MODELING, SIMULATION AND PID CONTROL OF WATER TANK MODEL USING MATLAB AND SIMULINK

Jiri Vojtesek and Lubos Spacek
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nam. TGM 5555, 760 01 Zlin, Czech Republic
E-mail: {vojtesek,lspacek}@utb.cz

KEYWORDS
Modeling, Simulation, Matlab, Simulink, Steady-state Analysis, Dynamic Analysis, Identification, PID control.

ABSTRACT

Paper deals with the modeling and simulation of the real model of the water tank as a part of the process control teaching system PCT40. The mathematical model of the water tank was created and constructed in Matlab's simulation tool Simulink. Then the steady-state and dynamic analyses were performed and showed us limitations and behavior of the system. The simple identification in Matlab with minimization of the square of the simulation error was also introduced. Lastly, the PID control in Matlab was applied for controlling the water level in the tank. The PID block in Simulink offers also tuning of the output response by the setting of the robustness and speed of the control.

INTRODUCTION

There are two types of models (Maria, 1997). The first is the real model which is usually small representation of the real system. Experiments are then performed on this smaller system which saves time and mainly costs less because experiments need smaller amounts of inputs. Results are understood as similar to results on the real system. Experiments are sometimes more accurate then simulations because they are more realistic. On the other hand, sometimes it is not so simple to construct a good real model of the system, or it is unrealistic to do this model.

This limitation makes the space to the second type of simulation models – abstract model. An abstract model is a representation of the real system which uses mathematical expressions for describing the system. The abstract model is usually called the mathematical model and uses differential equations for the mathematical description (Ingham et al. 2000).

Computer simulation (Gould et al. 2017) is a great tool which is very famous because of its main advantages – we can save time compared with the experiments on the real model, which needs some time for preparation of the experiments, feeding, cleaning, etc. Once we have a computer model, we can perform thousands of experiments that are much quicker. Moreover, simulations are safe and mainly much cheaper because we do not need any physical parts, reactants, water, etc.

The design of the controller usually precedes steady-state analysis that observes the behavior of the system in time equal to infinity, where we suppose that the state variable does not change. This analysis could show us a limitation of the system and helps us with the choice of the optimal working point (Ingham et al. 2000).

The next step is the dynamic analysis which puts step change on the input to the system and observes the reaction of the system on this input. Resulting output response could help us for example with the choice of the optimal control strategy (Vojtesek and Dostal 2015).

Matlab from MathWorks is a multiplatform computer program for numerical computations (MathWorks 2019a). It is widely used for teaching purposes and also for research. It also has a lot of toolboxes from various fields from engineering, through statistics to economics, etc. The great tool, especially for teaching purposes, is Simulink which is a graphical programming environment that can be used for modeling, simulation, analysis, etc.

Matlab also has tools for identification of the system which can also be used in the control synthesis. We can use System Identification Toolbox (MathWorks 2019b) which provides various Matlab functions and Simulink blocks that can be used for identification.

The Proportional-Integral-Derivative (PID) controller is one of the most common controllers in the industry in the feedback control because of its advantages – it is simple, easily programmable and mostly provides sufficient results (Graf 2016). Matlab has special Control System Toolbox (MathWorks 2019c) which offers various control exercises also including PID controller. Simulink itself has a special block "PID controller" that has also tuning options where we can choose the speed of the output response and robustness.

The goal of this contribution is to give a reader an overview of the modeling, simulation, and control of the technological processes using Matlab and Simulink. Proposed methods are applied to the real model of the water tank as a simple nonlinear model described mathematically by the first order nonlinear ordinary differential question that could be solved numerically (Mathews and Fink 2004). Matlab and Simulink provide various tools that help with this task which makes them great teaching tool.

Communications of the ECMS, Volume 33, Issue 1,
Proceedings, ©ECMS Mauro Iacono, Francesco Palmieri,
Marco Gribaudo, Massimo Ficco (Editors)
ISBN: 978-3-937436-65-4/978-3-937436-66-1(CD) ISSN 2522-2414

REAL MODEL OF THE WATER TANK

The system under consideration is a water tank which represents the water dam, the water reservoir, etc. The real model of the water tank is one part of the multifunctional process control teaching system PCT40 displayed in Fig. 1 (Armfield 2005). This teaching system offers various control exercises on the heat exchanger, the continuous stirred tank reactor (CSTR) or the water tank models.

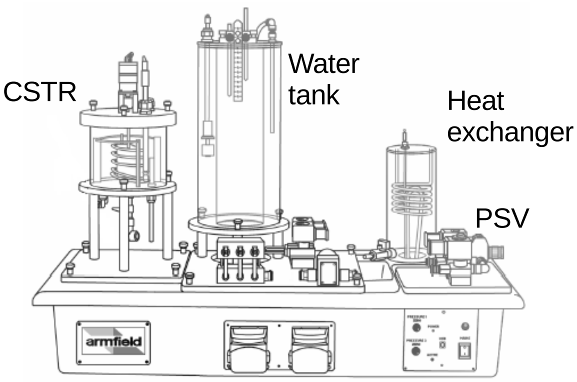


Figure 1: Multifunctional process control teaching system PCT40

Let us now focus on the water tank model, scheme of which is shown in Figure 2. This model is a transparent plastic cylinder with inner radius $r_1 = 0.087\ m$. Due to quicker dynamics and reduction of water consumption is added another transparent cylinder inside the tank.

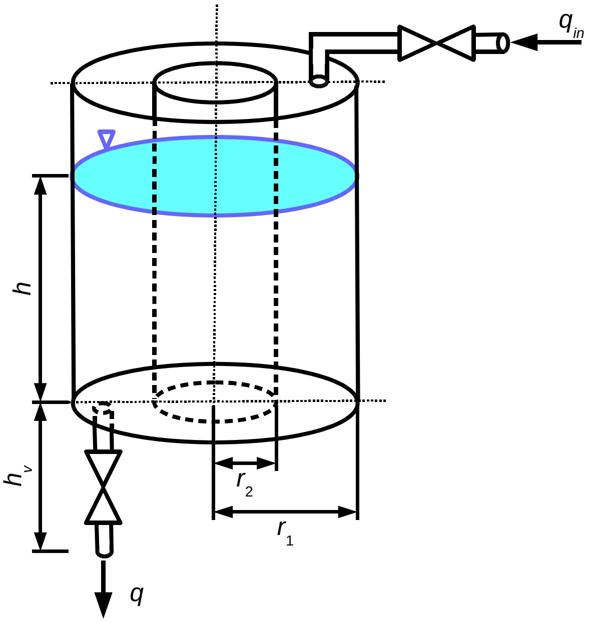
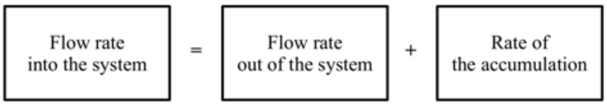


Figure 2: Scheme of the water tank model

This inner cylinder has a diameter $r_2 = 0.057\ m$ and the maximal water level in the tank is $h_{max} = 0.3\ m$ and it means that the maximal volume of this tank is $V_{max} = 0.0041\ m^3$ which is 4.1 *l*. The water could feed the tank with the input flow rate in the range $q_{in} = 0 - 1.5\ l.min^{-1}$ that is also $q_{in} = 0 - 0.0015\ m^3.min^{-1}$. The water comes out from the water tank with the output volumetric flow rate *q*.

The mathematical model is constructed with the use of material balance inside the system in the word form:



Mathematical description of this equation is

$$q_{in} = q + \frac{dV}{dt} \qquad (1)$$

with *V* as a volume of the tank and *t* as a time. The input volumetric flow rate q_{in} could be set and controlled by the Proportional Solenoid Valve (PSV) in the range:

$$q_{in} = \langle 0; 1.5 \rangle \cdot 10^{-3}\ m^3.min^{-1} \qquad (2)$$

The volume of the tank as a cylinder is computed from the area of the basement *F* and the level of the water tank *h*, i.e.

$$q_{in} = q + \frac{d(F \cdot h)}{dt} \qquad (3)$$

The area of the basement is constant and computed as

$$F = \pi \cdot r_1^2 - \pi \cdot r_2^2 = 0.0136\ m^2 \qquad (4)$$

The output flow rate from the reactor *q* is a nonlinear function of the water level *h* and the constant of the valve *k*

$$q = k \cdot \sqrt{h} \qquad (5)$$

The differential equation (3) is then rewritten to the form

$$\frac{dh}{dt} = \frac{q_{in} - k \cdot \sqrt{h}}{F} \qquad (6)$$

Unknown variable in (6) is the constant of the valve *k* that could be computed from the steady state, where input volumetric flow rate is equal to the output volumetric flow rate ($q_{in} = q$). The difference of the water level is zero in the steady-state which means that the constant of the valve is computed from

$$k = \frac{q_{in}^s}{\sqrt{h^s}} \qquad (7)$$

Our previous experiments (Vojtesek and Dostal 2015) have shown, that we must take into the consideration also the height of the output valve $h_v = 7.6\ cm = 0.0076\ m$ which means that the valve constant for flow rate $q_{in}^{\ s} = 1.008 \cdot 10^{-3}\ m^3.min^{-1}$ and measured height $h^s = 0.195\ m$ is computed as

$$k = \frac{q_{in}^s}{\sqrt{h^s + h_v}} = \frac{1.008 \cdot 10^{-3}}{\sqrt{0.195 + 0.076}} = 1.93 \cdot 10^{-3}\, m^{5/2} min^{-1} \qquad (8)$$

We can say, that this system is a nonlinear system because the mathematical model (6) is a nonlinear differential equation of the state variable h as a water level.

Steady-state Analysis

The steady-state analysis is the first step of the simulation. The steady-state is a state in time $t = \infty$ which means that the derivative concerning time is 0 and the differential equation (6) is then transformed to the equation where steady-state water level h^s is a nonlinear function of the input volumetric flow rate q_{in}

$$h^s\left(q_{in}\right) = \left(\frac{q_{in}}{k_n}\right)^2 \tag{9}$$

This steady-state function could be modeled with the use of Simulink – the input variable is volumetric flow rate q_{in}, k_n is constant, and the Simulink scheme is shown in Figure 3.

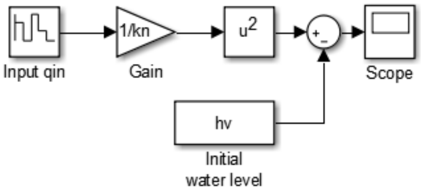


Figure 3: Simulink scheme for Steady-state analysis

It was already mentioned, that the height of the output valve must be taken into account in the computation, but the water level in the real model of the tank is measured from the bottom of the water tank. That is why we add constant value h_v representing the height of the valve which is subtracted from the currently computed water level. The results are displayed in the scope.

Dynamic Analysis

The dynamic analysis is the second simulation step after the steady-state analysis. Mathematically speaking, the dynamic analysis means a numerical solution of the mathematical model represented by the differential equation (6). For example, Runge-Kutta's method could be used for this job.

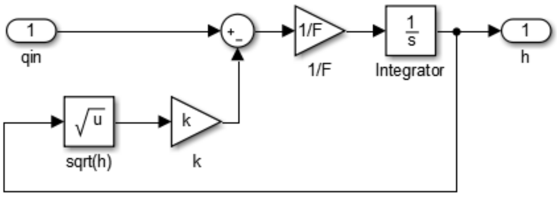


Figure 4: Simulink scheme of the mathematical model

The goal of this contribution is to show the usability of the Simulink as a simulation tool. Figure 4 shows the scheme in the Simulink that is equivalent to the differential equation (6) where the block "Integrator" represents the derivative of the water level h.

Simulink offers to create a new block by making „mask" of blocks. We can create block Water tank from blocks displayed in Figure 4. An advantage of this feature is that the scheme is then "cleaner" and we can also create the library of blocks that could be used for other simulation experiments such as identification, control, etc. The block scheme used for simulation of the dynamic behavior is shown in Figure 5.

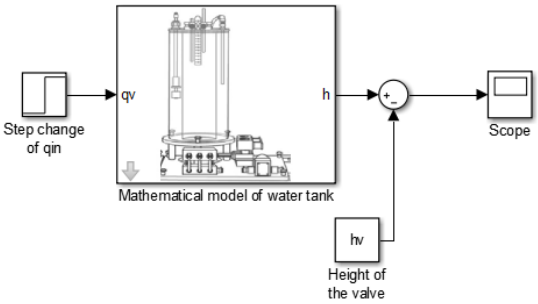


Figure 5: Simulink scheme for dynamic analysis

The dynamic analysis explores the behavior of the system after the step change of the input variable, in our case input volumetric flow rate. This step change is the first block on the left side. The second block in Figure 5 represents the mathematical model of the water tank, i.e., masked scheme from Figure 4. An input to this block is the step change of $q_{in,}$ and the output is the course of the state variable (a water level h) in time. This output variable is reduced by subtracting the height of the valve h_v = 0.0076 m. This reduction was made mainly from the practical point of view – the real model of the water tank has a scale on the shell of the cylinder which has 0 at the bottom of the tank, but zero points for this model is at the valve mouth which is just below the height of the valve.

Identification

Identification is the next step after the dynamic analysis. It helps us with the mathematical description of the system that can be used later in the control synthesis.

One option is to describe the course of the output variable by the transfer function in the continuous-time form, generally

$$G(s) = \frac{b(s)}{a(s)} \tag{10}$$

Where degrees of polynomials $a(s)$ and $b(s)$ must fulfill the condition $\deg a(s) \geq \deg b(s)$. These degrees could be guessed from the output response in the dynamic analysis.

Parameters of polynomials $a(s)$ and $b(s)$ could be then identified for example from dynamic step responses. The Least-squares method is suitable for this task because it is easily programmable and simple with good results.

Let us now find appropriate functions that can be used for identification. The dynamic analysis produces time vector t, a vector of input u and output variables y.

Let the square of the difference between original output y and output generated by from the identified transfer function $G(s)$ with the same input u. This output could be denoted y_{id} and the sum of these differences for the whole time interval $t = <0, t_f>$ is then

$$\varepsilon = \sum_{t=0}^{t_f} \left(y(t) - y_{id}(t) \right)^2 \tag{11}$$

The task of the Least squares method is to find the minimum of ε by changing the transfer function $G(s)$. Matlab can do this minimization for example with the use function `fminsearch` which uses nonlinear programming solver for searching for the minimum of a function using derivative-free method (Lagaris et al. 1998)(MathWorks 2019d). The syntax of this function is

```
x = fminsearch(fun,x0)
```

where `fun` defines the function of `x` to minimize, and `x0` are initial value or values of unknown `x`. Usage of this function in our concrete example is shown later in the Simulation results part.

Control of the Plant

The last step after the identification is the control of the system. Our previous papers present the use of the adaptive control for controlling this system (Vojtesek and Dostal 2015), but now we want to use Matlab and Simulink for controlling this system. Simulink has several control blocks; one of the simplest is the PID (Proportional-Integral-Derivative) controller. PID control in Simulink is very user-friendly and offers "tuning" options that will be shown later in the results part.

Continuous-time PID controller uses compensator formula (MathWorks 2019e)

$$P + I\frac{1}{s} + D\frac{N}{1 + N\frac{1}{s}} \tag{12}$$

Where P, I and D are appropriate coefficients of PID controller and N is a derivative filter coefficient.

SIMULATION RESULTS

The simulations verify analyses introduced in the previous chapter on the mathematical model (6). Results are shown in the next chapters.

Steady-state Analysis

The steady-state analysis was performed in Simulink scheme displayed in Figure 3. The input variable in the range of Equation (2) and results from this analysis are shown in Figure 6.

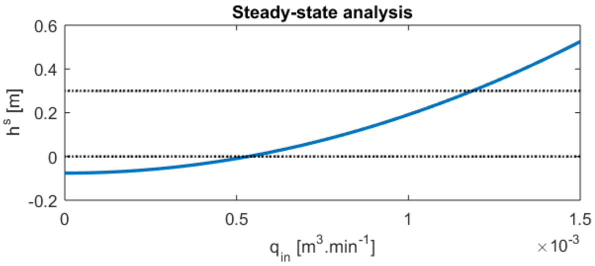


Figure 6: Result of steady-state analysis

We can see expected nonlinear behavior of the water tank, and the simulation experiment also shows, that optimal working interval of the input variable is q_{in} = <0.531; 1.186>·10^{-3} $m^3.min^{-1}$ because lower flow rate has not enough power to reach a minimum level in the tank and higher bound on the other hand results in the water level h^s more than 0.3 m which is maximal water level in the tank.

This range could be understood as the main result of the steady-state analysis. It makes no sense to input lower and higher volumetric flow rate that this range.

Dynamic Analysis

The static characteristic also shows that ideal working point is in the middle of the optimal working interval, i.e., $q_{in} = 0.903 \cdot 10^{-3}$ $m^3.min^{-1}$ and its steady-state water level inside the tank $h^s = 0.14$ m. Simulation time was 50 min.

We have performed several step changes of the input flow rate u_p = [30% 15% -15% -30%], and the input volumetric flow rate inside the system is then

$$q_{in} = q_{in}^s + \frac{u_p \cdot q_{in}^s}{100} \left[m^3 \cdot min^{-1} \right] \tag{13}$$

An analysis was done by Simulink with the scheme from Figure 5 and the results are shown in Figure 7.

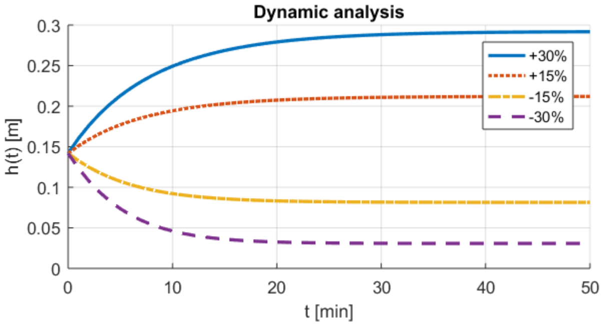


Figure 7: Results of the dynamic analysis

Results of dynamic analysis help us with the understanding of systems behavior after the change of the input variable. We can see that the positive step of the input variable produces a positive step of the output variable and oppositely, the negative step of input decreases the value of the output variable. We must also mention that the same positive step did not result in the same difference of the output variable as for negative step change.

A practical discovery from this analysis is the real limitation of the output variable u that could be for this case in the range u_p = <-40, 32> % because lower and higher values will then again result in output water level that is out of the physical range of the water tank where h = <0, 0.3> m.

In this case, results in Figure 7 tells us that the system could be described by the first order transfer function

$$G(s) = \frac{b(s)}{a(s)} = \frac{b_0}{a_1 s + a_0} \tag{14}$$

This trasnfer function is later used for identification.

Identification

The vector of parameters for identification has for the chosen transfer function (14) three parts

$$\hat{\theta} = [b_0, a_1, a_0] \tag{15}$$

And our task is to estimate this vector by the minimizing of the value ε in (11).

The theoretical part introduces Matlab function `fminsearch` but before we will use it, we must define the minimization function `fG`

```
fG = @(X) tf(X(1),[X(2) X(3)]);
```

where `@(X)` denotes so called *anonymous function* of the variable `X` and this function has form of the transfer function $G(s)$ in Equation (14) which is in Matlab constructed via function `tf`. X is vector of parameters (15) which means that `X(1)` = b_0, `X(2)` = a_1, `X(3)` = a_0.

Initial values of this vestor are `X = [50 10 2];`

Optimization is then performed by running this line

```
X = fminsearch(@(x) krit(x,fG,t,u,y),X);
```

where `krit` defines subfunction for computing sum of squeres (11), i.e.

```
function eps = krit(x, fG, t, u, y)
    G = fG(x);
    yid = lsim(G,u,t);
    epsilon = sum((y-yid).^2);
end
```

for `t` as a vector of time, `u` is a vector of the input variable, and `y` is output variable. Function `lsim` in the third line computes estimated output variable y_{id} for the same time interval `t`, input variable `u` and defined transfer function (14). The computed and estimated output for the step change u_p = +30 % is shown in Figure 8.

The vector of identified parameters of the transfer function is

$$\hat{\theta} = [b_0, a_1, a_0] = [9917, 614.2, 1.3] \tag{16}$$

and the transfer function is then

$$G(s) = \frac{b_0}{a_1 s + a_0} = \frac{9917}{614.2s + 1.3} \tag{17}$$

We can see that the last identified parameter a_0 is much lower than the others and we can set it equal to 1, e.g. $a_0 = 1$. The identified vector could be then $\hat{\theta} = [b_0, a_1, 1]$ and computation is a bit quicker.

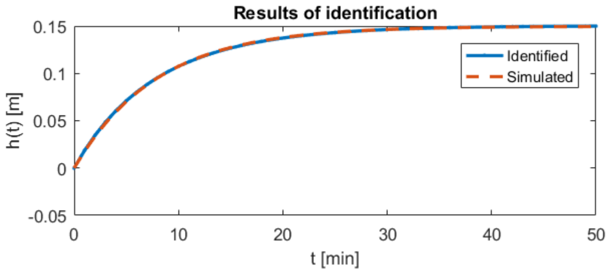


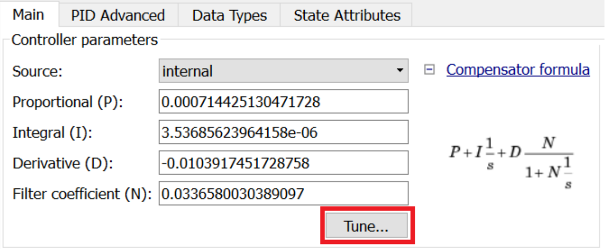
Figure 8: Results of identification for step change of input variable u_p = +30%

Note, that the output variable starts from 0 because of the identification reasons. Identified parameters for other step changes are very similar. The courses of the simulated and identified output variable in Figure 8 are nearly the same which means, that parameters of the identified transfer function are good.

Control of the Plant

It was already mentioned, that PID control can be used for controlling this plant. Simulink has build-in block PID controller and the feedback control scheme is shown in Figure 10.

This block has one big advantage – it has a tuning option that helps us with the setting of the P, I, D and N-parts. Part of the options window that sets the PID controller is shown in Figure 9.



Controller parameters

Source: internal

Proportional (P): 0.000714425130471728

Integral (I): 3.53685623964158e-06

Derivative (D): -0.0103917451728758

Filter coefficient (N): 0.0336580030389097

$$P + I\frac{1}{s} + D\frac{N}{1 + N\frac{1}{s}}$$

Figure 9: Setting of the PID controller in Simulink

You can see values of P, I, D parts and also button “Tune…” that computes these values by simulation experiments on the controlled systems. We can see, that there are two branches in the scheme – the first one directly through the controlled system and the second branch goes through transfer function parameters of which are the same as in (17). This branch is in this scheme just because of this tuning. If we want to tune parameters of the PID controller, we must double click on the Manual switch block which connects this “identification branch” and PID block simulates the

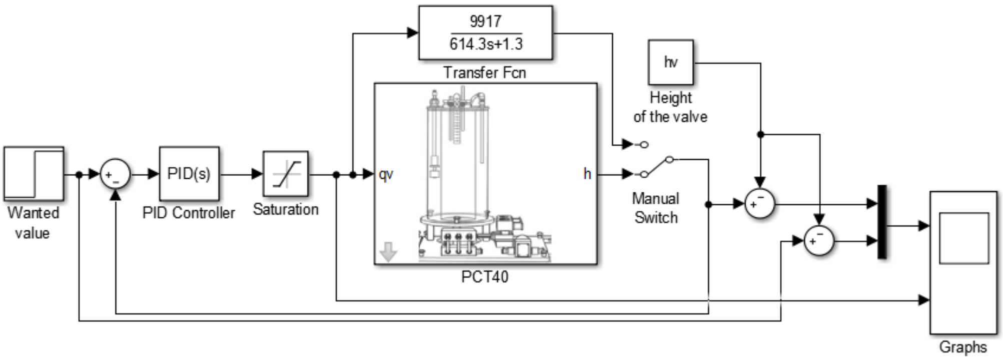
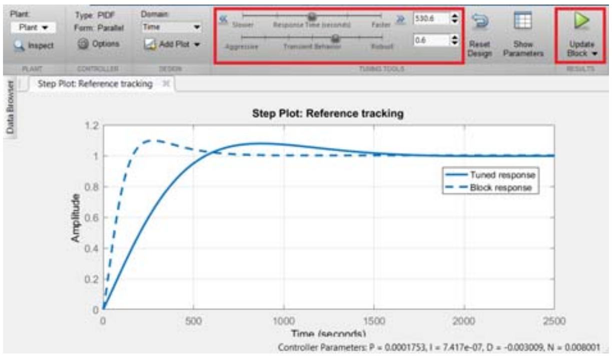


Figure 10: Feedback control scheme in Simulink

behaviour of the system for various PID setting. In the next window (Figure 11) we can see the resulting window after the tuning.



Response Time (seconds): 530.6

Transient Behavior: 0.6

Controller Parameters: P = 0.0001753, I = 7.417e-07, D = -0.003009, N = 0.008001

Figure 11: Tuning of the PID controller in Simulink

We can see the course of the actual output (dashed line) that is quick but with relatively big overshoot. This PID controller offers two tuning parameters – the first one affects the speed of the response and the second one aggressiveness/robustness of the controller. PID tries to set better results that is why the solid line which represents tuned response much smoother course. We can change these values to obtain our desired course. Once we are satisfied with the response, we can press button "Update block" that sets computed values of parameters *P*, *I*, *D* and *N* in the appropriate windows in Figure 9.

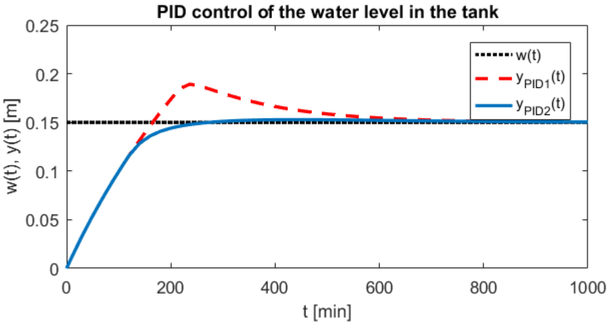


Figure 12: Course of the reference signal $w(t)$ and the output variables $y(t)$ in PID control

Now the trigger in the Manual switch could be set back to the branch with the model of the tank, and we can simulate the control on the mathematical model.

The graph in Figure 11 represents the course of the output variable from the identified linearized transfer function (14).

We have performed two simulations of the PID control. The first one (PID1) with the speed parameter 120 and robustness 0.7 in tuning window Figure 11 results in parameters:

$$P = 0.00097; I = 5.79 \cdot 10^{-6}; D = 0.0026; N = 0.017 \quad (18)$$

The result of control is then shown in Figure 12 and Figure 13. The second controller (PID2) has speed coefficient of 320 and robustness 0.8 that produces PID parameters:

$$P = 0.00034; I = 1.42 \cdot 10^{-6}; D = 0.0067; N = 0.011 \quad (19)$$

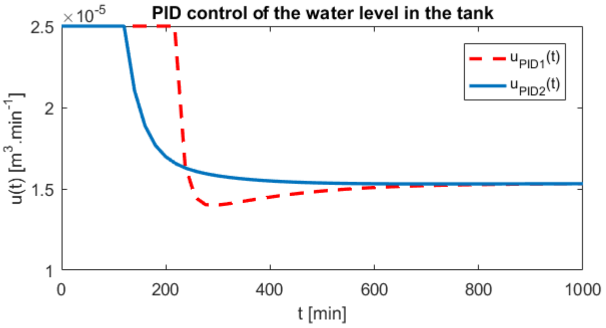


Figure 13: Course of the input variables $u(t)$ in PID control

Presented graphs have shown, that the first controller PID1 represented by the dashed line produces not very optimal results with overshoot 26%. But if we change parameters of the controller to the setting of the second controller PID2 (solid line), results are much better, the course of the output is much smoother without overshoot.

The course of the input variable $u(t)$ in Figure 13 shows smooth course this variable. We can also see limitation of the input variable to value $2.5 \cdot 10^{-5}$ $m^3.min^{-1}$ especially in the beginning of the control. This limitation was done by the block Saturation in Figure 10 and it was applied because of the physical limitation of the input volumetric flow rate in which has this value as a maximal limitation.

CONCLUSION

The contribution shows the advantages of computer simulation in the design of the controller. The paper presents the whole procedure from the design of the mathematical model through the simulation of the steady-state and dynamics of the system, identification and finally control of the system.

All simulations were done in Matlab, and its simulation tool Simulink and reader has an overview of the usage of these programs on the simulation of the mathematical model of the real system represented by the water tank. This system offers control of the water level in the tank by the volumetric flow rate of the input water flow.

The output response was identified by the first order transfer function parameters of which were identified by the minimization of the square of the difference of the simulated and identified output.

The PID continuous-time control was used for control of the water level in the tank. Simulink offers the block PID controller that can be easily implemented into the Simulink scheme. Moreover, this controller could be tuned by choice of the speed and the robustness of the control. There were shown results of two control simulations that demonstrate the usability of this PID settings.

The next step will be the verification of these controllers on the real model of the water tank.

ACKNOWLEDGMENT

This article was created with the support of the Ministry of Education of the Czech Republic under grant IGA reg. n. IGA/CebiaTech/2019/004.

REFERENCES

Armfield: Instruction manual PCT40, Issue 4, February 2005.

Gould, H., Tobochnik, J., Wolfgang, Ch.: An Introduction to Computer Simulation Methods: Applications To Physical Systems. 3rd Revised Version, 2017, CreateSpace Independent Publishing Platform

Graf, J. 2016. PID Control Fundamentals. CreateSpace Independent Publishing Platform.

Ingham, J., Dunn, I. J., Heinzle, E., Přenosil, J. E.: Chemical Engineering Dynamics. An Introduction to Modelling and Computer. Simulation. Second, Completely Revised Edition, VCH Verlagsgesellshaft, Weinheim, 2000.

Lagaris, J. C., Reads, J. A., Wright, M. H. and P. E. Wright. 1998. "Convergence properties of the Nelder-Mead simplex method in low dimensions", SIAM Journal of Optimization 9, 112-147.

Maria, A.: Introduction to modeling and simulation. In: Proceedings of the 1997 Winter Simulation Conference. 7-13.

Mathews, J. H., Fink, K. K.: Numerical Methods Using Matlab. Prentice-Hall 2004.

MathWorks. 2019a. Oficial website of MathWorks, Matlab producer. Available at < https://www.mathworks.com/>

MathWorks. 2019b. System Identification Toolbox documentation. Available at < https://www.mathworks.com/help/ident >

MathWorks. 2019c. Control System Toolbox documentation. Available at < https://www.mathworks.com/help/ident >

MathWorks. 2019d. Fminsearch documentation. Available at <https://www.mathworks.com/help/matlab/ref/fminsearch.html>

MathWorks. 2019e. PID controller documentation. Available at <https://www.mathworks.com/help/simulink/slref/pidcontroller.html >

Vojtesek, J., Dostal, P.: Adaptive Control of Water Level in Real Model of Water Tank. In 20th International conference on Process control ´15. Piscataway: IEEE Operations Center, 2015, s. 308 – 313

AUTHOR BIOGRAPHIES

JIRI VOJTESEK was born in Zlin, Czech Republic. He studied at Tomas Bata University in Zlin, Czech Republic, where he received his M.Sc. degree in Automation and control in 2002. In 2007 he obtained Ph.D. degree in Technical cybernetics at Tomas Bata University in Zlin. In the year 2015 he became an associate professor. His research interests are modeling and simulation of continuous-time chemical processes, polynomial methods, optimal, adaptive and nonlinear control. You can contact him on e-mail address vojtesek@utb.cz.

LUBOS SPACEK studied at the Tomas Bata University in Zlín, Czech Republic, where he obtained his master's degree in Automatic Control and Informatics in 2016. He currently attends PhD study at the Department of Process Control. His e-mail address is lspacek@utb.cz.