepočtem napětí na sílu u_x , u_y . Toho jsem docílil tak, aby seděly směrnice průběhů polohy x a y , viz. obr. 3 a obr. 4. Tření bylo zafixováno a hodnoty napětí byly zjištěny pro zásahy u_x , u_y od 0,2 do 1 (po 0,2). Výsledné hodnoty napětí u_x a u_y byly zjištěny zprůměrováním experimentálně zjištěných hodnot, viz Tab. II.

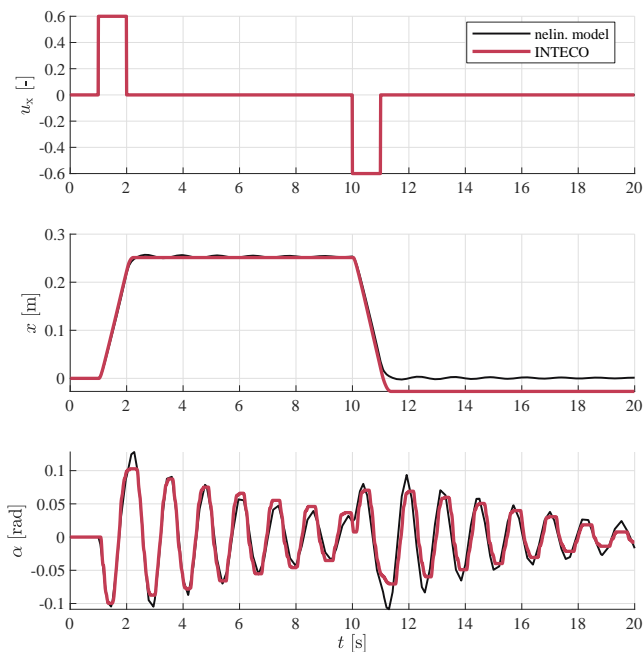
Pro měření odezvy byl použit kladný skok v čase $t = 1$ s a záporný skok v čase $t = 10$ s o velikosti u_x , $u_y = 0,6$. Měření byla prováděna pro každý směr zvlášť. Jejich odezvy lze vidět na obr. 3 (pro směr x) a na obr. 4 (pro směr y).

Na obr. 3 lze vidět odezvu nelineárního modelu a jeřábu na vstupní napětí $u_x = 0,6$. Průběhy nelineárního modelu a jeřábu jsou velmi podobné. Cestou zpět kolejnice ujede více, než dopředu. Mohlo by to být dáno svazkem kabelů, který je ke kolejnici připevněn. Do kladného směru ho kolejnice tahá, do záporného vrací zpět.

Na obr. 4 lze vidět odezvu nelineárního modelu a jeřábu na vstupní napětí u_y . Zde se průběhy po kladném zásahu

příliš neliší, avšak při pohybu zpět se nelineární model odchýlí od reálné soustavy. Myslím si, že je to dáno tím, že v momentu zásahu záporným napětím bylo kyvadlo na nelineárním modelu více utlumené oproti jeřábu a proto zde došlo k větší změně kyvu a tím k lehkému narušení průběhu. Jelikož se shoduje první zásah obou soustav, považuji výsledek za přívětivý.

Jelikož se první kyvy skvěle shodují u nelineárního modelu i u reálné soustavy, bude tento model vhodný na budoucí návrhy řízení.



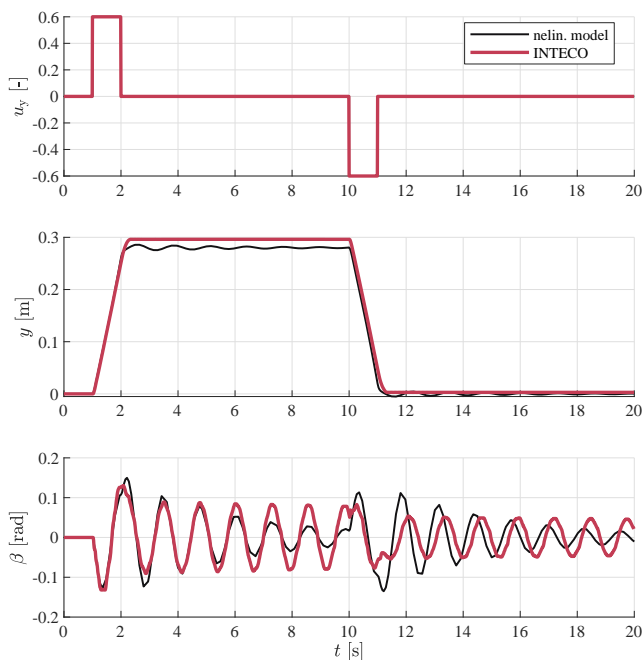
Obrázek 3. Porovnání odezvy nelineárního modelu a jeřábu INTECO ve směru při zásahu $u_x = 0,6$

VII. ZÁVĚR

Udělal jsem nelineární model pomocí Lagrangeových rovnic druhého druhu. Pro model byly zjištěny všechny potřebné parametry. Výsledkem práce bylo porovnání chování navrženého nelineárního modelu s jeřábem INTECO. Pro směr x se modely shodují skvěle. Pro směr y se nepodařilo parametry navrhnout tak dobře jako pro směr x . Kvůli tomu je kyvadlo více tlumeno oproti skutečné soustavě a ve chvíli zásahu v záporném směru dochází k menší nepřesnosti na výstupu z modelu. Pro budoucí navrhování řízení nás zajímá hlavně průběh prvních několika vteřin, proto bude nelineární model dostačující pro další aplikace s ním spojené. Všechny cíle práce byly splněny.

REFERENCE

- [1] AKSJONOV, Andrei, Valery VODOVOZOV a Eduard PETLENKOV. Three-Dimensional Crane Modelling and Control Using Euler-Lagrange State-Space Approach and Anti-Swing Fuzzy Logic. *Electrical, Control and Communication Engineering* [online]. 2015, 9(1), 5-13 [cit. 2020-11-07]. ISSN 2255-9159. Dostupné z: doi:10.1515/ecce-2015-0006



Obrázek 4. Porovnání odezvy nelineárního modelu a jeřábu INTECO ve směru při zásahu $u_y = 0,6$

- [2] ISMAIL, R.M.T. Raja, M.A. AHMAD, M.S. RAMLI a F.R.M. RASHIDI. *Nonlinear Dynamic Modelling and Analysis of a 3-D Overhead Gantry Crane System with System Parameters Variation* [online]. Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang, Pekan, 26600, Pahang, Malaysia, , 8 [cit. 2021-01-04]. ISSN 1473-804x. Dostupné z: <https://core.ac.uk/download/pdf/159178801.pdf>

Poděkování

Tato práce byla podpořena grantem Studentské grantové soutěže ČVUT č. SGS SGS20/159/OHK2/3T/12.

Matematický model věžového jeřábu

Tomáš Pelcl

Fakulta strojní, České vysoké učení technické v Praze
Přístrojová a řídicí technika
Praha, 2020

Abstrakt—Matematické modelování je mocný nástroj s širokým spektrem využitelnosti. Ačkoliv nalezneme využití v celé škále profesí, pro automatické řízení je zcela neodmyslitelnou součástí pro popisování systémů, predikce jejich chování a navrhování regulace. Tento článek popisuje postup návrhu matematického modelu pro věžový jeřáb, což je poměrně složitý a silně nelineární MIMO systém s pěti stupni volnosti. Důraz je kladen na vytvoření modelu způsobem, který následně umožní jeho efektivní využití a změny v postupu či parametrech budou aplikovatelné ihned bez nutnosti jakýchkoli dalších úprav. Model byl navržen s motivací jeho budoucího použití pro návrh vhodné řídicí metody, proto je popsán také postup pro získání linearizovaného modelu, který pro většinu regulačních metod vyžadován.

Klíčová slova- Věžový jeřáb, matematický model, nelineární model, stavový popis, MIMO systém, INTECO.

I. MOTIVACE

Těžko bychom dnes hledali vědecký obor nebo výzkumnou činnost, ve kterých matematické modelování nenajde využití. Ať už se jedná o ekonomii, přírodní vědy nebo inženýrské disciplíny, matematický model dokáže být velmi silným nástrojem pro popis chování určitého systému, k simulování jeho vývoje nebo reakcí na různé podněty. V technických oborech se k sestavení takového modelu nejčastěji využívá fyzikální popis soustavy pomocí diferenciálních rovnic.

Hlavním předmětem této práce je představení postupu pro tvorbu matematického modelu pro složitý MIMO (z angl. *Multiple Output Multiple Input*) systém a předvedení na konkrétní soustavě - stavebním jeřábu.

II. PŘEDSTAVENÍ SOUSTAVY

Práce byla realizována na laboratorním modelu věžového jeřábu od firmy INTECO. Jedná se o silně nelineární dynamický systém s pěti stupni volnosti, jehož zjednodušené schéma s fyzikálním popisem znázorňuje obrázek 2. V celé této práci je délka lana uvažována konstantní, i tak se jedná o silně nelineární systém se čtyřmi stupni volnosti.

III. MATEMATICKÝ POPIS

K získání vlastních pohybových rovnic systému existuje celá řada metod. Její výběr by měl být přizpůsoben charakteru úlohy, kterou má řešit, jelikož na každou soustavu lze aplikovat metody více či méně vhodné. Při pohledu na stavební jeřáb je patrné, že se jedná o kombinaci rovinného a rotačního pohybu vozíku a prostorového kyvadla v podobě lana s břemenem. Z toho důvodu se jeví jako vhodná metoda použití Lagrangeova přístupu, kde je využito potenciální a kinetické energie T a V dílčích částí soustavy.



Obrázek 1: Laboratorní model věžového jeřábu od firmy INTECO [4]

$$T = \frac{1}{2}m_m v_m^2 + \frac{1}{2}m_T v_T^2 + \frac{1}{2}J\omega_{\text{magn}}^2 \quad (1)$$

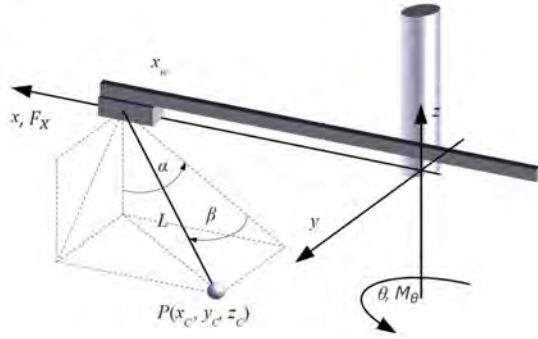
$$V = -m_m g L \cos(\beta) \cos(\alpha) \quad (2)$$

Z rovnice (1) je zřejmé, že je potřeba vyjádřit rychlost břemene v_m a rychlost vozíku v_T . Ty lze získat z příslušných zderivovaných polohových vektorů, které udávají polohu břemene a vozíku. Tyto vektory byly vyjádřeny za použití transformačních matic pohybu, jak je naznačeno rovnicemi (3) a (4). Matice R_x je matice rotačního pohybu kolem osy x , matice T_x značí rovinný pohyb ve směru x atd.

$$r_{1m} = R_z(\theta)T_x(x)R_y\left(\frac{\pi}{2} - \alpha\right)R_z(\beta)T_x(L) \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad (3)$$

$$r_{1T} = R_z(\theta)T_x(x) \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad (4)$$

Dalším krokem je výpočet tzv. Lagrangianu L_L , který představuje rozdíl kinetické a potenciální energie, tedy $L_L = T - V$.



Obrázek 2: Model soustavy se zakóvanými souřadnicemi a působícími silami

$$\frac{d}{dx} \left(\frac{\partial L_L}{\partial \dot{q}_j} \right) - \left(\frac{\partial L_L}{\partial q_j} \right) = F_j, j = 1, 2, 3, 4 \quad (5)$$

$$F_j = \begin{pmatrix} F_x - k_x \dot{x} \\ -k_\alpha \dot{\alpha} \\ -k_\beta \dot{\beta} \\ F_\theta - k_\theta \dot{\theta} \end{pmatrix} \quad (6)$$

Dalším krokem je již dosažení do Lagrangeových rovnic druhého druhu, které jsou popsány v rovnici (5). Je zde zaveden vektor proměnných $q_j = (x \ \alpha \ \beta \ \theta)$. Rovnicí (5) jsou popsány zobecněné síly, kde záporné členy reprezentují ztráty v příslušném směru. Dopočtem vyjdou čtyři vlastní pohybové rovnice soustavy. Protože vstupní veličiny jsou zatím vyjádřeny pouze jako síly F_j , musí být přepočteny na napětí U . Toho je docíleno použitím modelu DC motoru v kapitole IV.

Celý výpočet pohybových rovnic, tak jak je popsáno v této kapitole, byl od začátku proveden v programu Matlab pomocí symbolických proměnných (Symbolic Math Toolbox). Zvolený postup a matematický popis je přehledný a snadno pochopitelný, nicméně jeho výsledkem jsou velmi složité a dlouhé diferenciální rovnice. Tento postup byl upřednostněn kvůli možnosti odvození a hlubšího pochopení matematického popisu soustavy, který ve většině prací s touto problematikou chybí.

IV. MODEL DC MOTORŮ

Jeřáb INTECO je poháněn třemi DC motory. Jedná se o jediné akční členy soustavy a pohánějí rotaci výložníku ve směru θ , pohyb kočky ve směru x a navíjení lana délky L . Délka lana je v této práci uvažována jako konstanta, tento akční člen je tedy zanedbán.

$$F_x = \frac{\eta_x k_{gx} k_{m,x}}{R_{ax} r_x} G_{ax} U_x - \frac{\eta_x k_{gx}^2 k_{m,x}^2}{R_{ax} r_x^2} \dot{X}_w \quad (7)$$

$$F_\theta = \frac{\eta_\theta k_{g\theta} k_{m,\theta}}{R_{a\theta}} G_{a\theta} U_\theta - \frac{\eta_\theta k_{g\theta}^2 k_{m,\theta}^2}{R_{a\theta}} \dot{\theta} \quad (8)$$

Dynamické chování motorů pro směr x a θ je popsáno rovnicemi (7) a (8), v tomto pořadí. Odvození těchto rovnic

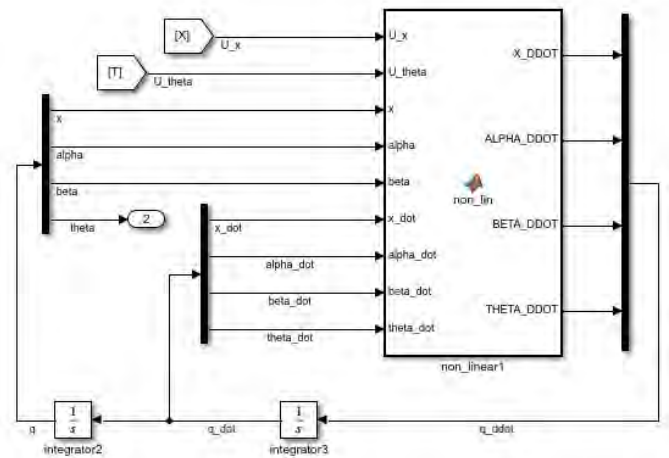
Popis	Parametr	Hodnota
hmotnost břemene	m_m	0,33 [kg]
hmotnost kočky	m_T	0,7 [kg]
moment hybnosti	J	2,4 [kgm ²]
délka lana	L	0,45 [m]
viskózní tření	k_x	14 [Ns/m]
viskózní tření	k_θ	14 [Nms/rad]
viskózní tření ve směru β	k_α	0,00005 [Nm/s]
viskózní tření ve směru α	k_β	0,015 [Nm/s]
účinnost motoru a převodu	η_x	0,36
účinnost motoru a převodu	η_θ	0,24
poloměr řemenice	r_x	0,0375 [m]
konst. momentu síly motoru	$k_{m,x}$	0,055 [Nm/A]
konst. momentu síly motoru	$k_{m,\theta}$	0,024 [Nm/A]
převodní poměr	k_{gx}	76
převodní poměr	$k_{g\theta}$	275
el. odpor rotoru	$R_{a,x}$	25 [Ω]
el. odpor rotoru	$R_{a,\theta}$	0,5 [Ω]
zesílení motoru	$G_{a,x}$	20,5
zesílení motoru	$G_{a,\theta}$	11

Tab. I: Parametry modelu - indexy x a θ značí příslušnost k pohybu (motoru) v též směru

je naznačeno v práci [1]. Parametry modelu jsou společně s jejich hodnotami uvedeny v tabulce I. Hodnoty některých parametrů byly převzaty z práce [2], zbylé byly doměřeny, zjištěny experimentálně nebo odhadnuty.

V. NELINEÁRNÍ MODEL

Vlastních pohybové rovnice dále poslouží k sestavení samotného matematického modelu. Pro interaktivní použití a práci s modelem může být využito prostředí Matlab Simulink, kde ho lze poskládat z funkčních bloků. Vzhledem k silné nelinearitě modelu a složitosti rovnic je vhodné sestavit nelineární model pomocí programovatelného bloku namísto skládání z dílčích bloků, jak je zvykem u jednoduchých modelů. Pro tento účel byl použit blok Matlab function, který je možné vygenerovat tak, aby měl nastavené předem definované vstupy, výstupy a obsahoval celý matematický model. Nelineární model je zobrazen na obrázku 3.



Obrázek 3: Zapojení nelineárního modelu pomocí programovatelného bloku *Matlab function* - prostředí Simulink

Díky tomuto postupu je možné s modelem pracovat velice efektivně. Po změně parametrů, kinematického popisu soustavy nebo kterékoliv rovnice z výpočtu, je možné jedním spuštěním skriptu aktualizovat nelineární model bez nutnosti změn v Simulinku. To dovoluje rychle a přehledně ladit parametry modelu, testovat vliv (ne)zanedbnání libovolných vlivů apod.

Nelineární model byl vytvořen zejména pro ověření funkčnosti dynamického popisu soustavy. Z průběhu jednotlivých veličin v závislosti na čase můžeme porovnat, zda průběhy odpovídají představám o tom, jak by se měl model chovat. Dále je možné průběhy porovnat s chováním reálné soustavy, pokud je k dispozici.

VI. STAVOVÝ POPIS SYSTÉMU

Pro stavový popis byly definovány následující stavové proměnné. Vektor $\mathbf{u}(t)$ obsahuje vstupní proměnné, vektor $\mathbf{y}(t)$ pak výstupní měřitelné proměnné systému.

$$\mathbf{x}(t) = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{pmatrix} = \begin{pmatrix} x \\ \alpha \\ \beta \\ \theta \\ \dot{x} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\theta} \end{pmatrix} \quad (9)$$

$$\mathbf{u}(t) = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} U_x \\ U_\theta \end{pmatrix} \quad (10)$$

$$\mathbf{y}(t) = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \quad (11)$$

Jedním z hlavních požadavků na řízení jeřábu je eliminace kývání břemene a hodnoty úhlu α a β tedy nabývají pouze malých hodnot. Získané diferenciální rovnice jsou velmi složité a pro další práci s nimi může být výhodné použít aproximaci goniometrických funkcí pro malé úhly podle obecného vzorce $\sin(\gamma) \approx \gamma$ a $\cos(\gamma) \approx 1$, kde γ je malý úhel.[3] V případě této práce zmíněná úprava není nutná, protože všechny výpočty jsou provedeny programově a MATLAB disponuje dostatečně velkým výpočetním výkonem. I po aproximaci jsou rovnice příliš dlouhé na zobrazení v rámci tohoto článku.

Stavový popis systému je reprezentován dvěma základními rovnicemi (12) a (13)

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (12)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t), \quad (13)$$

kde matice \mathbf{A} se nazývá matice systému, \mathbf{B} je matice řízení, \mathbf{C} matice vazeb výstupu na stav a \mathbf{D} matice vazeb vstupu na výstup, která má nulovou hodnotu a rozměry 4×2 . Hodnoty matic pro tento systém jsou popsány (14) - (16). Matice (14) a (15) byly získány pomocí tzv. Jacobiho matice.

Ta je výsledkem parciálních derivací vlastních pohybových rovnic podle stavových proměnných za dosazení pracovního bodu. Maticí \mathbf{C} jsou definovány výstupy lineárního modelu a je možné jí modifikovat dle potřeby. V případě (16) jsou výstupními veličinami x , α , β a θ .

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -\frac{231}{50} & 0 & 0 & -\frac{1378}{5} & -\frac{1}{5000} & 0 & 0 \\ 0 & -\frac{3179}{100} & 0 & 0 & -\frac{15176}{25} & -\frac{1}{1000} & 0 & 0 \\ 0 & 0 & -\frac{1111}{50} & 0 & 0 & 0 & -\frac{23}{100} & \frac{283}{20} \\ 0 & 0 & \frac{3}{5} & 0 & 0 & 0 & \frac{1}{100} & -\frac{683}{50} \end{pmatrix} \quad (14)$$

$$\mathbf{B} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{4701}{100} & 0 \\ \frac{5177}{50} & 0 \\ 0 & -\frac{353}{25} \\ 0 & \frac{341}{25} \end{pmatrix} \quad (15)$$

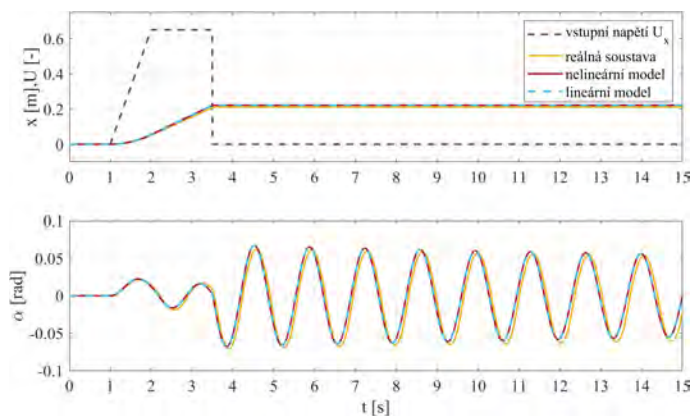
$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (16)$$

Stejně jako nelineární, i tento model byl sestaven tak, aby byl automaticky aktualizován po každé změně v odvození pohybových rovnic. K jeho vytvoření v prostředí Simulink lze použít několik variant, z nichž nejjednodušší je použití bloku *state-space*, do kterého jsou přímo dosazeny matice $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ a počáteční podmínky.

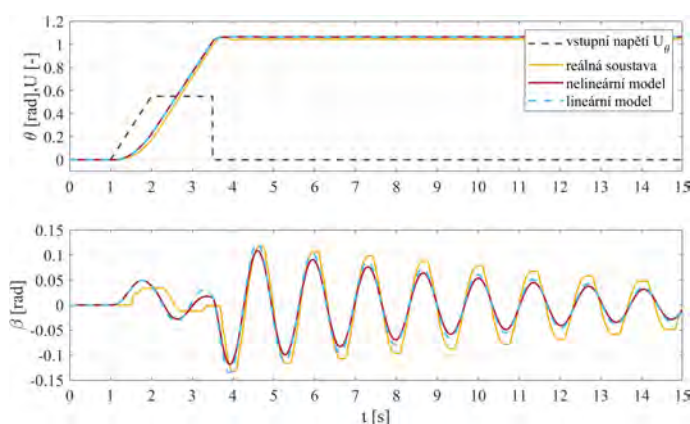
VII. POROVNÁNÍ MODELŮ S REÁLNOU SOUSTAVOU

Porovnání nelineárního a lineárního modelu s naměřenými daty reálné soustavy je zobrazeno na obrázku 4 pro pohyb ve směru x a 5 pro pohyb ve směru θ . Vynesené průběhy jsou odezvou na vstupní signál, který byl zvolen jako náběhová funkce se skokovou změnou, což je signál dostatečně složitý na správné ověření přesnosti modelů. Po prvním měření bylo provedení „fitování“ modelu - některé parametry modelu byly upraveny tak, aby jeho chování více odpovídalo realitě. Finální verze modelu je použita v tomto porovnání.

Z porovnání je zřejmé, že model funguje správně a chování kopíruje reálnou soustavu dostatečně přesně. Podle očekávání je pohyb ve směru x simulován mnohem přesněji, protože v něm nefigurují tak výrazně nelineární členy. Oproti tomu rotace ve směru θ má nelinearitu výraznější, což se podepíše



Obrázek 4: Porovnání modelů s reálnou soustavou - pohyb ve směru x a příslušný úhel α



Obrázek 5: Porovnání modelů s reálnou soustavou - pohyb ve směru θ a příslušný úhel β

také na větší odchylce mezi nelineárním a linearizovaným modelem. I v případě pohybu ve směru θ bude model pro návrh regulace účinný, protože u regulace harmonicky kmitajících pohybů záleží především na frekvenci kmitů, která je zde simulována s dobrou přesností.

VIII. ZÁVĚR

Představený model byl úspěšně porovnán s reálnou soustavou a cíl práce byl tímto splněn. V návaznosti na tuto práci může být model použit na simulaci odezvy na různé vstupy, ale především umožňuje návrh vhodné regulace, což byla hlavní motivace k jeho vytvoření. Byly vytvořeny dvě varianty modelu, z nichž nelineární poslouží lépe k simulaci chování soustavy, naopak linearizovaný se hodí lépe právě k ladění a návrhu regulačních metod. Plánované řízení se bude týkat regulace polohy a eliminace kývání břemene. Příkladem plánovaných řídicích metod mohou být PID regulátor a tvarovače vstupního signálu (input shapery).

REFERENCE

[1] Breuning, P (2015) Linear model predictive control of a 3D tower crane for educational use. PhD Thesis, University of Stuttgart, Stuttgart, Germany.

[2] Iles, S., Matusko, J., & Kolonic, F. (2014). Real-time Predictive Control of 3D tower crane. 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE). doi:10.1109/isie.2014.6864615

[3] Win, T. M., Hesketh, T., & Eaton, R. (2016). Robotic Tower Crane Modeling and Control (RTCMC) with LQR-DRO and LQR-LEIC for Linear and Nonlinear Payload Swing Minimization. International Review of Automatic Control (IREACO). doi:10.15866/ireaco.v9i2.8431

[4] INTECO, Tower Crane User's Manual, 9.6 Edition. <http://www.inteco.com.pl/>

Poděkování

Tato práce byla podpořena grantem Studentské grantové soutěže ČVUT č. SGS SGS20/159/OHK2/3T/12.



Editor: Vladimír Hlaváč
Název díla: Nové metody a postupy v oblasti přístrojové techniky,
automatického řízení a informatiky 2021
Vydalo: České vysoké učení technické v Praze
Zpracoval: Ústav přístrojové a řídicí techniky Fakulty strojní ČVUT v Praze
Kontaktní adresa: Ústav přístrojové a řídicí techniky, FS ČVUT v Praze, Technická 4, Praha 6
Tel.: +420 2 2435 9750
Počet stran: 153 Pořadí vydání: 1
Vydáno dne 1. 10. 2021 jako pdf soubor na CD.
Dostupné na webových stránkách Ústavu přístrojové a řídicí techniky: <http://iat.fs.cvut.cz/nmp>
ISBN 978-80-01-06889-2