Control of Real Distributed Parameter Systems Modeled by COMSOL Multiphysics®

C. Belavý[*], G. Hulkó, S. Lipár and B. Barbolyas
Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava
*Corresponding author: Nám. Slobody 17, 812 31 Bratislava, Slovak Republic, e-mail: cyril.belavy@stuba.sk

Abstract: In the paper, first a basic concept of the engineering approach for modeling and control of distributed parameter systems based on interpretation of controlled systems as lumped input and distributed parameter output systems is introduced. Next, finite element method modeling of temperature fields in casting mould and extruder body as two various real distributed parameter systems by means of the software environment COMSOL Multiphysics is demonstrated. Obtained characteristics either in stationary or transient form are exported to the MATLAB & Simulink, where lumped and distributed models for control synthesis purpose are created. By means of Distributed Parameter Systems Blockset for MATLAB & Simulink, feedback distributed parameter systems of control are arranged. Finally, after simulation, real-time control of temperature field in casting mould is realized and for extruder body also block scheme for co-simulation joining COMSOL Multiphysics and MATLAB & Simulink software environment is prepared and control process is executed.

Keywords: distributed parameter systems, partial differential equations, finite element method, modeling, control.

1. Introduction

Many technical and non-technical systems and processes in the practice have the dynamics, which depends on both position and time. Such systems are classified as distributed parameter systems (DPS). The time-space coupled nature of the DPS can be mathematically described by partial differential equations (PDE) with boundary and initial conditions. For the last period we have witnessed a big boom using methods for numerical modeling of a wide variety such systems in the software environment COMSOL Multiphysics. Resulting models obtained through finite element method (FEM) solution of PDE give possibilities not only for analyzing the dynamics, but also for optimization and control of these systems as DPS. For advanced analysis and design of control is also co-simulation joining COMSOL Multiphysics and MATLAB & Simulink software environment at disposal. Base conception of FEM modeling and control is presented for two various real DPS: casting mould and extruder body.

2. LDS representation of DPS

In general, DPS are systems whose state or output variables $X(x,y,z,t)/Y(x,y,z,t)$ are distributed variables, or fields of variables, where (x,y,z) are spatial coordinates in 3D. PDE describes of DPS as infinite-dimensional system. However, from point of view of implementation of DPS control in technological practice, where a finite number of sensors and actuators for practical sensing and control is at disposal, such infinite-dimensional systems need to be approximated by finite-dimensional systems. There are many dimension reduction methods, which can be used to solve this problem, e.g. DPS very frequently are found in various technical and non-technical branches in the form of lumped-input/distributed-output system (LDS), [1], [2], see Fig. 1.
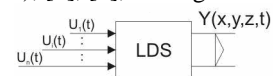

Figure 1. Lumped-input/distributed-output system representation of distributed parameter systems.

2.1 Dynamics of LDS

Output of the linear LDS in time domain, or in s-domain is in the form:

$$Y(\overline{x},t) = \sum_{i=1}^{n} Y_i(\overline{x},t) = \sum_{i=1}^{n} G_i(\overline{x},t) \otimes U_i(t) \quad (1)$$

$$Y(\overline{x},s) = \sum_{i=1}^{n} Y_i(\overline{x},s) = \sum_{i=1}^{n} S_i(\overline{x},s) U_i(s) \quad (2)$$

where $\overline{x} = (x, y, z)$ is position vector in 3D, $U_i(t)$ - lumped input quantity, $\mathcal{G}_i(\overline{x}, t)$ - i-th distributed impulse response, $\otimes$ - denotes convolution product, $Y_i(\overline{x}, t)$ - system response to the i-th input, $S_i(\overline{x}, s)$ - i-th transfer function. When $U_i(t)$ is a unit-step (Heaviside) function, $Y_i(\overline{x}, t)$ is in the form of distributed transient response function $\mathcal{H}_i(\overline{x}, t)$.

For a discrete-time system considering zero-order hold (ZOH) units, the overall distributed output variable of LDS and ZOH can be expressed in the form:

$$Y(\overline{x}, k) = \sum_{i=1}^{n} Y_i(\overline{x}, k) = \sum_{i=1}^{n} \mathcal{G}H_i(\overline{x}, k) \oplus U_i(k) \quad (3)$$

where $\oplus$ denotes convolution sum. For points $\overline{x}_i = (x_i, y_i, z_i)$ located in surroundings of lumped input variables, $U_i(t)$, where partial distributed transient responses $\mathcal{H}_i(\overline{x}_i, t)$ attain maximal amplitudes, partial particular distributed output variables are obtained in time-domain, or in z-domain respectively:

$$\left\{ Y_i(\overline{x}_i, k) = \mathcal{G}H_i(\overline{x}_i, k) \oplus U_i(k) \right\}_{i=1,n} \quad (4)$$

$$\left\{ Y_i(\overline{x}_i, z) = SH_i(\overline{x}_i, z) U_i(z) \right\}_{i=1,n} \quad (5)$$

For the space dependency and in steady-state we can define reduced transient step responses between i-th input variable at point $\overline{x}_i = (x_i, y_i, z_i)$ and corresponding particular distributed output variable in steady-state:

$$\left\{ \mathcal{H}HR_i(\overline{x}, \infty) = \frac{\mathcal{H}H_i(\overline{x}, \infty)}{\mathcal{H}H_i(\overline{x}_i, \infty)} \right\}_{i=1,n} \quad (6)$$

Dynamics of LDS is decomposed to time and space components. In the time dependency, there are, discrete transfer functions of the form $\left\{ SH_i(\overline{x}_i, z) \right\}_{i=1,n}$ and in the space direction there are $\left\{ \mathcal{H}HR_i(\overline{x}, \infty) \right\}_{i=1,n}$.

2.2 Feedback control loop based on LDS

Structure of DPS feedback control loop based on LDS is on Fig. 2, where the goal of control is to ensure the steady-state control error to be minimal:

$$\min \left\| \overline{E}(x, \infty) \right\| = \min \left\| W(\overline{x}, \infty) - Y(\overline{x}, \infty) \right\| \quad (7)$$
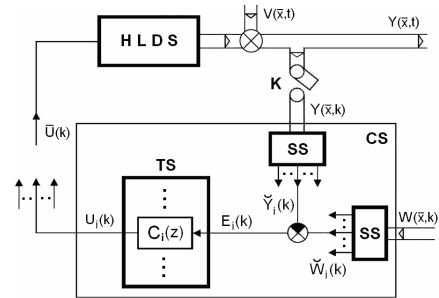

Figure 2. Distributed parameter feedback control loop: HLDS - LDS with zero-order holds $\{H_i\}_i$ on the input, CS - control synthesis, TS - control synthesis in time domain, SS - control synthesis in space domain, K - time/space sampling, $V(\overline{x}, t)$ - disturbance variable, $Y(\overline{x}, t)$ - distributed controlled variable, $Y(\overline{x}, k)$ - sampled distributed controlled variable, $\left\{ \overline{Y}_i(k) \right\}_i$ - approximation of controlled variable, $W(\overline{x}, k)$ - reference variable, $\left\{ \breve{W}_i(k) \right\}_i$ - approximation of reference variable, $\left\{ E_i(k) \right\}_i$ - control errors, $\left\{ C_i(z) \right\}_i$ - lumped parameter controllers, $\left\{ U_i(k) \right\}_i$ - lumped control variables, $\overline{U}(k)$ - vector of lumped control variables.

First, in the SS blocks, the approximation both of sampled distributed controlled variable $Y(\overline{x}, k)$ and reference variable $W(\overline{x}, \infty)$, on the set of reduced steady-state distributed step responses $\left\{ \mathcal{H}HR_i(\overline{x}, \infty) \right\}_i$, are solved in following form:

$$\min_{\overline{Y}_i} \left\| Y(\overline{x}, k) - \sum_{i=1}^{n} Y_i(\overline{x}_i, k) \mathcal{H}HR_i(\overline{x}, \infty) \right\| =$$
$$= \left\| Y(\overline{x}, k) - \sum_{i=1}^{n} \breve{Y}_i(k) \mathcal{H}HR_i(\overline{x}, \infty) \right\| \quad (8)$$

$$\min_{\breve{W}_i} \left\| W(\overline{x}, \infty) - \sum_{i=1}^{n} W_i(\overline{x}_i, \infty) \mathcal{H}HR_i(\overline{x}, \infty) \right\| =$$
$$= \left\| W(\overline{x}, \infty) - \sum_{i=1}^{n} \breve{W}_i(k) \mathcal{H}HR_i(\overline{x}, \infty) \right\| \quad (9)$$

Next, based on the solution of approximation problem, the vector of control error is created:

$$\bar{E}(k)=\left\{E_i(k)\right\}_i=\left\{\breve{W}_i(k)-\breve{Y}_i(k)\right\}_i \qquad (10)$$

The control errors vector $\bar{E}(k)=\left\{E_i(k)\right\}_i$ enters into the block TS, where the vector of control variables, $\bar{U}(k)=\left\{U_i(k)\right\}_i$ is generated by controllers $\left\{C_i(z)\right\}_i$ in single-parameter control loops. During the control process, for $k\rightarrow\infty$ the control task (7) is accomplished.

3. Modeling and control of temperature fields in casting mould

The casting mould is one of the key components of a casting. It is well known, that the quality of the castings is affected strongly by the distribution of temperature in the mould, which has both time and space dependence. For study of the physical phenomena occurring during casting solidification of aluminium-alloy in the permanent mould, from a DPS control point of view, control system development as well as mathematical model validation, a benchmark of the casting processes with a two-part steel mould of a complex-shape has been made, Fig. 3. Inside of the mould are built-in 26 electric heating elements in the cylinder shape, each with maximal heating power 400 W. Heating elements are grouped to 5 zones and their heating power is actuated by the input voltage range of (0–10) V. In the body of the mould is also placed 7 water-cooled copper chills and 11 thermocouples. Location of these elements has been carefully designed in order to have the possibility of preheating the mould achieving desired temperature profile.
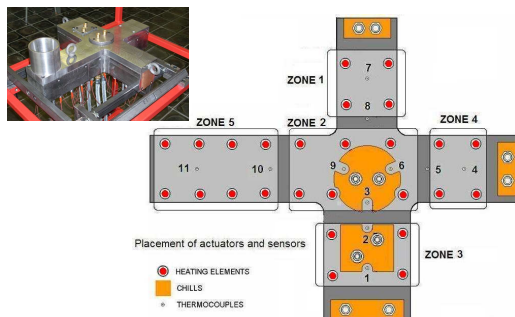

Figure 3. Benchmark casting plant and bottom side of the steel casting mould.

3.1 Modeling of temperature fields of casting mould in COMSOL Multiphysics

Temperature field of the mould in the casting process is a typical case of DPS. For FEM modeling of temperature fields as DPS, wide possibilities are offered by COMSOL Multiphysics with *Heat Transfer Module*.

Distribution of temperatures $T(\bar{x},t)$ in the casting mould over the definition domain $\Omega\in E_3$ for lumped input variables $\left\{U_i(\bar{x},t)\right\}_{i=1,5}$ in the form of heat sources $Q_i\,(\text{W/m}^3)$ is modeled by PDE of parabolic type with initial condition and boundary conditions for heat flux:

$$\frac{\partial T(\bar{x},t)}{\partial t}-a\nabla^2 T(\bar{x},t)=\sum_{i=1}^{5}U_i(\bar{x},t) \qquad (11)$$

$$-n(-\lambda\nabla T)=h(T_{ext}-T)$$

$$T(\bar{x},0)=T_{init}$$

where $a=\lambda/\rho c$ is temperature conductivity $(\text{m}^2\cdot\text{s}^{-1})$, h is heat transfer coefficient $(\text{W}\cdot\text{m}^{-2}\cdot\text{K}^{-1})$, T_{ext} is external temperature and T_{init} is initial temperature.

3D geometry model of the mould was drawn in a CAD environment and imported into COMSOL Multiphysics, where all parameters of PDE and other necessary conditions for simulation and data processing were specified through graphical user interface (GUI).

There by means of the toolbar menu in the *Physics* menu subdomains of the mould with their parameters were defined. Parameters like thermal conductivity, density, heat capacity and heat source either for Iron or Copper were loaded from Library material. Heating elements in the cylinder shape, where heat sources as actuating quantities were defined, were grouped into 5 subdomains of the mould. Similarly were defined as groups copper chills, next bottom and upper part of the mould. Also in the Physics menu Boundary conditions were defined.

In the *Mesh* menu mesh for the given geometry was analysed, e.g. initial mesh, various predefined mesh sizes, etc. The *Mesh Statistics* dialog box contains statistical data for the current mesh.

In the *Solve* menu temperature fields of the mould either in stationary, or transient form were computed by FEM. Actuating of heating elements was performed separately in each zone.

Obtained temperature fields were analyzed by means of various forms of plots and their parameters, including AVI animation with help of the *Postprocessing* menu. Several results, e.g. for Zone #5 are on Fig. 4 and Fig. 5.

Temperature fields of the mould obtained by FEM solution in COMSOL Multiphysics were exported to MATLAB & Simulink, in order to determine lumped and distributed models. Transfer functions $\left\{ SH_i\left(\overline{x}_i, z\right) \right\}_{i=1,5}$ are used for control synthesis in time domain TS and reduced transient step responses in steady-state $\left\{ \mathscr{H}HR_i\left(\overline{x}, \infty\right) \right\}_{i=1,5}$ serves as basis functions for solution of approximation task in the control synthesis in space domain SS. Temperature fields were also analyzed at 11 points, where measuring points of thermocouples in the casting mould are located.
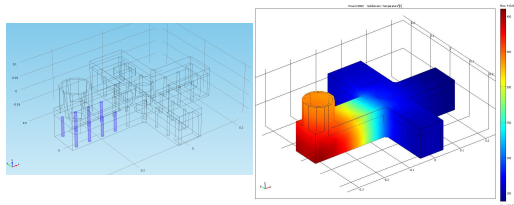
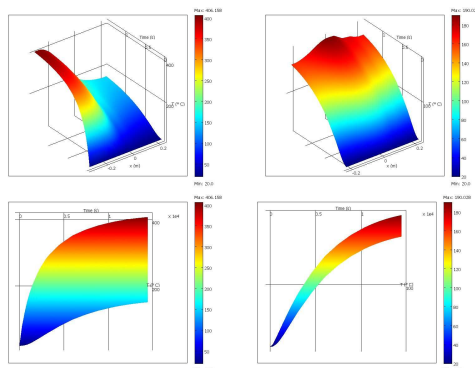**Figure 4.** Temperature fields in steady-state actuated by heating elements in Zone #5.

Figure 5. Temperature fields in transient form at z=0.05 m along x and y axis actuated by heating elements in Zone #5.

3.2 Control of temperature fields of casting mould

Control system for preheating of casting mould in the benchmark casting plant consist of five single parameter control loops, where discrete PID controllers are used. In the

MATLAB & Simulink environment by means of the DPS Blockset, [3] DPS control scheme was arranged, Fig. 6.
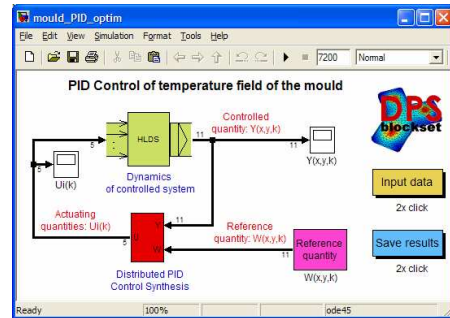

Figure 6. PID feedback control loop in DPS Blockset.

Synthesis of PID controllers there was adjusted in order to assure aperiodic running of the quadratic norm of distributed control error. Optimized controllers were then implemented to the control scheme for the real-time control of temperature fields of the casting mould. Results of the PID control process for given reference variable - temperature given in 11 positions, where thermocouples are embedded, are on Fig. 7. The performance of the control both in the time and space domain is given by the quadratic norm of the distributed control error.
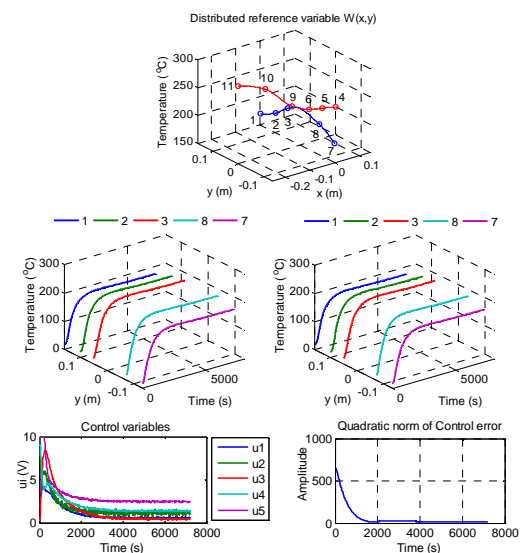

Figure 7. PID control of temperature field of the mould: distributed reference variable, controlled variables at given position of the mould, control variables $U_i(k)$ in zones 1-5 and quadratic norm of distributed control error.

4. Modeling and control of temperature fields in extruder body

Extrusion is the most common plastics and rubber processing technology. It is being used to pre-process the plastic mixture for subsequent operations (e.g. injection moulding and blow moulding) as well as for the production of finished profiles. The corresponding device is called an *extruder*. Several configurations are known, most commonly single and twin-screw extruders with various modifications and enhancements, [4].

The barrel temperature field has a major influence on the product quality as well as the process itself. To get a closer insight to the problem, an experimental rig was built, Fig. 8.
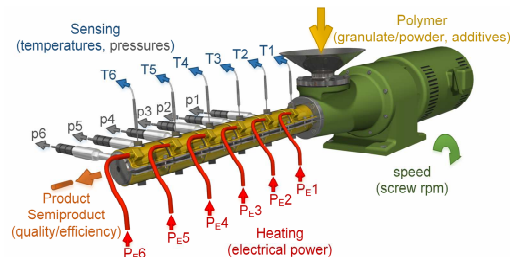

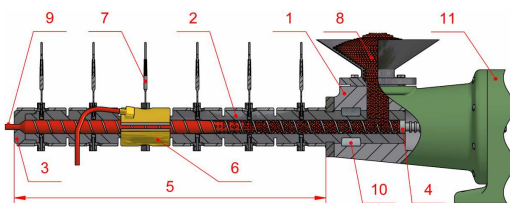**Figure 8**. Extruder assembly

Figure 9. Cross-section of the extruder: 1 - hopper, 2 - barrel, 3 - die, 4 - screw, 5 - heated part, 6 - heater band, 7 - temperature probe, 8 - material feed, 9 - extruded profile, 10 - cooling channel, 11 - screw drive motor.

The barrel's heated part, see Fig. 9, is subdivided into 6 zones, each having a heater band and temperature probe in order to control the temperature profile, which depends on the particular material being processed as well as the desired extruded profile and/or subsequent operations, [5]. Before starting screw rotation, the barrel has to be preheated to the desired temperature profile. Insufficient preheating would cause the material not melting where it is supposed to melt, consequent immense pressure

rise, which eventually results in screw seizing. Higher temperatures on the other hand may be beneficial in order to facilitate material flow in the barrel and lowering production costs, however though, overheating results in too soft output material, or worse, rendering it useless due to irreversible heat induced degradation. The cooling channel serves the protection of the screw drive preventing it from overheating.

4.1 Modeling of temperature fields of extruder in COMSOL Multiphysics

As stated, the barrel temperature field is created by heater bands and monitored using temperature probes. The barrel-plastic-screw system is governed by a nonlinear PDE. Modeling of this process was done by FEM using COMSOL Multiphysics and the model validated on the laboratory rig.

We developed a simplified model of the extrusion process focused on the relation between the heater inputs and barrel temperature field, Fig. 10. Since the model is used to estimate only temperatures and heat flux, the screw rotation and rheology of the extruded plastic are not considered.
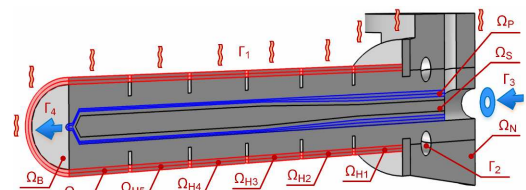

Figure 10. The computational model comprises the domains Ω_B – the barrel, Ω_P – the plastic, Ω_S – the screw, Ω_I – the hopper, and $\Omega_{H1...6}$ – the heater bands; further the boundaries: Γ_1 – the outer surface, Γ_2 – the cooling channel, Γ_3 – cold plastic inlet, and Γ_4 – the hot plastic outlet.

The PDE governing this system is

$$\rho c \left(\frac{\partial T}{\partial t} + \nabla T \, v \right) - \nabla \left(\lambda \nabla T \right) = \dot{q} \qquad (12)$$

where T (K) is the temperature, $\dot{q}$ (W) heat sources/sinks, ρ (kg·m^{-3}) density, c (J·kg^{-1}·K^{-1}) specific heat, λ (W·m^{-1}·K^{-1}) thermal conductivity, v (m·s^{-1}) flow velocity.

Heater bands are six nonlinear domains serving the heat input generation. This is generated in the heater bands by electric

resistance heating, which is controlled by thyristors for each channel separately. Joule's law states:

$$Q_{el} = R(T)I^2 t = P_{el} t \qquad (13)$$

where R (Ω) is the heaters resistance, I (A) current, P_{el} (W) electric power.

The heat generated by the heating element permeates into the barrel, wherein is led and passes it to the inflow part or to cooling water and to plastic.

The extruder outer surface is subject to free air cooling.

$$\dot{q}(x,t) = h.(T(x,t) - T_{ext}), \quad x \in \Gamma_1 \qquad (14)$$

where $x = (x, y, z)$ is the spatial coordinate, h $(W \cdot m^{-2} \cdot K^{-1})$ is the convection heat transfer coefficient, T_{ext} is temperature of the surrounding air. For the operating temperature range the effect of radiation may be neglected, or better, can be easily linearized by including it in the convection heat transfer coefficient.

Cooling is defined as a simplified law of cooling:

$$\dot{q}(x,t) = h.(T(x,t) - T_{CW}), \quad x \in \Gamma_2 \qquad (15)$$

where T_{CW} is the temperature of cooling water. If $h = 0 \Rightarrow \dot{q} = 0$, in such case is cooling off.

Plastic inlet Γ_3 is computed from defined mass flow rate Q_m $(kg.s^{-1})$:

$$v(x,t) = \frac{Q_m}{(\rho(x,T).S_{\Gamma 3})} \quad x \in \Gamma_3 \qquad (16)$$

where S_{Γ_3} (m^2) is the inlet cross-section surface.

The mass conservation law requires the in and out-flow rate being equal, therefore the flow velocity is changing with fluid density throughout the domain. So the plastic outlet boundary condition is as follows:

$$v(x,t) = \frac{Q_m}{(\rho(x,T).S_{\Gamma 4})} \quad x \in \Gamma_4 \qquad (17)$$

where S_{Γ_4} (m^2) is the outlet cross-section surface.

For solution of a simplified extrusion process was used module *Conjugate Heat Transfer*, which allows to solve tasks of heat transfer together with fluid flow.

4.2 COMSOL & MATLAB LiveLink for co-simulation and control

LiveLink provides an interface for transferring data between programs. The advantage of such a link programs MATLAB and COMSOL is the possibility of creating co-simulation, Fig. 11. Co-simulation is the current running programs MATLAB & Simulink and COMSOL Multiphysics, which provides the option of setting up, testing and application management, possibly arbitrary algorithms directly to a nonlinear finite element model of the extruder in the program COMSOL.

Co-simulation algorithm is implemented as Interpreted MATLAB Function in the program MATLAB & Simulink. This function performs simulation with finite element model extruder program COMSOL, based on information in the scheme. Information Scheme Simulink with links through to block co-simulation model in COMSOL program that performs the necessary calculations and the results of these calculations come to the output of the block co-simulation.

Using co-simulation, we obtain in MATLAB & Simulink nonlinear model of the extruder with possibility its simulation activity with any setting performance heating elements with smooth starting and stopping the flow of plastic and cooling, possibly with varying ambient temperatures and, for example, the density of granules. The number of input variables can be expanded, e.g. other plastic material parameters. Created to block co-simulation is of course also possible to connect the controller, optionally apply additional algorithms and test them.
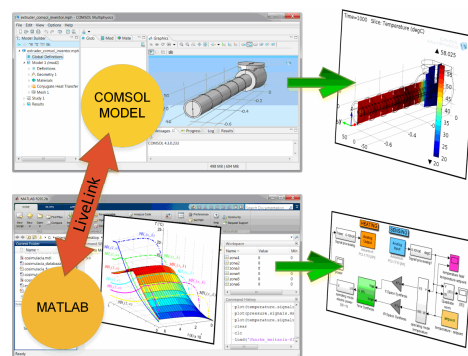

Figure 11. Modeling and control of temperature field of the extruder body by means of co-simulation joining COMSOL Multiphysics and MATLAB & Simulink software environment.

Co-simulation scheme for analysis of influences of various input variables to the extruder barrels temperature is on Fig. 12. Results of real-time control of temperatures in extruder as DPS, executed according scheme on Fig. 13 with PSD controllers, are on Fig. 14.
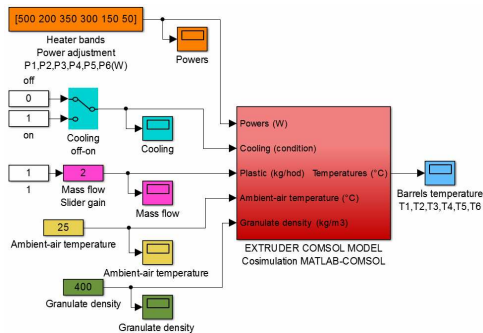

Figure 12. Scheme with co-simulation block for actuation and sensing of model parameters extruder.
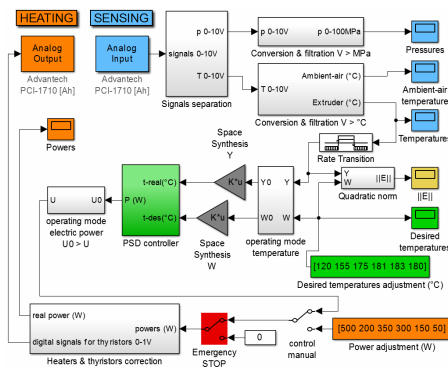

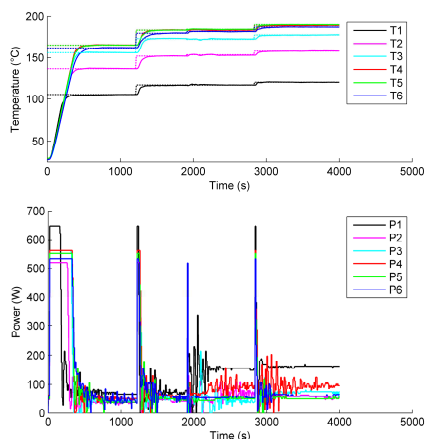**Figure 13**. Scheme for DPS feedback control of temperatures in extruder body.

Figure 14. Real-time control of temperatures in extruder body.

5. Conclusions

COMSOL Multiphysics is a powerful interactive software environment for modeling and dynamical analysis all kinds of scientific and engineering problems based on PDE. In this way COMSOL Multiphysics opens new possibilities also for control of distributed parameter systems described by numerical structures along with Distributed Parameter Systems Blockset for MATLAB & Simulink in engineering practice.

Presented methodical approach of FEM modeling and design of control of DPS based on LDS approach is suitable for various real technological processes and may be an important tool for reduce manufacturing costs, shorten lead times and improve of the quality of resulting products.

6. References

1. G. Hulkó, et al., *Modeling, Control and Design of Distributed Parameter Systems with Demonstrations in MATLAB*, Publishing House of STU Bratislava, monograph (1998)

2. G. Hulkó, et al., Engineering methods and software support for modelling and design of discrete-time control of distributed parameter systems, *European journal of control,* vol. 15, No. 3–4, pp 407-417, ISSN 0947-3580 (2009)

3. G. Hulkó, et al., *Distributed Parameter Systems Blockset for MATLAB & Simulink*. Third-Party MathWorks product. Available: www.mathworks.com/products/connections/ (2003-2012)

4. C. Rauvendaal, *Polymer extrusion, 4th edition*, Hanser Verlag, München (2001)

5. F. G. Harold, J.R. Wagner and E.M. Mount, *Extrusion: The definitive processing guide and handbook*, Wiliam Andrew Inc., Norwich (2005)

7. Acknowledgements