Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Simple Pulse-Step Model Predictive Controller ⋆

Miloš Schlegel * **Jaroslav Sobota** **

* *University of West Bohemia in Pilsen, Department of Cybernetics, Univerzitní 8, 306 14 Plzeň, Czech Republic, (e-mail: schlegel@kky.zcu.cz)*
** *(e-mail: jsobota@kky.zcu.cz)*

Abstract: This paper describes a novel simple model based predictive controller with manipulated value constraints. This controller is suitable for substitution of the classical PID controller used in industrial practice. It is assumed that the controlled system is stable, linear and t-invariant FIR system. The discrete step response sequence is used as the process model. Alternatively it is possible to use three-parameter model. To make the open-loop optimization easier the set of admissible control sequences is restricted to stepwise pulse-step sequences. The optimization procedure is then executable in reasonable time. A single tuning parameter is available for manual fine-tuning of the controller - the control moves penalty coefficient.

Keywords: optimal control, predictive control, step function responses, constraints, reduction, quadratic performance indices.

1. INTRODUCTION

The predictive control approach is an up-to-date topic in the automation sphere, which can be proven by many recent papers and books [J.M. Maciejowski, 2002, Huang et al., 2002]. It is the only advanced control technique which found its place in industrial process control aside the classical PID control. The main advantage of the predictive control is its general principle which is suitable for both linear and nonlinear systems and also the possibility to include constraints directly into the design procedure. On the other hand, this generality brings several problems, especially the computational cost which makes the implementation of predictive control algorithms into compact controllers and PLCs almost impossible.

To become a classical PID controller substitution, the predictive controller must fulfill the following:

- it is as easy (or easier) to use as the classical PID controller
- the performance of the closed loop system is distinctively better
- the existing compact controllers and PLCs are able to meet its computational and memory requirements

2. SIMPLE PREDICTIVE CONTROLLER

2.1 The pulse-step control sequence

A tough problem in predictive control with constraints is its complexity and computational cost. To lower the computational burden, it is possible to use some blocking strategy [Tondel and Johansen, 2002], for example constant manipulated value or constant manipulated value differences over time intervals of specified length. Another possibility is the so-called functional predictive control [Richalet et al., 1987], where the control sequence is restricted to a linear combination of suitable base functions.

Alternative approach to complexity restriction presented in this paper is based on the so-called pulse-step control, a well known aggressive technique used for manual control in industrial practice. The properties of pulse-step feedforward control in combination with the classical PID feedback control were studied by Wallén and Åström [2002]. This paper shows how to incorporate the pulse-step control idea into MPC. As shown in Figure 1, the control sequence $u(k)$ begins with n_1 maximal (minimal) elements, followed by n_2-n_1 minimal (maximal) elements according to the constraints $u^- \leq u(k) \leq u^+$. The remaining part of the control sequence is constant, $u(k) = u^\infty$ for $k > n_2$, where n_1 and n_2 are limited by the control horizon H_C, $0 \leq n_1 \leq n_2 < H_C$ and of course u^∞ is subject to constraints $u^- \leq u^\infty \leq u^+$. So the whole control sequence is determined by only 3 variables n_1, n_2, and u^∞.

2.2 The controlled process model

The model based predictive control always employs some model of the controlled process. In this approach, the discrete step response $g(j),\ j = 1, \ldots, N$ is used. Figure 2 shows how to obtain the discrete step response $g(j),\ j = 0, 1, 2, \ldots$ and the discrete impulse response $h(j),\ j = 0, 1, 2, \ldots$ with sampling period T_S from continuous step response. Note that $g(0) = h(0) = 0$, $h(j) = g(j) - g(j-1)$ for $j \geq 1$.

For stable, linear and t-invariant FIR systems with monotonous step response it is also possible to use the

⋆ This paper was supported by the Ministry of Industry and Trade of the Czech Republic, grant no. FI-IM3/037. This support is gratefully acknowledged.

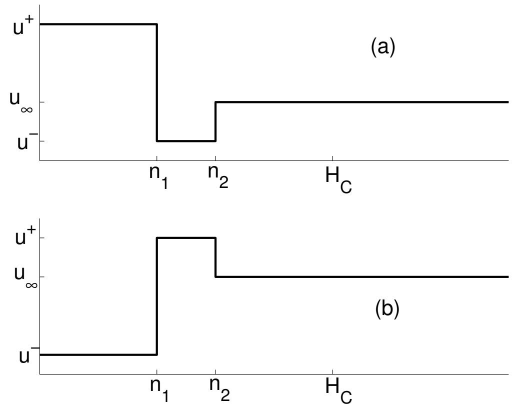

Fig. 1. Example of pulse-step up (a) and pulse-step down (b) control sequence

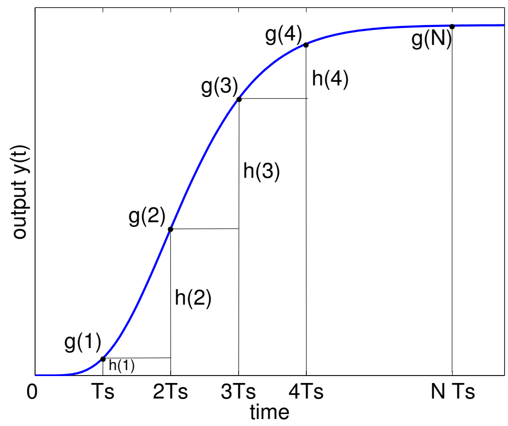

Fig. 2. Discrete step response

moment model set approach [Schlegel and Večerek, 2005] and describe the system by only 3 characteristic numbers κ, μ, and σ^2, which can be obtained easily from a very short and simple experiment. This identification technique has been widely accepted in industrial practice for PID controllers tuning purposes. The characteristic numbers κ, μ, and σ^2 of the system in the form

$$P(s) = \frac{K}{\prod_{i=1}^{l}(\tau_i s + 1)} \cdot e^{-Ds}$$

are defined as

$$\begin{aligned} \kappa &= K, \\ \mu &= D + \sum_{i=1}^{l} \tau_i, \\ \sigma^2 &= \sum_{i=1}^{l} \tau_i^2. \end{aligned} \tag{1}$$

Thus the controlled system can be approximated by first order plus dead-time system

$$P_{FOPDT}(s) = \frac{K}{\tau s + 1} \cdot e^{-Ds}, \tag{2}$$
$$\kappa = K,\ \mu = \tau + D,\ \sigma^2 = \tau^2$$

or second order plus dead-time system

$$P_{SOPDT}(s) = \frac{K}{(\tau s + 1)^2} \cdot e^{-Ds}, \tag{3}$$
$$\kappa = K,\ \mu = 2\tau + D,\ \sigma^2 = 2\tau^2$$

with the same characteristic numbers. The discrete step response of these systems is then used to model the controlled system.

As shown in Figure 3, the characteristic numbers have a clear physical meaning for the systems (2) and (3), so it is also possible to adjust them manually to fit the step response of the real system. The characteristic number κ is static gain, the number μ has the character of time delay (known also as resident time constant, it shifts the step response along the time axis) and the parameter σ^2 changes the slope of the step response.

2.3 Computing the control sequence

Consider the controlled system described by the discrete step response $g(j),\ j = 1, \ldots, N$ obtained either directly from the measurements on the real system or from three-parameter models (2) or (3) described in the previous section. The input $u(k)$ and output $y(k)$ of linear discrete system are related by well known convolution

$$\begin{aligned} y(k) &= \sum_{j=0}^{\infty} h(j)u(k-j) \approxeq \\ &\approxeq \sum_{j=1}^{N} h(j)u(k-j), \end{aligned} \tag{4}$$

where $h(j),\ j = 1, \ldots, N$, is the discrete impulse response of the system and N is suitable natural number ($h(j) \approxeq 0$ for $j = N+1, \ldots, \infty$). From (4) we can obtain another relation which will be used further. It holds

$$\begin{aligned} y(k) &= \sum_{j=1}^{N} h(j)u(k-j) = \\ &= \sum_{j=1}^{N} [g(j) - g(j-1)]\, u(k-j) = \\ &= g(1)u(k-1) + g(2)u(k-2) + \ldots + \\ &+ g(N-1)u(k-N+1) + g(N)u(k-N) - \\ &- g(0)u(k-1) - g(1)u(k-2) - \ldots - \\ &- g(N-2)u(k-N+1) - g(N-1)u(k-N) = \\ &= \sum_{j=1}^{N} g(j)\, [u(k-j) - u(k-j-1)] + \\ &\qquad + g(N)u(k-N-1). \end{aligned}$$

In other form

$$\begin{aligned} y(k) = &\sum_{j=1}^{N} g(j)\Delta u(k-j) + \\ &+ g(N)u(k-N-1), \end{aligned} \tag{5}$$

where $\Delta u(k) = u(k) - u(k-1)$.

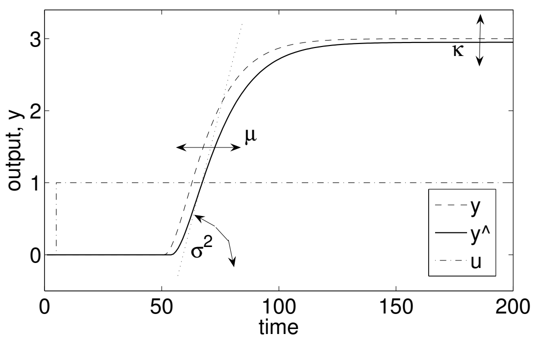

Fig. 3. Physical meaning of the characteristic numbers κ, μ, and σ^2

Then the i-step ahead output prediction at time k, $0 \le i \le N$, is given by

$$\begin{aligned}\hat{y}(k+i|k) &= \sum_{j=1}^{N} g(j)\Delta u(k+i-j) + \\ &+g(N)u(k+i-N-1) = \\ &= \sum_{j=i+1}^{N} g(j)\Delta u(k+i-j) + \\ &+g(N)u(k+i-N-1) + \\ &+\sum_{j=1}^{i} g(j)\Delta\hat{u}(k+i-j|k) = \\ &= \hat{y}_f(k+i|k) + \sum_{j=1}^{i} g(j)\Delta\hat{u}(k+i-j|k), \qquad (6)\end{aligned}$$

where the first term is the response caused by the past inputs and the sum represents the response determined by future changes of the input signal $\Delta\hat{u}(k+i-j|k)$, $j = 1,\dots,i$. The disturbance d (prediction error) is defined as

$$d \triangleq y_m(k) - \hat{y}(k|k-1), \qquad (7)$$

where $y_m(k)$ is the real (measured) output of the system at time k.

For the pulse-step control strategy described in section 2.1 we get from (6)

$$\begin{aligned}\hat{y}(k+i|k) &= \hat{y}_f(k+i|k) + g(i)\Delta\hat{u}(k|k) + \qquad (8)\\ &+g(i-n_1)\Delta\hat{u}(k+n_1|k) + g(i-n_2)\Delta\hat{u}(k+n_2|k),\end{aligned}$$

where

$$\Delta\hat{u}(k|k) = u^+ - u(k-1), \quad \Delta\hat{u}(k+n_1|k) = u^- - u^+,$$
$$\Delta\hat{u}(k+n_2|k) = u^\infty - u^-$$

for the pulse-step up case (Figure 1a) and

$$\Delta\hat{u}(k|k) = u^- - u(k-1), \quad \Delta\hat{u}(k+n_1|k) = u^+ - u^-,$$
$$\Delta\hat{u}(k+n_2|k) = u^- - u^\infty$$

for the pulse-step down case (Figure 1b).

Now the requirement that the system output reaches the desired value w in N_1 steps and stays steady until $N_2{}^{\text{th}}$ step is formulated by

$$\begin{aligned}&\hat{y}(k+N_1|k) + d = \dots = \\ &= \hat{y}(k+N_2|k) + d = w, \qquad (9)\end{aligned}$$

where N_1, N_2 are appropriate natural numbers defining the prediction horizon. Note that the disturbance d given by (7) is presumed to be constant over the whole time interval $0,\dots,N$ (btw. this presumption incorporates integrator into the structure of the controller, which ensures total compensation of arbitrary constant disturbance acting on the system). From equations (8) and (9) we obtain

$$\begin{aligned}w &= \hat{y}_f(k+i|k) + g(i)\Delta\hat{u}(k|k) + \\ &+g(i-n_1)\Delta\hat{u}(k+n_1|k) + \\ &+g(i-n_2)\Delta\hat{u}(k+n_2|k) + d, \quad i = N_1,\dots,N_2. (10)\end{aligned}$$

Note that (10) is a set of linear equations with only a single variable u^∞ for fixed n_1 and n_2. The coincidence condition (9) (or (10)) cannot be fulfilled exactly so it is necessary to define quadratic performance index in the form

$$\begin{aligned}I &= \sum_{i=N_1}^{N_2} \left(\hat{y}(k+i|k) + d - w\right)^2 + \\ &+\lambda \sum_{i=0}^{H_C-1} \Delta\hat{u}(k+i|k)^2 \to \min, \qquad (11)\end{aligned}$$

where the optimized variables are n_1, n_2, $0 \le n_1 \le n_2 \le H_C$, and $u^\infty \in [u^+; u^-]$.

It is important to mention that the parameters N_1, N_2, H_C, and λ in the criterion (11) take the role of design parameters. The parameters N_1 and N_2 define the coincidence interval (9) and strongly influence the resulting optimal control sequence. The standard choice is $N_1 = 0$ and $N_2 = N-1$. If dead-time D is present at the controlled system, it is reasonable to set $N_1 > D/T_S$, where T_S is the sampling frequency. For monotonous step response systems it is possible to use $N_2 << N$ to speed up the closed loop. The control horizon H_C, $1 \le H_C \le N-1$, influences the closed loop performance and mainly the complexity of the optimization procedure. If the requirements on the speed of the closed loop are not critical, it is possible to use the prediction horizon of $H_C = 1$, otherwise the choice of $H_C = 10$ seems to be the most appropriate (supposing the sampling frequency is adequate with respect to the controlled system dynamics). In the case when $H_C = 1$, the nonlinear part of the control action cannot be applied and the only optimized variable

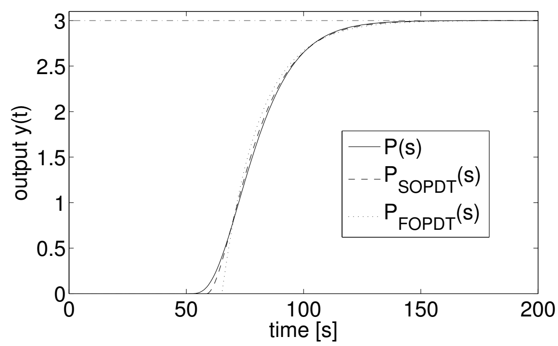

Fig. 4. Step responses of the system (12) and its approximations

is u^∞. Finally the parameter λ penalizes the changes in the control signal. The greater this parameter is, the less aggressive controller we get. Recommended value to start the tuning from is $\lambda = 0$.

The algorithm used for solving the optimization task (11) combines brute force and the least squares method. The value u^∞ is determined using the least squares method for all admissible combinations of n_1 and n_2 and the optimal control sequence is selected afterwards. The computational cost is proportional to ${H_C}^2$. The selected sequence in the pulse-step shape is optimal in the open-loop sense. To convert from open-loop to closed-loop control strategy, only the first element of the computed control sequence is applied and the whole optimization procedure is repeated in the next sampling instant.

3. EXAMPLE

The properties of the model based predictive controller based on the algorithm described in section 2.3 will be illustrated here. Consider the controlled system described by the transfer function

$$P(s) = \frac{3}{(5s+1)^2(10s+1)^2} \cdot e^{-50s} \tag{12}$$

and manipulated value constraint $u \in \langle 0, 1 \rangle$. The sampling frequency of $T_S = 1$s will be used. It is of course necessary to work with the discretized model of the system (12) because the predictive control algorithm is discrete by its nature.

Figure 4 illustrates the step responses which will be used for prediction of the controlled system behavior. They belong to the system (12) and its approximations in the form (2) and (3) with the same characteristic numbers κ, μ, and σ^2. Note that the most significant discrepancies occur at the beginning of the step responses, while the static gain of all systems is the same.

Firstly the exact discrete step response is used for prediction. Figure 5 compares the behavior of pulse-step predictive controller to the classical PID controller. The PID controller was tuned in the virtual PID laboratory [Schlegel and Čech, 2004] with respect to the following design specifications: gain margin $G_m = 2$, phase margin $P_m = 60°$, and restriction on the peak of the sensitivity function $M_S < 1.8$. The step response has only a small overshoot and reaches the steady state much faster with the predictive controller. Notice the pulse-step shape of the control sequence. The manipulated value constraints are kept and fully exploited when setpoint changes. The input disturbance is also rejected faster by the predictive

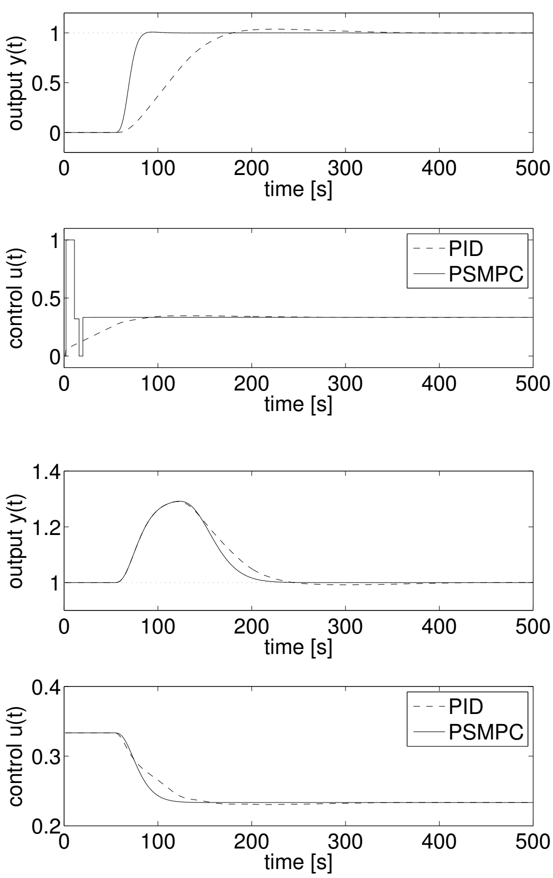

Fig. 5. The setpoint step change response and input step disturbance rejection, the exact model is used for prediction, $T_S = 1s$, $H_C = 5$, $N_1 = 50$, $N_2 = 150$, $\lambda = 0.1$, $N = 190$.

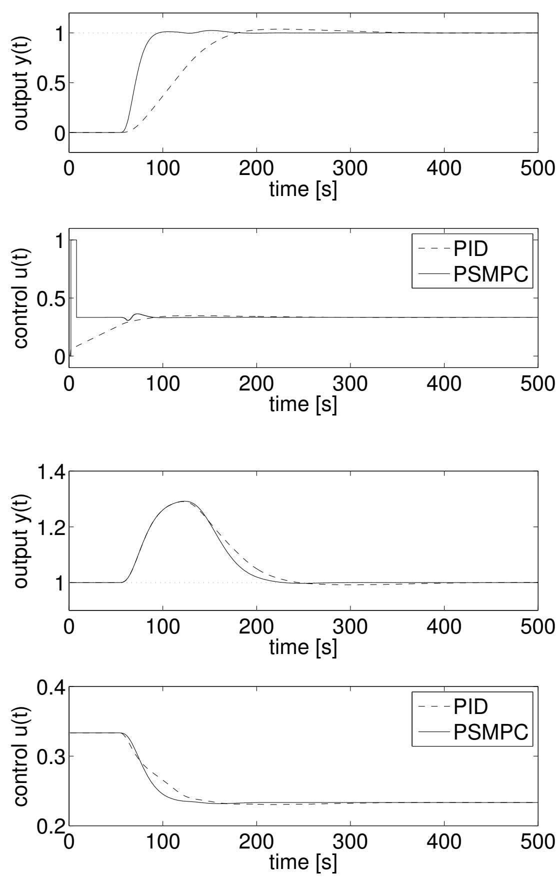

Fig. 6. The setpoint step change response and input step disturbance rejection, the SOPDT approximation is used for prediction, $T_S = 1s$, $H_C = 5$, $N_1 = 50$, $N_2 = 150$, $\lambda = 0.15$, $N = 190$.

controller but the improvement is not so significant in this case.

Now the second order plus dead-time (SOPDT) approximation was used for prediction of the controlled system behavior. It was necessary to increase the λ parameter to avoid too aggressive behavior of the controller resulting from discrepancies between the model and the real system. The controller is then more conservative and one can see in the Figure 6 that the control sequences changed. For the step response it degraded from pulse-step shape to pulse-constant shape. This leads to a small overshoot and a bit slower step response, but the PSMPC still outperforms the classical PID controller. Notice the wobbling control signal within the interval 50-100 seconds. The predictive controller deals with the discrepancy between the model and the real system during this phase.

The last set of responses is depicted in Figure 7. In this case the controlled system is modelled by first order plus dead-time (FOPDT) system, which means the difference between the real system and the model is even bigger. Thus the λ parameter had to be increased once again, otherwise wild oscillations would occur. Small overshoot occurs again but the step response is still significantly faster with the predictive controller. The input disturbance rejection remains almost the same regardless of the prediction model used.

Robustness

As was shown in the previous section even such simple predictive controller can provide a high quality closed loop. It was shown that the controller works fine even if a rough approximation of the controlled system is used for prediction. To test the robustness more deeply, inaccurate characteristic numbers κ, μ and σ^2 were used. Such situation can occur in industrial practice as the characteristic numbers are usually computed from measurements on the real system, where noise and disturbances are always present. All the characteristic numbers were successively perturbed by $\pm 10\%$ and the resulting perturbed FOPDT systems were used for prediction, while keeping the parameters from the experiment with the original FOPDT system.

Figure 8 compares the closed loop behavior when the FOPDT and perturbed FOPDT approximations are used for prediction. The inaccurate characteristic numbers result in small under- or overshoot but the overall performance is still acceptable. Furthermore it is possible to easily adjust the characteristic numbers manually in such case, because they have a clear physical meaning

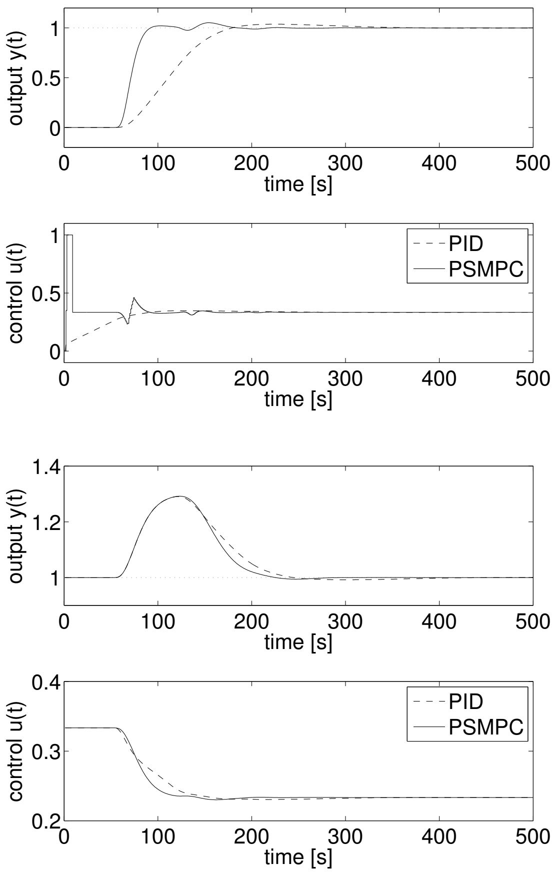

Fig. 7. The setpoint step change response and input step disturbance rejection, the FOPDT approximation is used for prediction, $T_S = 1s$, $H_C = 5$, $N_1 = 50$, $N_2 = 150$, $\lambda = 0.18$, $N = 190$.

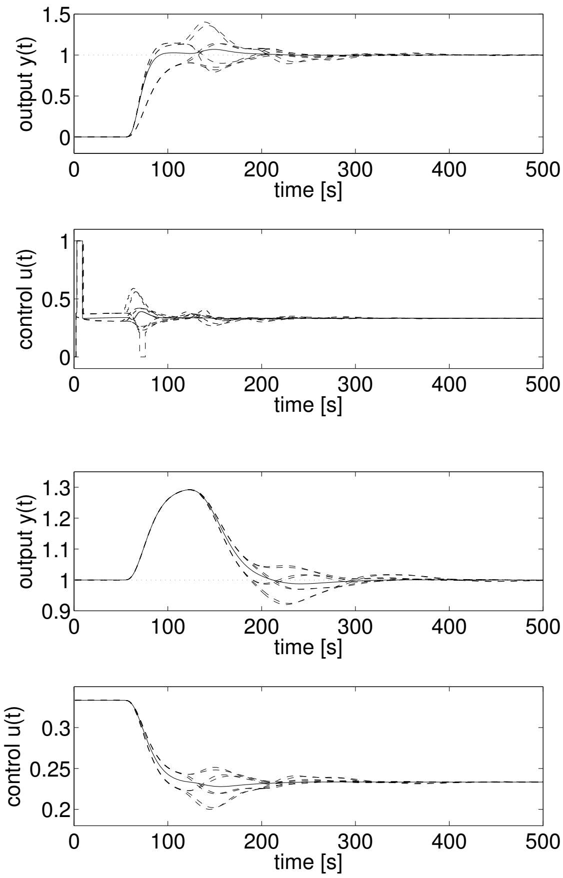

Fig. 8. The setpoint step change response and input step disturbance rejection, perturbed FOPDT approximations used for prediction, $T_S = 1s$, $H_C = 5$, $N_1 = 50$, $N_2 = 150$, $\lambda = 0.18$, $N = 190$.

as was already mentioned. The robustness can be further improved by increasing the λ parameter.

4. CONCLUSION

The described pulse-step model predictive controller ensures high quality behavior of the closed control loop. It keeps and exploits the manipulated value constraints while the computational cost is kept at a reasonable level. The tuning of the controller is very easy, as it has only 3 parameters (except the step response): the control horizon, the prediction horizon, and the weighting coefficient λ. Only the last one is meant for manual tuning of the controller, the others are determined automatically from the step response sequence. It was illustrated that this controller can also deal with model uncertainties very well. All this makes the PSMPC controller a suitable candidate for the PID controller successor in industry. The PSMPC controller has been recently added to the Matlab/Simulink compatible RexLib function block library, which is available for open public and whose general description can be found in Balda et al. [2005].

ACKNOWLEDGEMENTS

This paper was supported by the Ministry of Industry and Trade of the Czech Republic, grant no. FI-IM3/037. This support is gratefully acknowledged.

REFERENCES

P. Balda, M. Schlegel, and M. Štětina. Advanced control algorithms + Simulink compatibility + Real-time OS = REX. In *Preprints of the 16th IFAC World Congress*, Prague, Czech Republic, 2005.

S. Huang, K.K. Tan, and T.H. Lee. *Applied predicitive control*. Springer-Verlag, London, 2002.

J.M. Maciejowski. *Predictive control with constraints*. Prentice Hall, Englewood cliffs, NJ, 2002.

J. Richalet, S. Abu el Ata-Doss, C. Arber, and H.B. Kuntze. Predictive functional control: application to fast and accurate robots. In *Proceedings of the 10th IFAC World Congress*, Munich, Germany, 1987.

M. Schlegel and M. Čech. Internet PID controller design: www.pidlab.com. In *Preprints of Second IFAC Workshop on Internet Based Control Education IBCE04*, pages 1–6, Grenoble, France, 2004. ENSIEG.

M. Schlegel and O. Večerek. Robust design of Smith predictive controller for moment model set. In *Preprints of the 16th IFAC World Congress*, Prague, Czech Republic, 2005.

P. Tondel and T.A. Johansen. Complexity reduction in explicit model predictive control. In *IFAC World Congress Proceedings*, Barcelona, Spain, 2002.

A. Wallén and K.J. Åström. Pulse-step control. In *Proceedings of 15th Triennial World Congress*, Barcelona, Spain, 2002.