Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-25 (2016) 037–042

Model Predictive Control for Individual Room Control

Jan Glos *

* *Department of Control and Instrumentation, Faculty of Electrical Engineering and Communication, Brno University of Technology, Brno, Czech Republic (e-mail: xglosj00@stud.feec.vutbr.cz)*

Abstract: This paper deals with model predictive control (MPC) for individual room control (IRC). Two models of a selected room were assembled, each with different complexity. The first of them is used as an exact replacement of the real room for a controller verification, the second one is used for a state observer and MPC controller design. Both the models were created using only a building documentation, what can be suitable especially for large buildings with hundreds of rooms. It was shown that MPC controller can be advantageously used for room temperature control. In the next work weather forecast and occupancy estimation could be incorporated into MPC controller design, what can bring energy savings.

Keywords: model predictive control, individual room control, temperature control, state observer.

1. INTRODUCTION

Model predictive control (MPC) has been successfully used in various control applications, starting from oil plants (García et al. (1989)), chemical industry (Gustafson (1987)), power plants (Gibbs et al. (1991)) and others.

In last years MPC was many times proposed for use in heating, ventilation and air conditioning (HVAC). Prívara et al. (2011) deployed MPC for university building heating control and reached significant energy savings, Rogers et al. (2014) proposed MPC controller for family house and achieved improved comfort and Huang et al. (2014) dealt with HVAC control for airport terminal building. Comprehensive review of MPC usage in HVAC has been created by Afram and Janabi-Sharifi (2014). Most authors focuses only on control of the building as a whole, where heating and cooling is centralized. It is the common case of older buildings, but the situation is different for recently constructed buildings.

Usually almost every room in such a building is equipped with individual room control (IRC), which addresses temperature control in individual rooms of the building. It usually consists of (sometimes programmable) controller with inputs and outputs, a control panel with a temperature sensor, a heating radiator with an electrically driven valve and a cooling fan coil with a variable speed blower and an electrically driven valve. Certainly there are many possible configurations, we chose the one described above as the most common one. Commonly the valves are controlled by PI controllers, which needs to be properly designed to achieve sufficient comfort for people in the room and energy efficiency for the building owner.

However even if the classic control system is properly adjusted, it can not take into account the weather forecast or occupancy estimation. So for example when the ambient temperature should rise (or the sun will be shining) according to forecast, the MPC controller can turn off the heating radiator and anyway the temperature will reach the setpoint and moreover with some energy conservation due to no (or minimal) temperature overshoot. Considering one room it might not be substantial amount of energy saved, but when it would be extended to whole building or even a campus, the savings may be significant.

In this paper we focus only on basic MPC usage for IRC (without forecasts and estimations), because MPC controller suitability must be verified before implementing complex control rules.

As a process model is strictly needed for MPC, we had to choose the way how to obtain it. Since the building documentation we have is very exact, we tried to build the model only using it. So we found out the heating and cooling power, construction materials and room dimensions and these information was used to build a room thermal model. This approach is beneficial especially for its easy repeatability for another room.

2. THERMAL MODELS OF A ROOM

Two dynamic thermal models of the room were assembled. The first one is based on Simulink Simscape library and is very complex and accurate. The second model was constructed using replacement scheme, which was afterwards used to build the state-space model.

Simscape model is created using thermal masses, conductive heat transfer and heat sources. The base of the model is the air in the room, which is connected to the building elements (like wall, ceiling, window etc.) and these are coupled to the other rooms and to the ambient air. Each building element is composed of multiple layers, which corresponds to a real element structure. For example the wall between the room air and the ambient air is composed of seven layers (drywall, air, bricks, steel plate, mineral

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.12.007

$$
\frac{d}{dt}\begin{pmatrix}\vartheta_1(t)\\ \vartheta_2(t)\\ \vartheta_3(t)\\ \vartheta_4(t)\\ \vartheta(t)\\ \vartheta_o(t)\\ x_{FC}(t)\\ x_{rad1}(t)\\ x_{rad2}(t)\end{pmatrix} = \begin{pmatrix}
-\frac{1}{C_1R_1} & 0 & 0 & 0 & \frac{1}{C_1R_{1i}} & \frac{1}{C_1R_{1o}} & 0 & 0 & 0\\
0 & -\frac{1}{C_2R_2} & 0 & 0 & \frac{1}{C_2R_{2i}} & \frac{1}{C_2R_{2o}} & 0 & 0 & 0\\
0 & 0 & -\frac{1}{C_3R_3} & 0 & \frac{1}{C_3R_{3i}} & \frac{1}{C_3R_{3o}} & 0 & 0 & 0\\
0 & 0 & 0 & -\frac{1}{C_4R_4} & \frac{1}{C_4R_{4i}} & \frac{1}{C_4R_{4o}} & 0 & 0 & 0\\
\frac{1}{C_rR_{1i}} & \frac{1}{C_rR_{2i}} & \frac{1}{C_rR_{3i}} & \frac{1}{C_rR_{4i}} & -\frac{1}{C_rR_i}-\frac{C_{AHU}}{C_r} & \frac{C_{AHU}}{C_r} & -\frac{4P_{FC}}{75C_r} & 0 & \frac{32P_{rad}}{625C_r}\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{300} & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{160} & -\frac{2}{625}\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{512} & 0
\end{pmatrix}\begin{pmatrix}\vartheta_1(t)\\ \vartheta_2(t)\\ \vartheta_3(t)\\ \vartheta_4(t)\\ \vartheta(t)\\ \vartheta_o(t)\\ x_{FC}(t)\\ x_{rad1}(t)\\ x_{rad2}(t)\end{pmatrix}
+\begin{pmatrix}0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ \frac{1}{16} & 0\\ 0 & \frac{1}{16}\\ 0 & 0\end{pmatrix}\begin{pmatrix}u_c(t)\\ u_h(t)\end{pmatrix}, \quad (1)
$$

$$
y(t) = \begin{pmatrix}0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}\vartheta_1(t)\\ \vartheta_2(t)\\ \vartheta_3(t)\\ \vartheta_4(t)\\ \vartheta(t)\\ \vartheta_o(t)\\ x_{FC}(t)\\ x_{rad1}(t)\\ x_{rad2}(t)\end{pmatrix} + \begin{pmatrix}0 & 0\end{pmatrix}\begin{pmatrix}u_c(t)\\ u_h(t)\end{pmatrix}, \quad (2)
$$

wool, aluminium plate and terracotta panels). Each layer has defined thermal capacity and conductive heat transfer to both the adjacent layers. An example of one layer can be found in Fig. 1 (it can represent for example the bricks layer). Subsequently we added other important parts of the room model. The heating radiator and the fan coil are modelled as an ideal heat source with appropriate dynamics and adequate thermal power. Additionally influence of central air handling unit (AHU) was added to make the model as accurate as possible. The parameters of all model components were taken from building documentation, so the model can be constructed for any other room. The model was several times compared to the measured data and it was found that the model is satisfactory, see Fig. 2 and 3. Minor differences between the model and the real room behaviour can be caused by various disturbances (heat generated by human bodies, computers etc.), the measurements were carried out during normal room operation.

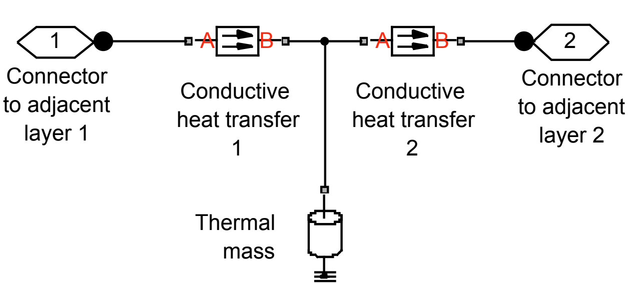

Fig. 1. One layer of construction element in Simscape room model

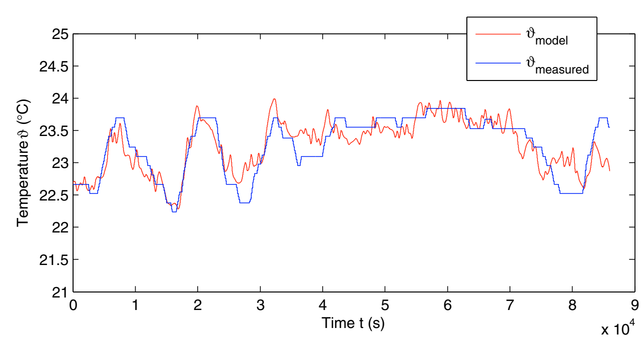

Fig. 2. Comparison of Simscape model and measured temperature during heating

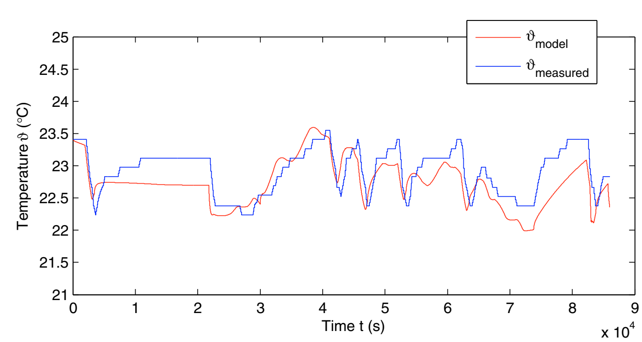

Fig. 3. Comparison of Simscape model and measured temperature during cooling

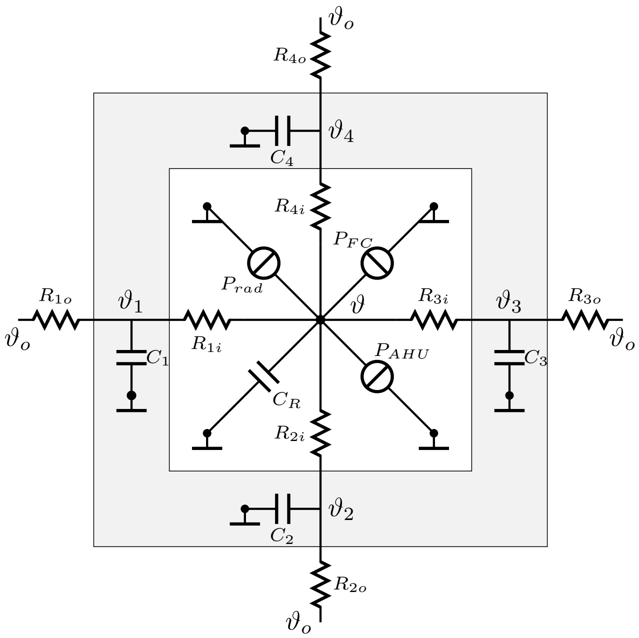

Fig. 4. Room replacement scheme for thermal model

The second model in (1) and (2) had to be constructed due to high complexity of Simscape model. The structure of the model was simplified, each building element is represented only by one thermal capacity and two thermal resistances. The dominant parameters of all element layers were taken as the parameters of resulting simplified element.

The replacement scheme in Fig. 4 was complemented with known transfer function of the fan coil

$$F_{FC}(p) = \frac{P_{FC}}{300p+1}, \tag{3}$$

where P_{FC} denotes cooling power of the fan coil and the time constant includes the valve and the heat exchanger dynamics. The radiator transfer function was found in the form

$$F_{rad}(p) = \frac{P_{rad}}{(800p+1)(200p+1)}, \tag{4}$$

where P_{FC} stands for heating power of the radiator and the larger time constant is caused by the radiator, the shorter one by the valve with thermoelectric drive.

For both the replacement scheme (Fig. 4) and (1-2) following symbols were used. ϑ_{1-4} denotes the temperature inside the construction element, ϑ_o stands for both the other rooms and the outside temperature, x are internal states of the heating radiator and the fan coil. C is used for thermal capacitance, R for thermal resistance and the subscripts have following meaning: first position denotes the construction element (r stands for controlled room and AHU for central air handling unit), second position is used to divide thermal resistance to inner and outer parts. Additionally two substitutions were used to simplify the matrix equations

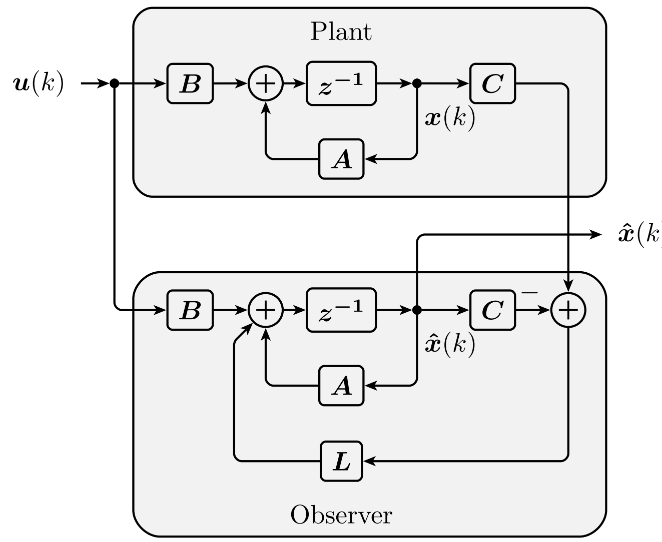

Fig. 5. State observer for MPC controller (from Lewis et al. (2007))

$$R_i = \sum_{n=1}^{4} R_{ni}, \tag{5}$$

$$R_n = R_{ni} + R_{no}, \tag{6}$$

where n refers to construction element index. P_{FC} and P_{rad} refers to cooling and heating power respectively.

After combining all parts of the model we obtained the state-space representation, which is almost suitable for MPC controller design (see (1) and (2)). Moreover we added integrators at the model inputs to get rid of steady-state error, then the output of MPC controller is not the valve opening degree (u), but the change of that degree (Δu). Afterwards the model was discretized in order to be used for a state observer and MPC controller design.

3. STATE OBSERVER

MPC controller needs current state vector values to compute the control signal. But not all of the system states are measurable, therefore a state observer in Fig. 5 had to be employed (from Lewis et al. (2007)).

We assume a dynamical system in the form

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k), \tag{7}$$

$$y(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k), \tag{8}$$

where $\boldsymbol{x}(k)$ are states, $\boldsymbol{u}(k)$ are inputs and $y(k)$ is output of the system with matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D} = 0$. Then the observer can be described by equation

$$\hat{\boldsymbol{x}}(k+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{L}(y(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k)) + \boldsymbol{B}\boldsymbol{u}(k) \tag{9}$$

For the estimation error $\tilde{\boldsymbol{x}}(k)$ we can write

$$\tilde{\boldsymbol{x}}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k), \tag{10}$$

$$\tilde{\boldsymbol{x}}(k+1) = \boldsymbol{x}(k+1) - \hat{\boldsymbol{x}}(k+1), \tag{11}$$

which after few modifications leads to error dynamics

$$\tilde{\boldsymbol{x}}(k+1) = (\boldsymbol{A} - \boldsymbol{LC})\tilde{\boldsymbol{x}}(k). \tag{12}$$

Now it is evident that $(\boldsymbol{A} - \boldsymbol{LC})$ will determine the estimation error behaviour. We choose $\boldsymbol{L}$ so that $(\boldsymbol{A}-\boldsymbol{LC})$ is stable and usually faster than the controlled system. Particular values of $\boldsymbol{L}$ can be obtained by solving equation

$$|p\boldsymbol{I} - (\boldsymbol{A} - \boldsymbol{LC})| = \prod_{i=1}^{n}(p - \lambda_i), \tag{13}$$

where λ_i are desired eigenvalues of matrix $(\boldsymbol{A} - \boldsymbol{LC})$ and n is the system order.

4. MODEL PREDICTIVE CONTROL

Based on Lee (2011), to find an optimal control for the system described by (7), a cost function

$$\begin{aligned} J_p(k) &= \hat{\boldsymbol{x}}^T(k+p)\boldsymbol{Q_t}\hat{\boldsymbol{x}}(k+p) \\ &+ \sum_{i=0}^{p-1}\Big[\hat{\boldsymbol{x}}^T(k+i)\boldsymbol{Q}\hat{\boldsymbol{x}}(k+i) \\ &+ \hat{\boldsymbol{u}}^T(k+i)\boldsymbol{R}\hat{\boldsymbol{u}}(k+i)\Big], \end{aligned} \tag{14}$$

is to be minimized. A prediction horizon is denoted as p, matrixes Q_t, Q and R are used as weighting matrixes for final state, future states in prediction horizon and control action respectively. Also constraints

$$\hat{\boldsymbol{u}}(k+i) \in \mathbb{U}, i \in \langle 0, p-1\rangle, \tag{15}$$
$$\hat{\boldsymbol{x}}(k+i) \in \mathbb{X}, i \in \langle 1, p-1\rangle, \tag{16}$$
$$\hat{\boldsymbol{x}}(k+p) \in \mathbb{Z}, \tag{17}$$
$$\hat{\boldsymbol{x}}(k+i+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k+i) + \boldsymbol{B}\hat{\boldsymbol{u}}(k+i), i \in \langle 0, p-1\rangle, \tag{18}$$

have to be fulfilled during the criterion minimizing.

We used MPT toolbox by Kvasnica et al. (2004) for MPC controller design. YALMIP by Lofberg (2004) was also used in this work.

Using tools above we designed two MPC controllers. The first one is intended for room heating and the second one for room cooling. We designed both the "explicit" and "on-line" controllers. The word "explicit" means that the control laws are pre-computed and during controller operation it only chooses the suitable control based on system states (Kouramas et al. (2011)). The "on-line" controller computes the optimal control action during the operation. Decision between these two types of controllers can be subject of next research, because it is not clear in this case. We used quite large sampling time (200 s for heating, 50 s for cooling), so it might be possible to compute the control action between the instances of sampling.

To ensure the temperature setpoint achievement, we extended our problem by time varying reference tracking. In that case the state vector is extended with reference state vector and input vector. Also the cost function (14) is completed with additional element

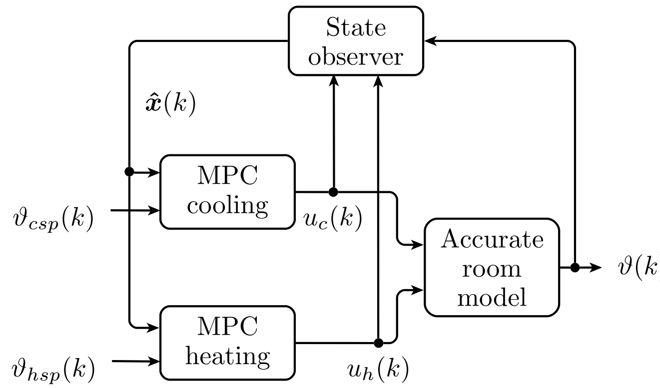

Fig. 6. Control loop for MPC controllers

$$\begin{aligned} J_p(k) &= \hat{\boldsymbol{x}}^T(k+p)\boldsymbol{Q_t}\hat{\boldsymbol{x}}(k+p) \\ &+ \sum_{i=0}^{p-1}\Big[\hat{\boldsymbol{x}}^T(k+i)\boldsymbol{Q}\hat{\boldsymbol{x}}(k+i) \\ &+ \hat{\boldsymbol{u}}^T(k+i)\boldsymbol{R}\hat{\boldsymbol{u}}(k+i) \\ &+ \boldsymbol{e}^T(k+i)\boldsymbol{Q_y}\boldsymbol{e}(k+i)\Big], \end{aligned} \tag{19}$$

where $\boldsymbol{e}$ denotes the control error - difference between references $\boldsymbol{y_{ref}}$ and measured outputs $\boldsymbol{y}$. Matrix $\boldsymbol{Q_y}$ is used as weighting coefficient of control error.

By specifying the values of matrices $\boldsymbol{Q_t}$, $\boldsymbol{Q_y}$, $\boldsymbol{Q}$ and $\boldsymbol{R}$ we adapted the controllers to fit the requirements of IRC (as quick as possible transient, minimal overshoot).

Fig. 6 illustrates the control loop with MPC controllers. There are temperature setpoints for cooling (ϑ_{csp}) and for heating (ϑ_{hsp}), measured (modelled) temperature ϑ, estimated state vector $\hat{\boldsymbol{x}}$, u_c and u_h denotes the statuses of the valves for cooling and heating respectively. The accurate room model was described in section 2 and the state observer in section 3.

5. SIMULATIONS

Several simulations were performed in order to verify MPC controllers suitability for IRC. In Fig. 7 there is a basic check of the controller for heating - a step of setpoint. It imitates the situation when the user changes the temperature setpoint using control panel, when he is cold. It can be seen that the transient is the fastest possible (the radiator valve is opened to 100 % until the reference is reached). Only a minimal overshoot occurs after reaching the setpoint.

Verification of MPC controller for cooling is presented in Fig. 8. The situation is similar to heating, a step of setpoint was performed, only the direction is different. In this case the undershoot is a little stronger, but the user will not experience this deviation.

In Fig. 9 a negative disturbance affected the room temperature. In real room operation it occurs when the outdoor temperature suddenly falls or in the case of ventilation through the window. It can be seen that the measured temperature follows the reference temperature, so MPC controller works well even for disturbance rejection.

Similar test was executed for MPC controller for cooling, in this case it can represent the same conditions as for heating

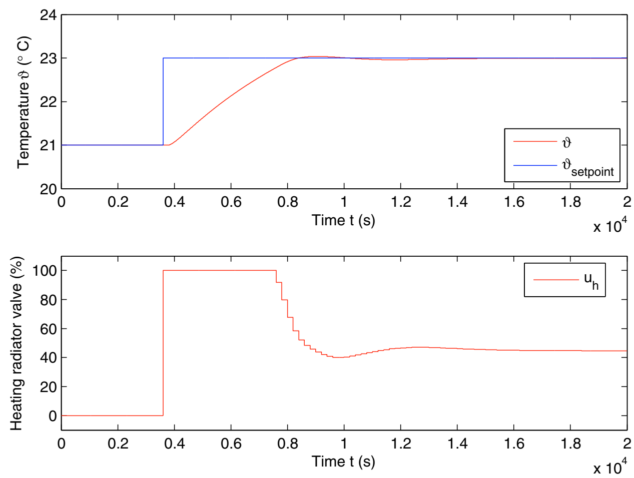

Fig. 7. Temperature setpoint change for heating using MPC controller

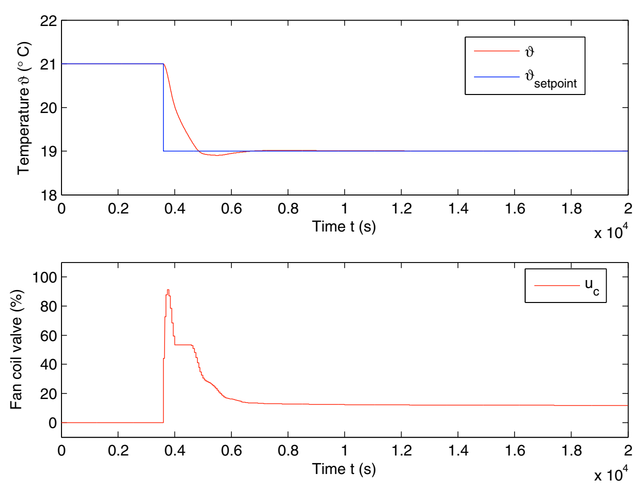

Fig. 8. Temperature setpoint change for cooling using MPC controller

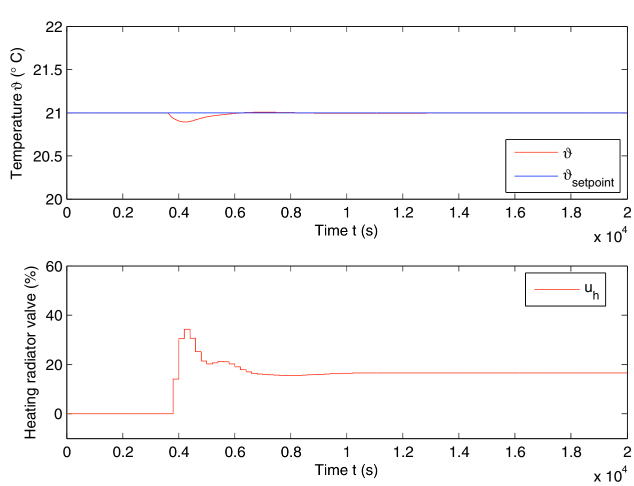

Fig. 9. Significant outdoor temperature decrease

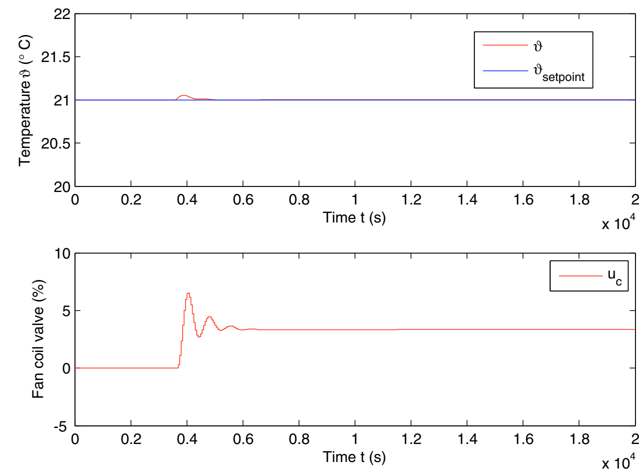

Fig. 10. Significant outdoor temperature increase

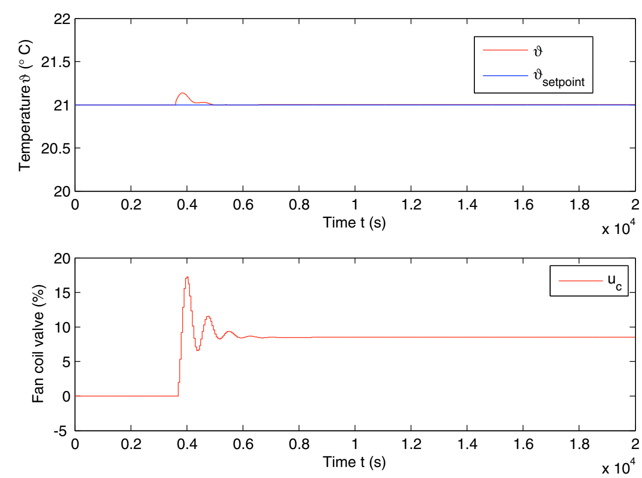

Fig. 11. Room occupancy change

(outdoor temperature increase or window ventilation). The result can be found in Fig. 10, the setpoint is satisfied quite well.

Another verification situation of MPC controller for cooling was conceived - heat gain inside the room. It can represent change of room occupancy (each human dissipates some heat), heating by computers and other equipment. A simulation experiment in Fig. 11 represents occupancy increase of five humans (each with heat gain of 80 W, together 400 W). It is apparent that MPC controller can handle even this situation properly, with only a small error just after disturbance start and with no steady state error.

6. CONCLUSION

In this paper we tried to investigate possible MPC applications in IRC. We constructed two dynamic models of the selected room. The first one (assembled using Simscape) is very precise and can be compiled for any other room based only on construction documentation. The second one was also build without using any measurements, but it had to be much more simple to allow MPC controllers design. So the model is not so accurate and a lot of averaging and simplifying had to be applied. But this approach also provides a possibility to set some constraints on the states of the system. For future work we would recommend using some kind of grey-box model. It could have fixed structure (like in Fig. 4) and the parameters would be identified using accurate model or measured data. This approach could ensure better model accuracy and preserve possibility of state constraints and simplicity for MPC controller design.

We designed two MPC controllers (for cooling and heating) and we verified them using several simulation experiments. The results show that MPC controllers can be advantageously used for IRC. There were only very small overshoots and all disturbances were rejected.

In this work we only investigated MPC controllers suitability for room temperature control in basic control problems (reference tracking and disturbance rejection). Next steps may consist of incorporating weather forecast combined with sunshine estimation into MPC controllers design. This extension could bring energy savings especially in case of heating.

Another possibility of improvement can be found in adding an occupancy forecast. It could provide better thermal comfort for people in the room and also some energy savings (heating can be disabled before setpoint reaching).

It may seem that mentioned energy savings are negligible and it might be truth if we consider only one room. But if we take into account that one building may consist of hundreds of rooms, then the savings can be significant. And if we take into consideration for example university campus, which may contain dozens of such buildings, substantial costs could be saved.

Obviously the next step should be MPC controllers verification with real room. There might be additional issues with implementing MPC controllers, because commercially available IRC controllers do not usually allow to change their internal software equipment (and usually the processor and the memory are insufficient). Therefore a completely new device would have to be developed, which will allow MPC controllers implementation. Other possible way would employ a standard IRC controller and another computer for MPC computations. This solution might be faster from the perspective of the development, but the system would be strictly dependent on communication reliability and it will become centralized instead of distributed.

Further work can also focus on MPC controller type decision (between "on-line" and "explicit" versions). Both the types have their advantages and disadvantages, so it should be verified in real operation which type is better in this case. On the one hand, explicit MPC controller has no special requirements for a processor, but pre-computed control rules can occupy quite large amount of memory. On the other hand, on-line controllers does not need large memory, but the control action must be computed in real-time, so the processor must be powerful enough. Another benefits of on-line MPC controller are easier and faster controller modifications.

ACKNOWLEDGEMENTS

This work was supported by grant No. FEKT-S-14-2429 - "The research of new control methods, measurement procedures and intelligent instruments in automation", which was funded by the Internal Grant Agency of Brno University of Technology.

REFERENCES

Afram, A. and Janabi-Sharifi, F. (2014). Theory and applications of HVAC control systems A review of model predictive control (MPC). *Building and Environment*, 72, 343–355. doi:10.1016/j.buildenv.2013.11.016.

García, C.E., Prett, D.M., and Morari, M. (1989). Model predictive control: Theory and practiceA survey. *Automatica*, 25(3), 335–348. doi:10.1016/0005-1098(89)90002-2.

Gibbs, B.P., Weber, D.S., and Porter, D.W. (1991). Application of Nonlinear Model-based Predictive Control to Fossil Power Plants. In *Proceedings of the 30th Conference on Decision and Control*, 1850–1856.

Gustafson, D.E. (1987). Model predictive control (MPC) of injection molding machines. In *Journal of Chemical Information and Modeling*, volume 53, 160.

Huang, H., Chen, L., and Hu, E. (2014). Model predictive control for energy-efficient buildings: An airport terminal building study. In *11th IEEE International Conference on Control & Automation (ICCA)*, 1025–1030. IEEE. doi:10.1109/ICCA.2014.6871061.

Kouramas, K., Faísca, N., Panos, C., and Pistikopoulos, E. (2011). Explicit/multi-parametric model predictive control (MPC) of linear discrete-time systems by dynamic and multi-parametric programming. *Automatica*, 47(8), 1638–1645. doi:10.1016/j.automatica.2011.05.001.

Kvasnica, M., Grieder, P., Baotic, M., and Morari, M. (2004). Multi-parametric toolbox (MPT). *Hybrid Systems: Computation and Control*, 2993, 121–124.

Lee, J.H. (2011). Model predictive control: Review of the three decades of development. *International Journal of Control, Automation and Systems*, 9(3), 415–424. doi:10.1007/s12555-011-0300-6.

Lewis, F.L., Xie, L., and Popa, D. (2007). *Optimal and Robust Estimation: With an Introduction to Stochastic Control Theory, Second Edition.*

Lofberg, J. (2004). YALMIP : a toolbox for modeling and optimization in MATLAB. *2004 IEEE International Conference on Computer Aided Control Systems Design*, 284–289.

Prívara, S., Široký, J., Ferkl, L., and Cigler, J. (2011). Model predictive control of a building heating system: The first experience. *Energy and Buildings*, 43(2-3), 564–572.

Rogers, D., Foster, M., and Bingham, C. (2014). Experimental investigation of a Recursive Modelling MPC system for space heating within an occupied domestic dwelling. *Building and Environment*, 72, 356–367. doi:10.1016/j.buildenv.2013.11.014.