TEST RIG OF HEATING SYSTEM FOR CONTROL DESIGN TASK USING REAL TIME TOOLBOX

Radovan Liska, Stanislav Lang, Radomil Matousek, Petr Minar

Brno University of Technology, Faculty of Mechanical Engineering,
Department of Applied Computer Science

Abstract

This paper deals with possible usage of the Real Time Toolbox (RTT) by Humusoft company for the control design of education test rig. The test rig is a heating system, where source of the warm is a light bulb as the action devices, the fan and surroundings are the control disturbance. The goal of the control is to reach desired value of temperature in use sensor. The sensor is in our case the controlled plant. The various controllers are designed in Simulink environment and by means of RTT together with data acquisition board MF624. The primary reason for this paper is to present our new education test rig in case of control design education.

Keywords: Control design, Test rig, Real Time Toolbox, MF 624

1. Introduction

There are many possibilities to present control design theory in education area. In this paper we present complete control loop which is based on the sophisticated simulation software Simulink with connection of Real Time Toolbox and data acquisition board MF624. The test rig is presented by Fig. 1. It is SISO system where the action value is the PWM signal for halogen light bulb control and outputs signal is the temperature of the plant. The plant is integrated sensor SMT160-30-92 with temperature range from -45^{o}C to 150 ^{o}C and resolution better then 0.005 ^{o}C.

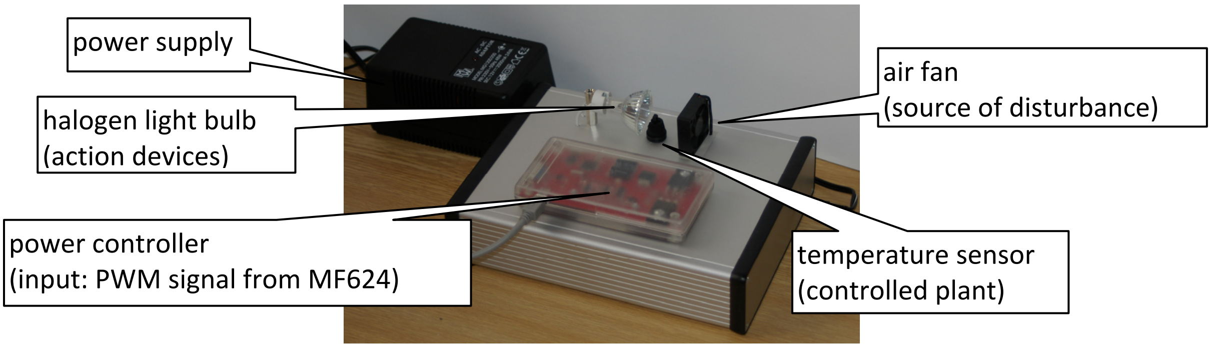


Figure 1: Heating system as a test rig

Artificial disturbance is realized as controlled fan and can be used in many ways (continues flow, pulse blow, noise, etc.). Also fluctuation of surrounding temperature can be included as disturbance. Purpose of the realized task is to present to students the identification and control design possibility based on very simple controlled plant.

Simulink contains many components for simulation. For connection with Test rig are used defined functions primitives – written as m-files, and components in Simulink which are supported by Humusoft in Real Time Toolbox Library. Overall view of logic connection with main components is displayed in Fig 2. As we can see there is two way communication. Data are measured from physical model, real time computed and sent through Humusoft's card (Adapter) to Real Time Toolbox in Simulink. In Simulink is created a visual schematic to control, measure and display values.

Schematic in Simulink is divided into 3 parts:

- Adjusting requested values, in this case adjusted value is Temperature and Fan speed, which actuate as disturbance.
- Plant with model and Regulator
- Scope with Output values history and temperature output.

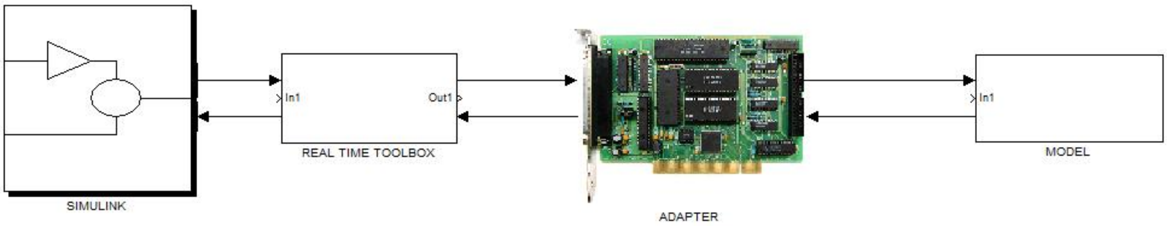


Figure 2: Schematic block connection

2. Building the Matlab/Simulink model schematic

2.1 Real Plant is modeled in Sub-block called Plant (attachment 1)

For communication with model is used Real Time Toolbox Card MF624. Temperature is read with S-function called "*CardInterface.m*" which is also with two input feeded. The first input is for halogen light and the second one is for fan.
Temperature is obtained from sensor SMT160-30-92. The output of this sensor is duty-cycle, which is linear to value of temperature. The higher temperature is considered as a higher duty-cycle. In script for obtaining duty is called primitive function *AIRead.m*, which is written and supplied by Humusoft.
After the obtained duty, there is need to compute the output temperature which will be displayed to user and used in simulation as well. Used formula is:

$$duty = 0.32 + 0.0047\ x\ t, \qquad (1)$$

where t is a temperature.

There is need to re-calculate formula to temperature. New formula is

$$temperature = (duty - 0.32)/0.0047 \qquad (2)$$

For measuring mean value of PWM signal there is used analog input port. The Mean value is considered as an output of Lowpass Filter. Input to this filter is a PWM signal from sensor with range from 1kHz to 4kHz.

The first Input parameter is fully delivered to the function *PWMWrite.m* (also written by Humusoft), where this function set up the value of PWM to the light. Need to notice that expected values to this function are from range 0-1. Because of this fact we wrote a function to calculate the action value from Regulator with its output range 0 – 100 to new value 0-1.
The second parameter serves as an input for Fan. Its limitations and applied algorithm are similar to behavior of the first Input.

2.2 Model Plant is modeled in Sub-block called model of the plant 1.order (attachment 1)

If we are talking about identification, it means we have data of this process. It is necessary to have a model of plant if we want to work with plant and do an analysis of the process including regulation. We consider static characteristics and dynamic characteristics. The work is focused to transfer function calculation of plant model. There is several possibilities how to identify the process. Two cases are chosen in this case.

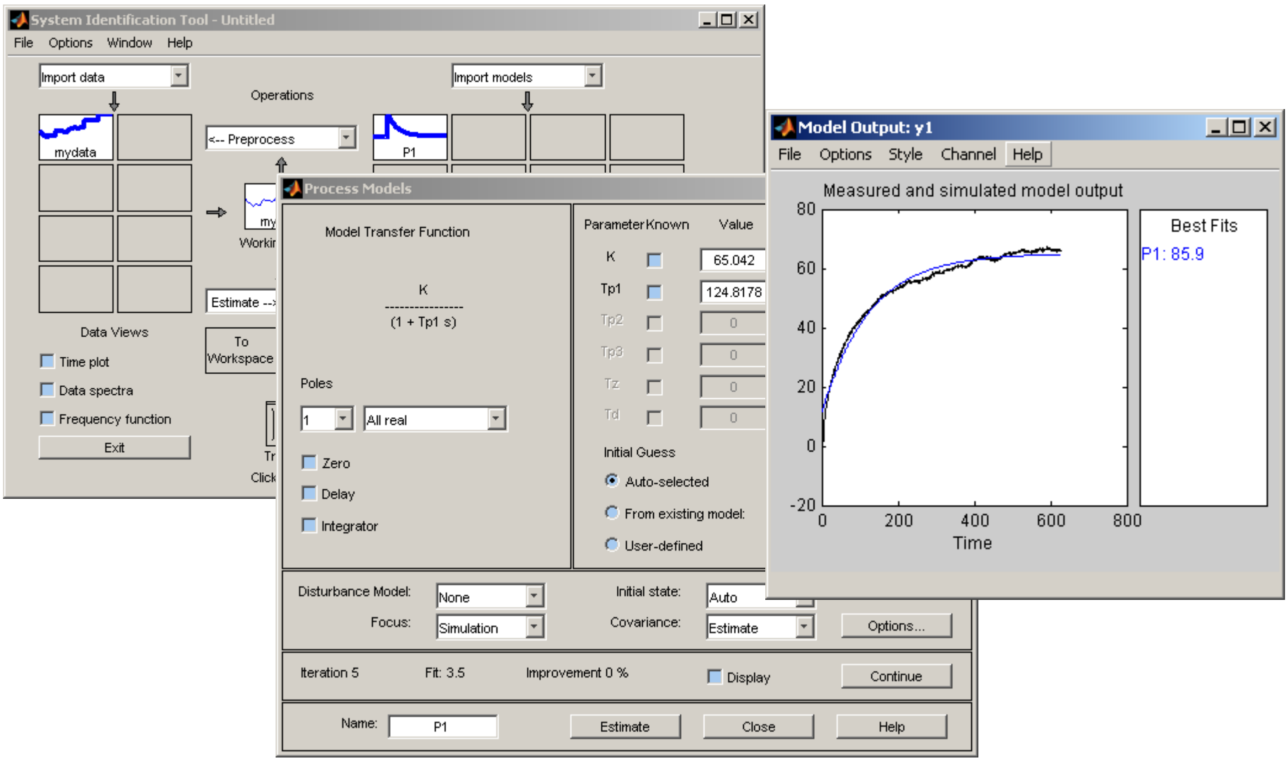
Identification via Matlab Identification Toolbox and step response calculation to the output of plant:

The modeling of real plant consist of several steps:

- Data acquisition
- Data Computing and creating output model response
- Choosing the best fitted model

Data acquisition: The input of real plant was fed by step response. In this case it means the full power of light was turned on. After some time has passed and the response of the plant went to steady-state condition the simulation has been stopped. Time for steady-state condition after step response is 600 seconds.

Data Computing and creating output model response: In this case Matlab identification toolbox was used. Imported data were sampled with 0.1sec. Measured data were slightly dispersed and contains noise which was added during sampling. The first look at the step response implied to identify this plant to the first order, also because of non zero derivation. Identification toolbox estimator was used to find correct parameters for transfer function. Transfer function for the first order F(p) = K /(T*p+1). Parameters for transfer function: K = 65, T = 125s.



Model Transfer Function

$$\frac{K}{(1 + Tp1\ s)}$$

Parameter	Known	Value
K	□	65.042
Tp1	□	124.8178
Tp2	□	0
Tp3	□	0
Tz	□	0
Td	□	0

Poles: 1, All real

□ Zero

□ Delay

□ Integrator

Initial Guess

● Auto-selected

○ From existing model:

○ User-defined

Disturbance Model: None

Focus: Simulation

Initial state: Auto

Covariance: Estimate

Iteration 5   Fit: 3.5   Improvement 0 %   □ Display

Name: P1

Best Fits

P1: 85.9

Figure 3: Identification toolbox

Other way to calculate transfer function from step response:

| In the picture Fig.4 is shown step response to the plant and lines to proceed with calculation the transfer function. Linear system is possible to identify by two ways. The first one is using the direction of tangent line and the second one is to find the value of 63% of steady-state condition. Both methods are for the system of first order identical and return the same result. From the figure bellow, is obvious that the mentioned methods are not same.

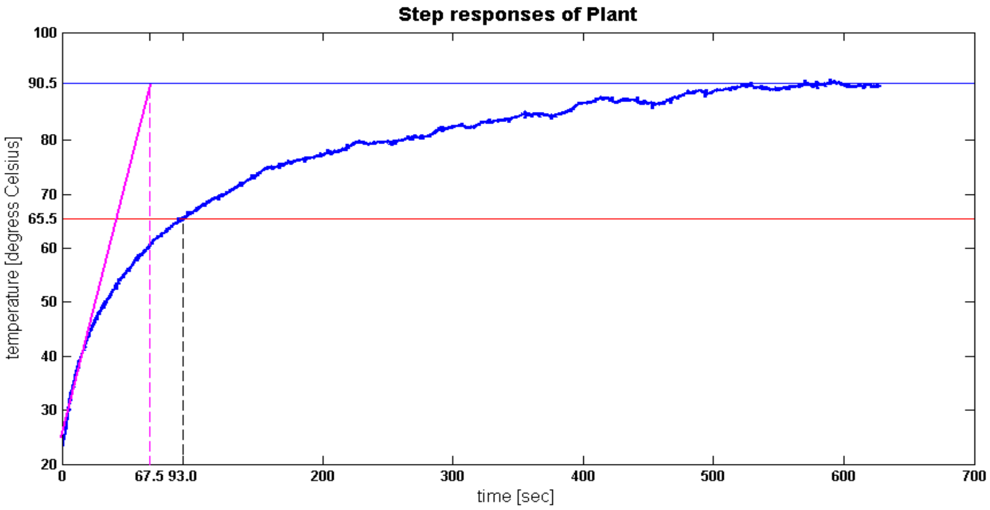


Figure 4: Identification from step response

The reason why methods are not the same is caused of nonlinearity of identified system. This non linearity is shown in the Fig. 5.

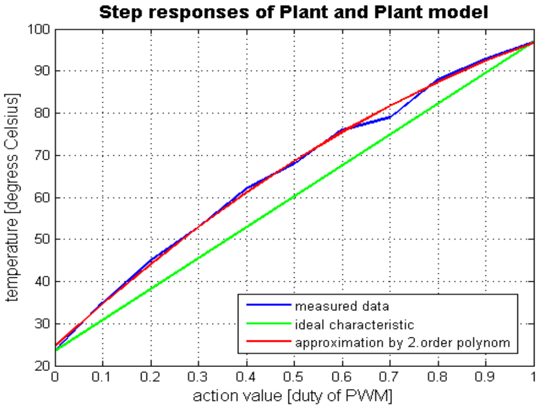


Figure 5: Static characteristics of the Plant

Non linearity of model is not significant and it is possible to approximate it as linear characteristic. Result of method 63% steady-state condition is : F(p) = K /(T*p+1) where K = 67, T = 93s

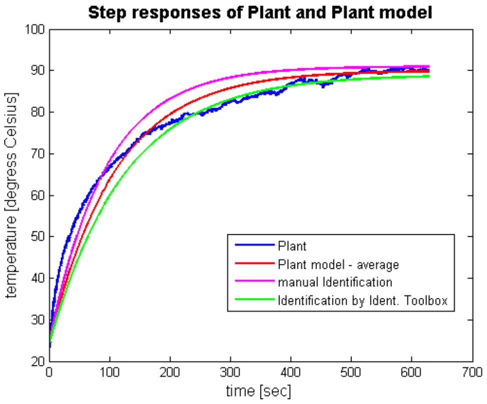


Fig. 6 Approximations by linear systems

As the final result of approximation transfer function is considered mean of values from calculated transfer functions.

2.3 Controls

Slider gain Light for desired Value of Temperature: This is interactive slider gain, where user can set up the desired value of temperature. Its lower range depends from environment and the maximum temperature is limited by power of light. In this case it is 100 degrees Celsius.

2.4 RT. Sync Component

RT Sync: Is a very useful extension of real-time toolbox. This block is synchronizing time of simulation with real-time. Displayed output time can be considered for a 1 s.

2.5 Regulators

Model Plant and real Plant has been tested for different conditions, such as adding more disturbance by fan. Three regulators were designed to control the output of the plant : Matlab tuned regulators and manually computed regulator PI.

Let’s consider manually computed regulator. There was used tangent method.

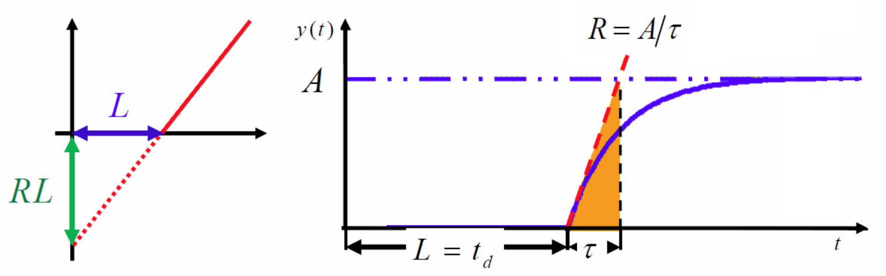


Figure 7: Tangent method

- The First, there is need to obtain step response to the plant.
- The Second, the tangent is drawn to the inflexion point
- The Third, values R, L are obtained
- The Fourth, calculate the gains according to estimations show bellow

P $k_P = 1/RL$

PI $k_P = 0.9/RL, T_I = 3L$

PID $k_P = 1.2/RL, T_I = 2L, T_D = 0.5L$

$$D_C(s) = k_P\left[\underset{P}{1} + \underset{I}{\frac{1}{T_I s}} + \underset{D}{T_D s}\right]$$

Transfer function for Regulator is designed with parameters: kp = 10, T_I = 3.

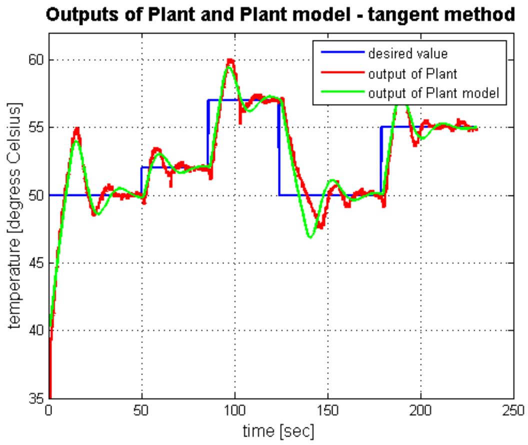


Figure 8: Outputs of plant and Plant model to desire value

Another approach to obtain parameters for PI is using Matlab/Simulink tune function.

For PID tuning is used model of plant, where tuning function can work with a Simulink time. The next step is need to limit action value due to dynamic of acting. Need to notice that plant include saturation in the input.

PI regulator is used. Regulation works fine for adjusting temperature to higher value. It is controlled by PI regulator, which has an information about output of plant. Input to the regulator is deviation, which means difference between desired and output value. On the other side acting regulator to the adjusting temperature to lower value is restricted by inertia of the plant. This is shown in Fig. 8. Output of plant with default regulator is shown in purple color. There was also tuned new regulator, which include parameters for faster step response. It is colored with red color. For the comparison in the same figure is shown the output of plant with regulator designed by manual tangent method.

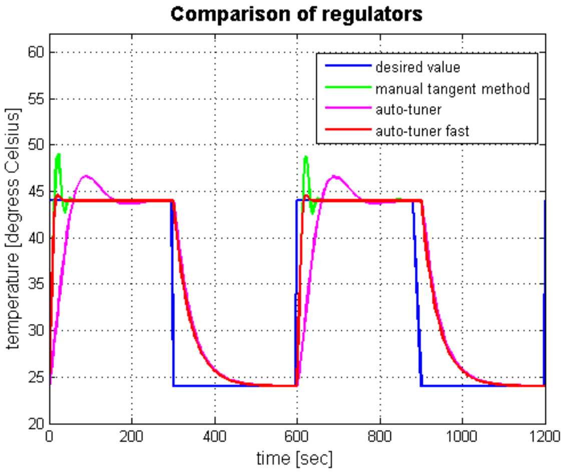


Figure 9: Comparison of regulators

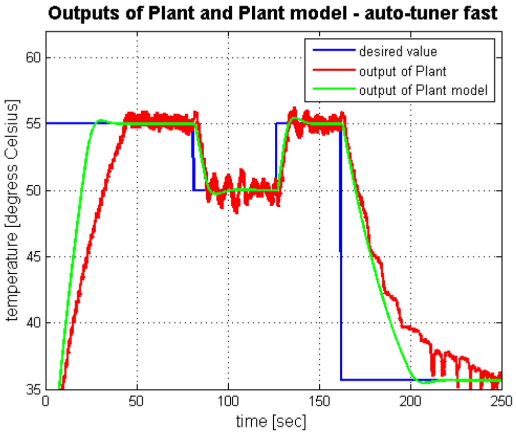


Figure 10: Outputs of plant and Plant model to desire value

In the model of plant the regulator designed by Matlab with fast response was looked very well. But when its regulator was acting with real plant the output was not too slight. The desired value wasn’t followed with good accuracy. The gain of the P part of regulator was too high and because of it. A high value of P part of PI regulator is not so appropriate, because the non linearity of model and noise of temperature sensor has become evident.

3. Conclusion

This work is focused to controlling plant by regulators in real time. Students have a possibility to adjust regulators and watch response of plant, actual value produced by regulator and other outputs in real time. They have also chance to identify the real plant as a model. This work offer to students meet with other powerful tools in Matlab and Simulink, such as Matlab Identification Toolbox, Tuning of regulator and many others.

Real Time toolbox is ideal tool for displaying and controlling physical processes, especially for test rig of heating system. This toolbox was used to predict model of the plant. For demonstrating identification by Matlab toolbox or other mentioned methods would be better to use for more linear systems, but in this case of identification, the students can meet with non linear system, which can be approximated by linear system. In this paper are presented two ways of identification. The final transfer function of the plant is used the mean of values from calculated transfer functions. Model of plant was designed by block of integrators with gains and other logic to acquire non zeros start conditions. This is the initial development of the test rig of heating system. There are other innovations how to improve the regulation, e.g. Fan can be used for acting in case of negative actual value, which cause faster cooling of system. The fan is now acting as a disturbance.

In the work was shown tangent method to calculate transfer function of Regulator. As a comparison are shown methods using Matlab tuning function. There is a lot of possibilities how to set up PI regulator. In the fig 10 is displayed the output of too fast regulator. There is need to consider the limitations of plant for adjusting the regulator.

From the Matlab point of view it is also possible to use Virtual reality toolbox, RT windows target system. Better usage could represent advanced solutions, which means connecting with dSpace or programming automats supported by B&R Automation.

Acknowledgement

This work was supported by the research projects of MSM 0021630529 "Intelligent Systems in Automation", GACR No.: 102/091668 "Evolutionary Control Design", and IGA (Internal Grant Agency of Brno University of Technology) FSI-S-11-31 "Application of Artificial Intelligence".

Reference

[1] C Charles; D Frederick, J Newell. *Modeling and Analysis of Dynamic Systems*. USA, WILEY, 2008. 576 s.

[2] M Šebek. *Automatické řízení* [in czech]. CZ, ČVUT, 2011.

Address: Technicka 2896/2, 61669 Brno, Czech Republic
E-mails: matousek@fme.vutbr.cz, {y74188, y73010, yminar08} @stud.fme.vutbr.cz