

Modelling of the Glass Melting Process for Real-Time Implementation

Wojciech Grega, Adam Pilat, and Andrzej Tutaj

Abstract—Improvement of process efficiency and product quality is available through implementation of more complex control algorithms and more accurate process models. It is especially critical for the glass industry production chain since glass production is a complex processes with high energy usage. The accuracy and robustness of advanced control algorithms is strictly dependent on the quality of the underlying mathematical model of the production process.

This paper presents the formalisation and an empirical investigation of the hypothesis that a simplified, Finite Element Method (FEM) - based model can capture the closed-loop process dynamics over longer time scales and is suitable for real-time applications. The paper demonstrates how a trade-off between model complexity and simulation time can be found.

Index Terms—Real-time control, predictive control, Finite Element Method, molten glass model, model reduction.

I. INTRODUCTION

The use of models in control is motivated by the possibility of improving control by incorporating knowledge about the process into the controller design procedure. Modern control is essentially model-based, implying that the model is a unique interface between the theory and the real world. For example, observers and Kalman filters use a copy of a plant model for operation when implementing a state feedback. This method of controller implementation usually comprises a process model (linear, nonlinear) and a cost function. The more explicit use of the model in the control configuration is seen in Model Reference Control or Model Predictive Control [1]. These controllers are not only model-based but also model-driven, in the sense that the models are directly involved in the architecture of the control systems.

Various issues related to models and modelling in control have been extensively worked on over the last two decades. The model must capture the process dynamics with sufficient accuracy to ensure good control performance. For example, most of the physical processes introduce nonlinearity into the process description. A linear model can only be a local valid approximation for the dynamics around the specified operating point. A variety of techniques are available for combining linear and nonlinear modelling methods [2].

Control systems designers do not care very much about real-time implementations of the models [3]. In many cases they do not understand the control-timing constraints. The typical proposed solutions are: "buy a faster computer" or "install more efficient data transmission network". In fact, the Control Theory does not provide advice on how to design controllers to overcome the limitations of real data processing systems.

Real time implementation of the models cannot be discussed separately from the broader issue of how to use them in an industrial environment. Most of the industrial control systems adopt a multilevel, vertical control hierarchy. Typically [4], the system is structured into three control levels (Fig. 1) which are: the direct (device) control level, supervisory level and management level.

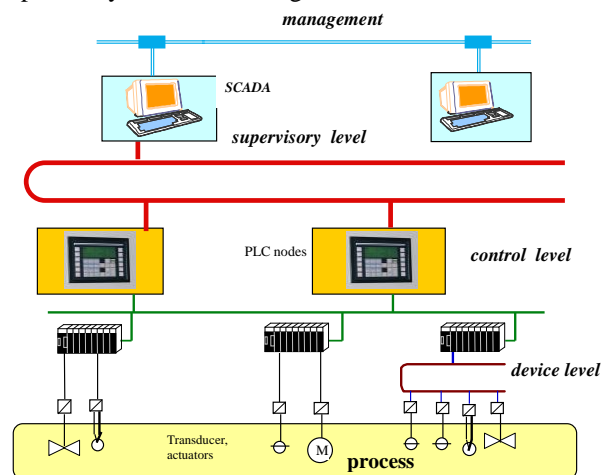


Fig. 1. Multilevel structure of the industrial control system.

The basic task of the direct (device) control level is to maintain the process states at the prescribed set values. The device controller level provides an interface to the hardware, either as separate modules or as microprocessors incorporated in the equipment to be controlled. Here, mainly PID digital control algorithms are implemented in real time. High speed networks and fieldbuses are implemented to exchange in real time the information between the front-ends and the device controllers and, vertically, with the supervisory control level. This architecture has the advantage of locating the hard real-time activities as near as possible to the equipment.

Classically, the supervisory level calculates set points for the controllers according to the defined criterion (Fig. 2).

For this purpose more complex mathematical models of the process are employed at this level to find the optimal set-points or trajectories for direct control devices. Industrial use of advanced control algorithms (predictive control,

Manuscript received September 20, 2015; revised December 1, 2015. This work was supported by the Polish National Grant PBS2/A3/16/2013.

The authors are with the AGH University of Science and Technology in Krakow, 30-059 Krakow, Mickiewicza Av. 30, Poland (e-mail: wgr@agh.edu.pl, ap@agh.edu.pl, tutaj@agh.edu.pl).

repetitive control) is growing slowly but surely at the supervisory level.

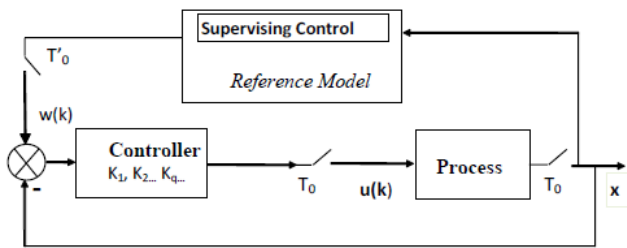


Fig. 2. Model-based digital supervisory control: desired trajectory $w(k)$ is delivered by the upper level. Time scale separation: $T_0 < T'_0$.

A vertical control hierarchy implements time scale separation [5]. Timescale separation is a practical necessity when considering the objectives of control tasks at different levels. With a reasonable time scale separation between the levels (typically by a factor of 5-10 in terms of closed-loop response time), we can observe the following advantages:

- The stability and performance of the lower (faster) level is not very much influenced by the presence of the upper (slower) control level;
- The performance of the upper (slower) level does not depend that much on the specific controller settings used in the lower layer;
- Allow for 'slow' (but more accurate) implementation of models at the supervisory level.

Technically, this control structure is represented by a number of data processing devices performing real-time tasks, coordinated by real-time operating systems. In most control applications in which the majority of tasks is periodic and the execution times have known upper bounds, it is possible to schedule tasks statically, based on explicit computer time allocation.

Activation of more complex algorithms requires an extension of the 'time slot' required for this task. This could be obtained by increasing the basic time step (T_0) or dividing the supervisory applications between the subsequent tasks. As a result of the first solution, a degradation of the control performance could be expected at the direct control level. The second results in an increase of the reference model application period (T'_0). This leads to the well-known conclusion: the real-time characteristics of controllers must be compatible with the timing constraints of the applications. The problem of selecting the control task frequencies to optimise the control system performance subject to real-time constraints has been addressed by several authors [3], [6]. However, its relations to the model complexity are rarely formulated and discussed in the current literature.

Model reduction is a typical approach implemented to derive cost-efficient representations of complex systems for control applications. It is important to assess the impact of model parameters on the quality of the solution and identify trade-offs between model complexity and real-time constraints. This is the main problem discussed in this paper. The proposed approach is illustrated using FEM modelling of the bottle manufacturing process.

II. PROCESS DESCRIPTION AND BASIC CONTROL METHODS

The bottle manufacturing process can be divided into four main parts: furnace, refiner, forehearths and forming machine (Fig. 3). Raw materials are continuously introduced into the furnace that delivers molten glass at approximately 1300 °C into the refiner where the glass is refined. Then, the molten glass flows to the forehearths towards the forming machines. At the outlet a plunger pushes portions of the glass through a ring and automated scissors cut the 'drops' of glass loose from the ring. The resulting drop falls into a machine that forms a product. The above processes are integrated. Poor quality control in one area may not have an immediate noticeable effect but could well cause problems and incur costs further down the line.

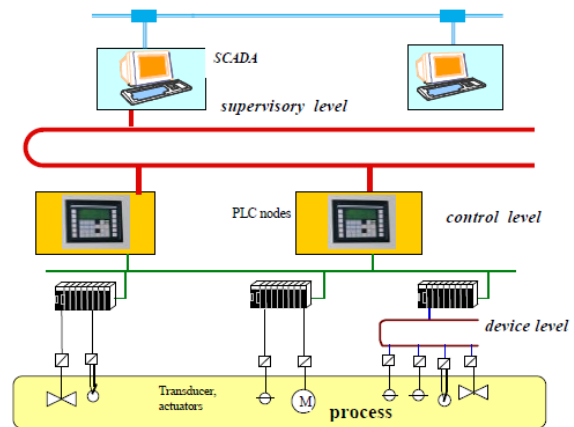


Fig. 3. Bottle manufacturing process.

The forehearth is the end part of a glass melting furnace. After melting and conditioning in a refiner, the glass enters the forehearth whose main purpose is to stabilise the output temperature. This temperature depends on the type of product that is being manufactured and must be maintained at a constant value. So, the task of the forehearth is to provide a consistent weight of glass, at a precise temperature and viscosity, to the forming machine.

In the forming process the time evolution and spatial distribution of the glass temperature are crucial since the glass viscosity depends greatly on the temperature. The main disturbances acting on the forehearths result from temperature variations in the furnace and glass flow changes. Controlling the glass temperature is key to achieving optimum glass viscosity and drop weight. Temperature variations as slight as one or two degrees can negatively impact the quality of the finished product and result in lost products.

Another challenge is the change in the type of bottle to be produced (referred to as a 'Job change'). A part of the outlet is changed, the pull rate becomes zero and the glass stops moving for approximately 20-30 minutes. As a result, the steady state of the glass furnace that feeds the feeder is disturbed. At this time no hot glass flows to the forehearths and the temperature decreases. Next, the outlet modification is finished and the glass melt is flowing freely (no cutting of drops) until the right glass melt temperature and chemical properties are reached. Then, the automated scissors start cutting again. It is important to reach the right glass temperature to form the new product as quickly as possible.

This situation can require several hours of special attention by the control system engineers to adjust the PID controller parameters in order to stabilise the process. After a Job Change, the glass temperature can typically take between 4 to 6 hours to stabilise at the set point and enable the forming machine to produce the bottles of an acceptable quality.

III. FORE HEARTH CONTROL

In Fig. 3 a forehearth example is shown with four zones. A typical forehearth control system is simple [7]. In each zone the glass temperature is controlled by an autonomous PI-controller. The output of temperature sensors in each zone is transmitted to a controller, usually to a single loop PI controller or a soft controller emulated by a SCADA system. PID controllers adjust the heating/cooling control actuators to bring the glass temperature back to a set point. The controllers are connected to a graphical operator interface via a proprietary network for data acquisition and off-line analysis. The matrix of 9 temperature sensors located at the end of the forehearth zone number 4 monitors the thermal homogeneity of the glass. There is no direct connection between the control loops in the various zones.

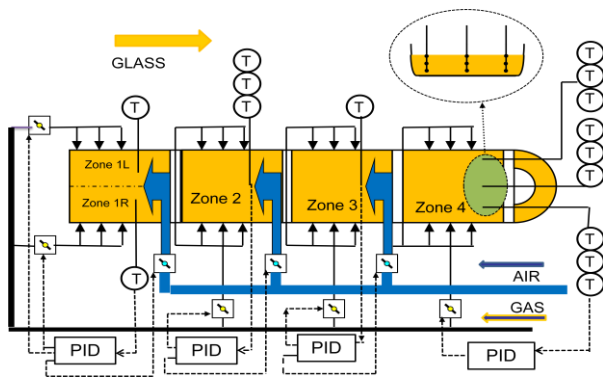


Fig. 4. A typical forehearth control system.

Theoretically, the operator only has to set the required temperature at the end of the forehearth channel and the control system should attend to everything else. In practice, it turns out that control of the molten glass thermal homogeneity is a difficult problem due to:

- The response time of the temperature control loop is quite long;
- The thermal and mechanical properties of the glass change with temperature producing nonlinear dynamics;
- The combined heating/cooling control actuators are nonlinear.

For this reason, operator action is necessary to determine the set points for each of the control zones. As the relationship between these values may significantly influence the thermal homogeneity of the glass at the end of the forehearth channel, his/her experience is an important operational factor.

IV. ADVANCED CONTROL OF THE FOREHEARTH

A large opportunity for improvement of the bottle manufacturing process efficiency and product quality is

available through implementation of more complex control algorithms and more accurate process models inside and around the configuration, as presented in Fig. 4. Because of the different time scales and levels of complexity, the operation of the forehearth control system implements the general structure from Fig. 1 and explores several hierarchical levels:

- 1) Process optimisation level: some process quality indices, such as energy consumption in steady-state, are formulated as an optimisation problem. The solution is glass melt temperature set points. It could also be automated by combining process FEM models and optimisation algorithms [8].
- 2) Predictive and multivariable process control [9]: A controller uses the process requirements from the optimiser level to calculate temperature set points or optimal trajectories for multiple input signals (e.g. a set of fuel flow rates of the burners) that result in optimal time dependent behaviour of multiple outputs (e.g. a set of bottom temperatures).
- 3) Direct control level: mostly PI or PID cascade controllers.

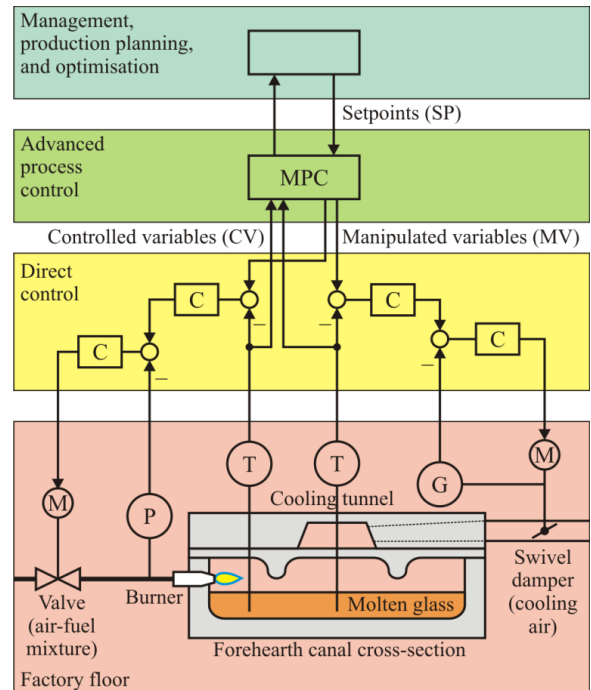


Fig. 5. Multilevel hierarchical structure of a forehearth glass conditioning process advanced control system: C – controller, M – servomotor, G – position sensor, P – pressure sensor, T – temperature sensor, MPC – Model Predictive Control.

For this purpose a multilevel hierarchical control system for forehearth zones has been considered. Its simplified structure is shown in Fig. 5. For a control algorithm at each particular level, all the levels beneath it can be considered as a generalised plant or control objects. The dynamical properties of such an equivalent plant differ considerably according to the location in the hierarchy. Hence, algorithms implemented at various levels require individual selection of the calculation time-steps.

At the direct control level, simple cascaded SISO feedback loops are employed. As described in the previous section, the forehearth is divided into several zones and each zone is equipped with independent controllers. Slave

controllers control the air-fuel mixture pressures as well as the cooling air dampers positions. Master controllers are engaged in molten glass temperatures loops. The dynamics of both mixture pressure and dampers position subsystems are relatively fast, hence a sampling period of the order of one second is justified. As uninterrupted control on this level is critical, PLC controllers running proprietary real-time operating systems are implemented. The dynamics governing temperature changes are considerably slow. It can be assessed based on Fig. 6 which presents the example temperature responses to abrupt changes of the air-fuel mixture pressure as recorded by a real industrial installation.

The forehearth comprises several cascaded zones with molten glass flowing along them. That structure implies strong cross-couplings and hence the Multi-Input, Multi-Output (MIMO) Model Predictive Control (MPC) method can bring considerable performance improvement compared to the classical SISO approach. It is especially the case during transients accompanying 'Job Change' operation (glass assortment changes).

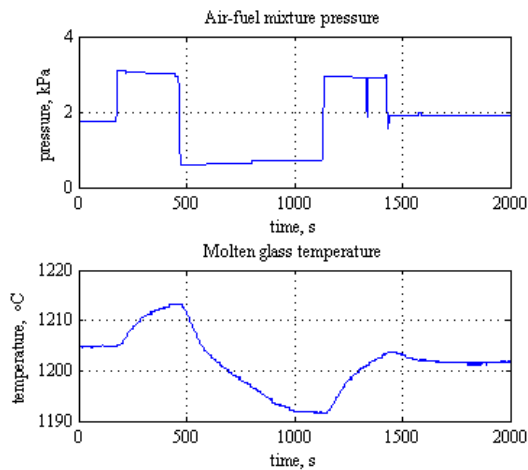


Fig. 6. Example of temperature changes in response to fuel-air mixture pressure changes for a single forehearth section.

A large number of industrial applications of the linear MPC algorithms confirm their practical usefulness [10]. Unlike direct control level techniques, the MPC methods incorporate a MIMO-type internal mathematical model of the plant. This model is used to predict the behaviour of a process in a discrete time setting: it starts from the current time instant by taking into account the current state, and it proceeds over a future prediction horizon. At each control interval the MPC algorithm attempts to optimise the future process behaviour by solving the open-loop optimal control problem on-line for an assumed prediction horizon and implementing it for one step. This delivers the optimal set-points or time trajectories in order to coordinate the operation of decentralised local controllers.

Most of the classical MPC algorithms used for prediction linear models, linear constraints and a linear quadratic objective result in a QP-problem, which can also be efficiently solved. However, when the process is nonlinear, a nonlinear model is a natural choice. Application of a nonlinear model for MPC calculation results in non-convex optimisation problems. This poses challenges for both MPC stability theory and numerical solution. During a single

MPC step, the optimisation algorithm seeks to find the optima; the solutions are then sent to the forehearth model, where the objective function value is calculated through simulation and returned back to the optimisation algorithm for the next time calculation. In real-time applications, when on-line nonlinear optimisation must be used, such iterations may be computationally demanding, especially when the dynamics of the process are fast [11]. It may turn out that the time necessary for calculation is longer than the assumed control interval. So, the computational requirements of the model execution are one of the challenges of data-based MPC.

With the configuration shown in Fig. 5, glass temperature measurements are selected as Controlled Variables (CV) for the MPC algorithm. Manipulated Variables (MV) from the MPC block are fed as set points to direct control level master cascaded controllers. Thus, from a MPC level perspective, the equivalent plant comprised both the forehearth and its cascaded SISO loops for all zones. The dynamics of this plant can be estimated based on a step response, as shown in Fig. 7. A simple linear first order transfer function equivalent model is characterised by a time constant of the order of tens of minutes. Hence, the MPC algorithm time step of minutes is recommended and one sufficient for supervisory forehearth control. Computations of the detailed Finite Element Method 3-D simulation models is very time-consuming: one steady state simulation of a complete forehearth can takes hours. The time demands of MPC algorithms indicate that the currently used detailed FEM simulation models are not suitable for model based Model Predictive Control of the considered process.

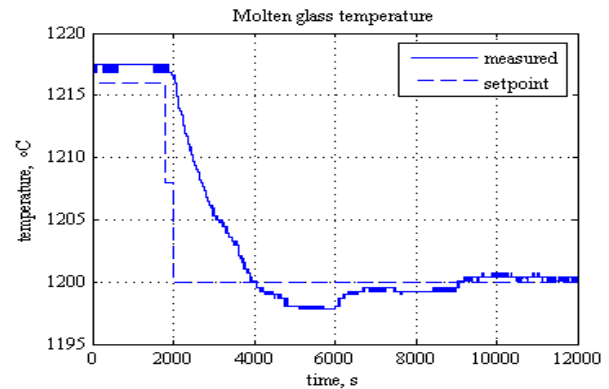


Fig. 7. Example of glass temperature response to step setpoint changes for a single forehearth section.

V. MODELS

For controlling and optimising industrial processes simulation models have become increasingly important over the last few decades. The models are very helpful for a number of problems that are particularly associated with the production of glass bottles: the optimisation of construction parameters, the investigation of the influence of single parameters on quality, the optimisation of the process efficiency (e.g. minimised energy use), the design of control strategies and real-time control. If we focus on the latter application, it can be observed that the accuracy and robustness of several control algorithms is entirely dependent on the quality of the underlying mathematical

model of the process. A model provides the controller with the dynamic behaviour of the process and determines the way by which the controller calculates process inputs (actuation signals) in order to stabilise the process, or to transfer the process from the current state to the desired state.

Basically, the forehearth mathematical model consists of two main parts (sub-models) describing [12]: (a) the glass flow process (Navier-Stokes Equation) and (b) the heat flow process in the glass (Heat Transfer Equation). The Partial Differential Equations (PDE) used for forehearth heat transfer modelling describe the distribution of temperatures in molten glass in the defined domain and are parabolic type with Neumann type boundary conditions [13]. Since the material parameters strongly depend on the temperature, both sub-models are strongly coupled. The sub-models are based on data for viscosity, density, thermal conductivity, heat capacity, absorption spectra and other relevant properties of the glass melt. The models are highly nonlinear due to the radiation and material parameters laws. An analytical solution of PDEs describing this process does not exist.

The literature [14] suggests that the Finite Element Method (FEM) gives the best results through discretisation of the glass flow equations and the heat transfer equations. Computational Fluid Dynamics (CFD) provides a qualitative (and sometimes even quantitative) prediction of fluid flows by means of mathematical modelling (PDEs). Numerical methods (FEM discretisation and solution techniques) and software tools support the calculations. CFD models are derived from a physical knowledge of the glass flow and environment. These models have the advantage that no special plant tests are needed to obtain the models. This approach does not replace the identification completely but the time of experimentation and the overall cost can be reduced. If a time dependent solver is used, this provides a possibility to study the process dynamics.

FEM models have a long tradition in the glass industry [12]. For example, very precise CFD models are used at the design phase of the bottle manufacturing process (furnaces, forehearths) or to better understand the phenomenon involved in the glass flow. However, their computation time is very lengthy and, in many cases, inadequate for real-time applications. Therefore, the CFD models used for the control application must take into account the limitations imposed by real-time constraints. It is expected that making the CFD models faster such that they can be used in real time will result in more advanced control applications in the glass industry.

During the discretisation of CFD and heat transfer models in space and in time the PDE systems are transformed into sets of algebraic equations. At this stage it is possible to optimise the model in terms of computation time. It is associated with:

- mesh generation (decomposition into cells/elements),
- space discretisation (approximation of spatial derivatives),
- time discretisation (approximation of temporal derivatives).

For modelling and analysis of the processes described by PDE, several possibilities are offered by COMSOL Multiphysics [15] - a specialised tool supporting numerical solutions of the sets of partial differential equations by FEM.

VI. IMPLEMENTING THE MODELS IN REAL-TIME

The operation of the controllers supporting the structure from Fig.5 can be split into three main tasks: read inputs (referred as 'sampling' at the factory floor level), control algorithm computation and exchange results ('actuation'). For advanced controllers, more tasks can be considered such as simulation (execution of the model tasks) and optimisation. The tasks are associated with the events (i.e. timer events) and are executed for each time step. The tasks usually share the same processor and exchange data with each other. Although a great variety of tasks scheduling policies is available [16], the periodic task scheduling method (fixed time step T_0) is the most popular in control applications.

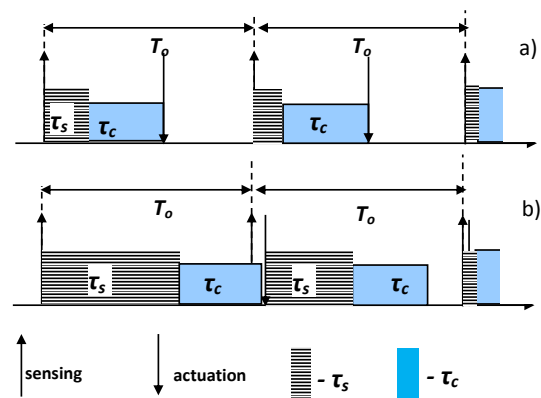


Fig. 8. Timing models that can be used for the periodic sampled process.

Fig. 8. illustrates typical timing of a controller that can be used for a regularly sampled process. Two tasks are considered: the simulation task (execution time of the model, i.e. the amount of time required to compute all equations and functions representing a process: execution time τ_s), and the control task i.e. implementing the simulation results (execution time: τ_c). The sensing and actuation are performed practically at the same sampling time (time delay can be neglected). The execution time of the simulation task depends on available computation power and the mathematical complexity of the model.

Figures 8a) and 8b) illustrate two cases. In a) the computing time is shorter than a fixed time-step, while in b) the computing time is longer. This is commonly known as an 'overrun'. Real-time constraints are not fulfilled for case b). For a given time-step, simulator operations results performed during the first step in Fig. 8b are lost - they cannot be implemented by the controller task in real time.

For a real-time simulation to be valid, the simulation task must accurately produce the internal variables and outputs of the simulation within the length of time no longer than its physical counterpart world [17].

To permit the real-time simulator to perform all the operations necessary to make a real-time simulation relevant, including execution of the model, we have two possibilities: (1) to extend the time step, (2) to reduce the simulation time. Solution 1) is limited by the control algorithm design principles: it is possible for the estimated dynamics of the process to define a maximal, admissible control time step, which would guarantee acceptable control performance. Solution 2) leads to a search for the simulation model

performing with an acceptable resemblance to its physical counterpart, without the occurrence of overruns.

VII. CASE STUDY

FEM simulation implemented on fine mesh grids allows for obtaining a very precise prediction of the glass flow process and heat flow in the glass, but at the cost of a large computation load. In order to implement an FEM model-based, real-time prediction control algorithm, the only solution is to use simpler but precise enough models.

Using COMSOL Multiphysics, the FEM forehearth model was solved for different parameters. After a number of numerical experiments it was found that in order to optimise the FEM model, the following parameters must be considered:

- model granularity, referred to as the number of degrees of freedom (DOF). COMSOL computes a solution at every node of the mesh. A tighter mesh results in a higher resolution but longer computing time;
- time step - a stationary model is solved quicker than a time dependent one but a time dependent problem is unavoidable as we wish to model the dynamics of the process;
- mesh shape. A model of the highest granularity and uniform mesh (but long execution time) was used as a benchmark (model M0). The temperature received in the steady-state (after 1000s) at the position of the matrix of 9 temperature sensors located at the end of the forehearth (zone number 4) was used to calculate the quality index in the form:

$$J^x = \sum_{i=1}^{i=9} (T_i^0 - T_i^x)^2$$

T_i^x - is the temperature simulated by the model x in the point i - number of temperature sensors location.

The second quality index shows how the reference temperature was reached for the different models. It also shows how the models simulate the operation of the local control loops:

$$J_{ref}^x = \sum_{i=1}^9 (T_{ref} - T_i^x)^2$$

T_{ref} - is the reference temperature at the end of the forehearth section number 4 (equal for each point),

The local controllers were tuned to receive a stable output of the process. However, the tuning of the local controllers was not optimal from the perspective of the supervisory coordinated control. Table 1 presents the results for the three models (M0, M1, M2, M3). The calculation time is shown for the time horizon of at 2600s.

TABLE I: COMPARISON OF MODELS

Model	M0	M1	M2	M3
Granularity (no. of DOF)	16 989	4 937	8 981	12 198
Quality index (J^x)	0	143.3	7 605	12631
Quality index (J_{ref}^x)	41	250	8412	13663
Calculation time [s]	462	39	342	195

Fig. 9 presents the grids implemented for each model. Fig. 10 shows the simulated profiles of temperature at the end of the forehearth. Fig. 11 shows examples of temperature transition simulated at the end of the forehearth (model M0).

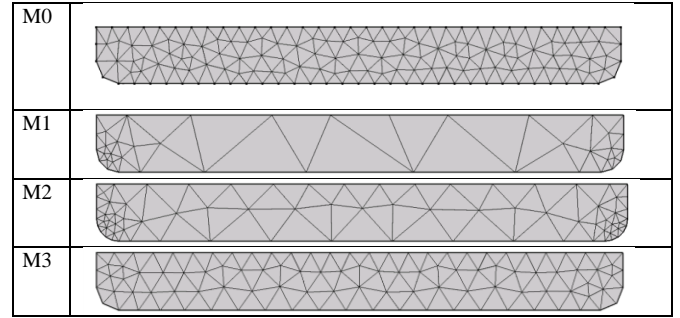


Fig. 9. FEM grids implemented for the models.

The first conclusion is that the quality of the model strongly depends on the mesh selection. This effect has been well known from some time [18]. One of the main concerns in finite element analysis is the optimisation of the FEM mesh. In our application, Model 1, with DOF 4 937, delivers quite a good performance as concerns quality indices and calculation time. Since the quality of the finite element approximated solutions directly depends on the quality of the meshes, an additional process to improve the quality of the meshes is necessary for better FEM approximations.

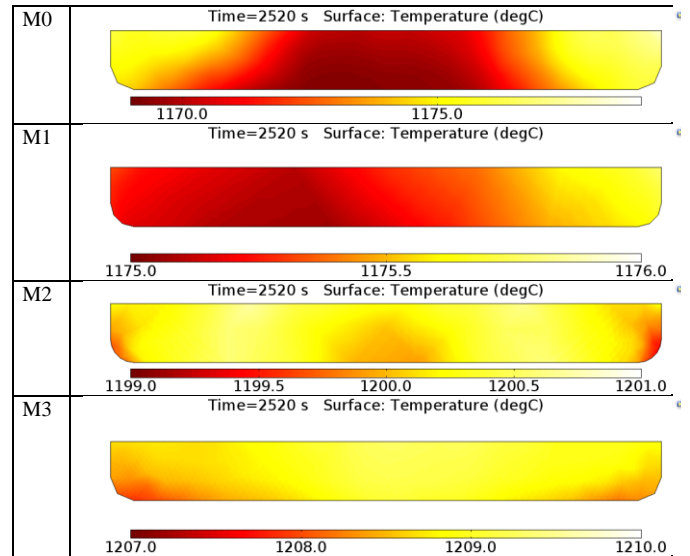


Fig. 10. Temperature profiles: steady state at the end of zone 4.

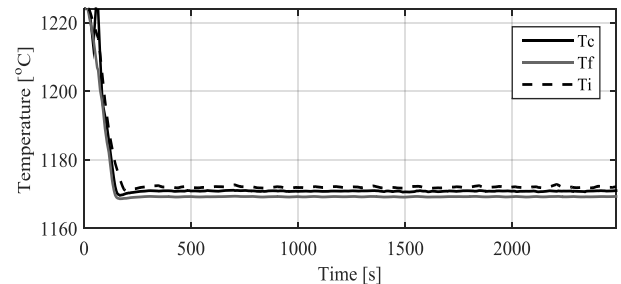


Fig. 11. Example of temperature transition at the end of zone 4, Tc, Tf, Ti - temperature sensor locations 4,5,6, respectively.

The next observation is that for the estimated dynamics of the considered forehearth process, model M1 can be implemented in real time. At the advanced process control

level, model M1 can be executed several times during a single step of the predictive controller. The estimated maximal admissible time step of the MPC controller is around 500s. It follows from the result of the simulation from Figure 11: the steady-state is reached after 400-450s. After this time new optimal setpoints can be delivered by the MPC to the decentralised local controllers. It should be noted that for the assumed 500s time horizon of the predictive controller, the calculation time will be proportionally shorter than those given in Table I.

Another observation related to the model was that the calculation time does not depend very strongly on the assumed internal time step of the analysed FEM models. The experiments were made for the steps from 5 up to 60 seconds of internal COMSOL calculation steps, and any major influence on the quality indices was noted.

VIII. CONCLUSIONS

Glass manufacturing processes have comparatively slow dynamics but are complex processes with high energy usage; especially, the forehearth parameters stabilizing process with multiple port gas burners cause the glass manufacturing industries to consume high amount of energy. Therefore, forehearth systems are facing great challenges in the reduction of natural gas consumption and emission by applying advanced control methods. The accuracy and robustness of an advanced control system is entirely dependent on the quality of the underlying mathematical model of the process and its environment.

Finite element methods become now be a useful tool for any pre-study in the field of automatic control. They allow for the building of complex models and perform the identification stage [19] to expanding knowledge about system functioning, dynamics and cross coupling. Moreover, it is possible to apply such a tool to develop a completely functional virtual prototype, which consists of physical phenomena, motion dynamics and control methods.

The main concept presented in this paper comes from two complementary design aspects. In model reference control, a more exact representation of a process provides better control performance. On the other hand, in a real-time environment a time interval assigned to the model solution task is limited. Therefore, a trade-off between model complexity and calculation time must be found. For FEM applications it can be done not only by the reduction of the number of DOF but also by optimisation of the grid.

In our solution concept we have proposed a real-time implementation of FEM for predictive supervisory control of the temperature in forehearth. A model reduction example has been developed in order to ensure a simulation time which is sufficient for real-time control.

IX. REFERENCES

- [1] P. Tatjewski, *Advanced Control of Industrial Processes, Structures and Algorithms*, London: Springer, 2007.
- [2] R. Kadiyala, "A toolbox for approximate linearization of nonlinear systems," *IEEE Control Systems*, vol. 13, pp. 47-57, April 1993.
- [3] W. Grega, "Performance evaluation of model-reference control," *7th IEEE International Conference on Methods and Models in Automation and Robotics*, pp. 407-412, Międzyzdroje, 2001.

- [4] W. Grega, "Robust networked control," *Challenges and Paradigms in Applied Robust Control*, pp. 349-372, 2011.
- [5] A. Brandao-Araujo, M. Baldea, S. Skogestad, and P. Daoutidis, "Time scale separation and the link between open-loop and closed-loop dynamics," in *Proc. the 16th European Symposium on Computer Aided Process Engineering*, 2006, pp. 1455-1460, Garmisch-Partenkirchen, Germany.
- [6] F. Xia and Y. Sun, "Control-scheduling co-design: A perspective on integrating control and computing," *Dynamics of Continuous, Discrete and Impulsive Systems - Series B*, vol. 13, no. S1, pp. 1352-1358, 2006.
- [7] J. McMinn, "Digital process control for a glass furnace and forehearth," *Glass International*, pp. 32-33, May/June 2005.
- [8] S. E. Sequeira, M. Herrera, M. Graells, and L. Puigjaner, "On-line process optimisation: parameter tuning for the real time evolution," *Computers and Chemical Engineering*, vol. 28, no. 5, 2004, pp. 661-672.
- [9] H. A. Morris, "Advanced modelling for small glass furnaces," MSc thesis, Department of Mechanical Engineering, West Virginia University, Morgantown, USA, 2007.
- [10] S. Di Cairano. (2012). An industry perspective on MPC in large volumes applications: Potential benefits and open challenges. [Online]. Available: <http://www.merl.com/publications/docs/TR2012-061.pdf>.
- [11] M. Ławryńczuk, *Computationally Efficient Model Predictive Control Algorithms: A Neural Network Approach*, 1st ed, Heidelberg, Springer, 2014.
- [12] L. Huisman, "Control of glass melting processes based on reduced CFD models," PhD dissertation, Technical University of Eindhoven, 2005.
- [13] M. K. Choudhary, R. Venuturumilli, and M. R. Hyre, "Mathematical modelling of flow and heat transfer phenomena in glass melting, delivery, and forming processes," *International Journal of Applied Glass Science*, vol. 1, pp. 188-214, 2010.
- [14] A. Farina, A. Klar, R. M. M. Mattheij, A. Mikelic, and N. Siedow, *Mathematical Models in the Manufacturing of Glass*, Lecture Notes in Mathematics. Springer-Verlag Berlin Heidelberg.
- [15] COMSOL Multiphysics Modelling Software. [Online]. Available: <http://www.comsol.co>
- [16] H. Kopetz, *Real-Time Systems: Design Principles for Distributed Embedded Applications*, 2nd ed, Springer, 2011.
- [17] J. Belanger, P. Venne, and J.-N. Paquin. (2010). The what, where, and why of real-time simulation. [Online]. Available: <http://www.opal-rt.com/technical-document/what-where-and-why-real-time-simulation>
- [18] L. Vasiliauskien and E. R. Baušys, "Intelligent initial finite element mesh generation for solutions of 2D problems," *Informatica*, vol. 13, no. 2, pp. 239-250, 2002.
- [19] A. Pilat, "Modelling, investigation, simulation, and PID current control of active magnetic levitation FEM model," *18th International Conference on Methods and Models in Automation and Robotics*, pp. 299-304, Międzyzdroje, 2013.



Wojciech Grega received his M.Sc. in electrical engineering and Ph.D. and D.Sc. degrees in automatic control from AGH University of Mining and Metallurgy in Krakow. Currently, he is a full professor of AGH in Krakow: digital control, optimisation methods, distributed control and industrial control systems. He as a author and co-author of more than 150 papers and books. He has been the coordinator or main researcher in 19 national and international projects; the vice dean of the Faculty in 1994, the head of the Control Laboratory since 2000, the head of the Faculty Commission for Education from 2001 to 2009. He is a elected member of the European Association for Education in Electrical and Information Engineering; IEEE Society Member; European Union Academic Expert; KIC InnoEnergy Poland+ Educational Director.



A. K. Pilat was born in Krakow in 1972. He received M.Sc. in 1996, Ph.D. in 2002, and D.Sc. in 2014, all of them in the field of control science and robotics from AGH University of Science and Technology, Krakow, Poland (degree in control science and robotics). He is working as an associate professor at the Department of Control Science and Biomedical Engineering at AGH. He is the author

of more than 90 publications, 5 patents and he supported 8 mechatronic devices with control systems or automation units design. His main field of interest is the design and development of a new devices, upgrading of existing ones and control optimisation supported by the virtual prototyping method. The main field of research is focused on design and control of active magnetic levitation technology applied to magnetic suspension and bearings. He is a member of the IEEE, POLSPAR, IEEE Control Systems Society and IEEE Robotics and Automation Society.



Andrzej J. Tutaj was born in Jaslo, Poland in 1975. He received his M.Sc. degree in automatics and robotics from AGH University of Science and Technology in Krakow, Poland in 2000 and the Ph.D. degree in automatics and robotics from AGH UST in 2009. Currently, he is an associate professor at the Department of Automatics and Biomedical Engineering at AGH University of Science and Technology in Krakow. His main areas of interest include distributed control systems, state and parameter estimation, model-based control methods and industrial automation systems implementation.