Receding Horizon Optimal Control of Wiener Systems by Application of an Asymmetric Cost Function

Daniel Alberer*, Harald Kirchsteiger*, Luigi del Re*, Hans Joachim Ferreau, Moritz Diehl****

**Institute for Design and Control of Mechatronical Systems, Johannes Kepler University Linz, Austria (e-mail: daniel.alberer@jku.at, harald.kirchsteiger@jku.at, luigi.delre@jku.at)*
***Department of Electrical Engineering, K.U. Leuven, Belgium (e-mail: joachim.ferreau@esat.kuleuven.be, moritz.diehl@esat.kuleuven.be)*

Abstract: Wiener models are an important class of nonlinear systems which well approximate many applications. Real time optimal control of Wiener models, for instance in the form of receding horizon optimal control, can be done using the nonlinear setup and corresponding nonlinear optimization tools. However, as this paper shows, under rather mild conditions on the static nonlinearity, it is possible to re-formulate the optimal control problem as a linear problem with an asymmetric cost function, whose solution can be computed using a slack variable extension of the initial quadratic problem with a small additional computational cost. This paper shows the approach and the achievable performance at the example of the emission control of a large gas engine used in the U.S. pipeline network.

Keywords: Receding horizon optimal control, Wiener model, asymmetric cost function, slack variables

1. INTRODUCTION

Wiener systems are a well known class of nonlinear systems consisting of a linear dynamics followed by a static nonlinear map. Due to their simple form, they have attracted much interest especially as a way to approximate more complex systems, see *e.g.* (Norquay, *et al.*, 1999) and references therein. Accordingly, the identification of Wiener and Hammerstein models has also attracted much attention, see *e.g.* (Guo, 2003; Pearson and Pottmann, 2000; Vandersteen and Schoukens, 1997).

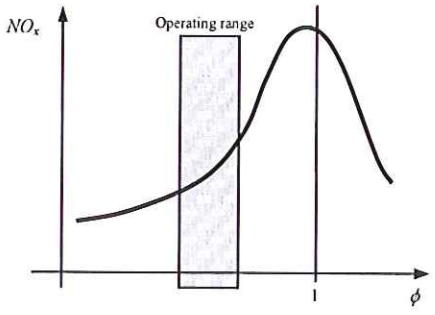


Fig. 1: NO_x as a function of ϕ

From a practical point of view, there are two different interpretations of Wiener models. The nonlinear static map can be interpreted as a measurement characteristic, implying the output of the linear part to be the real target value and the nonlinear element to be the deformation due to the sensor. In such a case, the nonlinear map can usually be inverted and the system treated to almost all purposes as a linear one. However, there are other cases in which the real target value is the output of the nonlinearity, so any linearization would induce substantial errors.

Consider, for example, the application behind this paper, *i.e.* the NO_x control of stationary operating gas engines used in pipeline compressor stations: the fuel/air ratio ϕ of these engines can be described rather well by linear models, but the real target quantity is the value of the exhaust NO_x concentration. Due to the limited speed and temperature range of these engines and the lean operation, it turns out that the relation between NO_x and ϕ can be approximated by a static map as shown in Fig. 1. It is evident that optimal control of ϕ is not equivalent to the optimal control of the NO_x concentration.

Receding horizon optimal control, in particular in the form of model predictive control, has been the topic of a huge number of publications and of many years of successful experience (Camacho and Bordons, 2004; Rawlings, 1999; Scokaert and Mayne, 1998). These methods have also been applied to the automotive field, see *e.g.* (Falcone, *et al.*, 2008; Ortner and del Re, 2007).

Still, very few works have been centered on the optimal control of Wiener models, some exceptions being a work on its application to a pH neutralization experiment (Norquay, *et al.*, 1999), or (Pérez, *et al.*, 2006) who consider the application of Wiener and Hammerstein models to predictive control of a Diesel engine airpath. Both have in common, that they only take into account the linear part in the optimal controller. (Bloemen, *et al.*, 2001) determine bounds for the nonlinearities and then apply a robust MPC approach. Recapitulatory, in all these cases a linear problem is solved instead of the original one.

To better understand the problem, recall that a discrete-time optimal control problem comprising linear system dynamics, a quadratic cost function and (possibly) linear constraints is

well-known to be a quadratic programming problem (QP). Moreover, this QP is strictly convex – and thus has a unique solution – whenever the cost function is strictly convex. Convex QPs can be solved very efficiently and reliably using dedicated QP solvers.

Unfortunately, when system dynamics are given in form of a Wiener system, this valuable structure is lost. In fact, without additional assumptions on the nonlinear static output map, the optimal control problem will result in a general, and usually non-convex, nonlinear programming problem (NLP). This loss of structure increases computational load for solving these problems significantly (easily by one or more orders of magnitude) and, due to the lack of convexity, the solver might even get stuck in a local minimum. While keeping convexity for systems with nonlinear dynamics is almost hopeless (see (Azhmyakov and Raisch, 2008) for a few limited exceptions), this property remains for optimal control problems comprising a Wiener system if the static map satisfies certain conditions, as we will point out in the next section. If they are met, the resulting optimal control problem is equivalent to a convex NLP, where every local solution is also a global one. In such a situation, we propose substituting this convex NLP by a convex QP using an asymmetric cost function. In doing so, computational effort is greatly reduced while the nonlinearities of the Wiener systems are still taken into account, thus improving performance over a purely linear model.

2. MOTIVATION OF AN ASYMMETRIC COST FUNCTION

2.1 MPC with Symmetric Cost Function

In receding horizon control the objective is formulated at every time instant by use of the system description. Frequently the objective is to minimize the squared deviations from a given reference and to keep the control action small. The determination of the control signal Δu can then be formulated as the following optimization problem (with positive definite matrices Q_y and R):

$$\begin{aligned}
&\arg\min_{\Delta u} \frac{1}{2}\sum_{i=0}^{n_{PH}} \left(y_i - y_{ref,i}\right)^T Q_y \left(y_i - y_{ref,i}\right) + \Delta u_i^T R \Delta u_i \\
&\textit{subject to} \\
&\quad u_i = u_{i-1} + \Delta u_i \\
&\quad x_{i+1} = f(x_i, u_i) \\
&\quad y_i = h(x_i) \\
&\quad \underline{y} \le y_i \le \overline{y} \qquad i = 0 \ldots n_{PH} \\
&\quad \Delta \underline{u} \le \Delta u_i \le \Delta \overline{u} \quad i = 0 \ldots n_{CH} - 1 \\
&\quad \Delta u_i = 0 \qquad i = n_{CH} \ldots n_{PH}
\end{aligned} \tag{1}$$

In (1) the predicted system output y (y_i is actually equal to $y(t_0 + t_i \mid t_0)$) is used to determine the optimal input Δu over the control horizon with length n_{CH} that minimizes the objective over the prediction horizon with length n_{PH}. In case of a linear state space representation

$$\begin{aligned} x_{i+1} &= A \cdot x_i + B \cdot u_i \\ y_i &= C \cdot x_i \end{aligned}, \tag{2}$$

formulation (1) simplifies to the standard linear MPC setup. In this case the optimization problem can be stated in the form of a quadratic problem

$$\begin{aligned} \min_{\Delta u} \quad & \frac{1}{2}\Delta u^T H \Delta u + \Delta u^T g \\ \text{s.t.} \quad & lbG \le G\Delta u \le ubG \\ & lb \le \Delta u \le ub \end{aligned}. \tag{3}$$

The matrices H and G are constant, whereas the gradient g and the constraint vectors need to be updated every time instant. As mentioned before, problems in the form of (3) can be solved efficiently by numerical QP solvers, *e.g.* using the online active set strategy as proposed in (Ferreau, *et al.*, 2008) for the MPC context. If the system dynamics $f(x_i, u_i)$ or the output function $h(x_i)$ is nonlinear, optimization problem (1) cannot be stated as a QP anymore and thus a nonlinear solver is required to solve the resulting generic NLP. This does not only increase the online computational load significantly, also the implementation and software maintenance effort of an NLP solver is typically higher than that of a simpler QP solver. In order to circumvent these undesirable complications, the next section presents a class of nonlinear systems where the optimization problem (1) can be reformulated to make it suitable for QP solvers.

2.2 Asymmetric Cost Function for a Class of Wiener Systems

In general, Wiener systems consist of a linear dynamic block followed by a static nonlinearity.

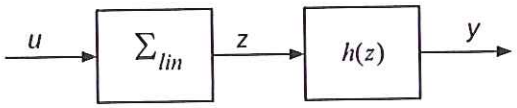


Fig. 2: Wiener system

In a state-space representation the system can be described by

$$\begin{aligned} x_{i+1} &= A \cdot x_i + B \cdot u_i \\ z_i &= C \cdot x_i \\ y_i &= h(z_i) \end{aligned}, \tag{4}$$

where we make the standing assumption that h is invertible. Still, due to the nonlinearity $y_i = h(z_i)$ a nonlinear solver is required for the solution of problem (1) for Wiener systems. In theory, regulating the system to a constant setpoint $z_{ref} = h^{-1}(y_{ref})$ is equal to regulating the nonlinear system to y_{ref} for steady state. However, in case of actuator and/or system limits as well as model-plant mismatches, a perfect tracking might not be possible and thus minimizing the deviations to z_{ref} is not equal to minimizing the deviations to y_{ref}. In other words, if a tracking of the linear system part is done, than the objective needs to be changed to maintain the initial performance requirements.

Now we present an approach where for a certain class of static nonlinearities the tracking of the output y is transferred to a tracking of z with a modified objective function, such that it still can be reasonably formulated as a QP. As these QPs will be convex, we believe that this reformulation makes most sense, if the NLP resulting from (1) comprising a Wiener system (4) is convex, too. Mostly for notational convenience we restrict our presentation to systems comprising only one output (MISO). However, for the typical setup where the cost function matrix Q_y is diagonal and the static output map is given separately for each output component, it is straightforward to extend these ideas to MIMO systems.

Lemma 1: Let an MPC problem (1) comprising a Wiener system (4) with exactly one output be given. If the scalar static output map $h(z)$ is (i) twice continuously differentiable and (ii) $\frac{\partial|h(z)|}{\partial z}$ is monotonically increasing, then the resulting NLP is convex over the domain of Δu.

Proof: It is sufficient to prove convexity of $h(z)^2$, as all other summands of the cost function of (1) are obviously convex, This is implied by the non-negativity of its second derivative, which follows from condition (ii) as it ensures that $h(z)\cdot h''(z)\geq 0$ for all z. Finally, as z can be expressed as a linear transformation of Δu, the cost function is a composition of a convex function with an affine mapping (Boyd and Vandenberghe, 2004) and thus remains convex w.r.t. to Δu. □

The conditions of Lemma 1 are moderate and can be often met in practical settings like in our example presented in Section 4. Nonlinear output mappings that satisfy the conditions of Lemma 1 are *e.g.* $h(z)=z^3$ or $h(z)=e^z-1$.

2.3 Exact Asymmetric Cost Function for Wiener Systems with Piecewise Affine Output Mapping

One interesting class of output mappings for which Lemma 1 can easily be extended, is the subclass of piecewise affine functions for which condition (ii) holds. An example is depicted in Fig. 3, formally they can be expressed as

$$h(z)=\begin{cases} k_{-n_{k-}}\cdot z+d_{-n_{k-}} & z\in Z_{-n_{k-}} \\ k_{-n_{k-}+1}\cdot z+d_{-n_{k-}+1} & z\in Z_{-n_{k-}+1} \\ \quad\cdots & \\ k_0\cdot z & z\in Z_0 \\ k_1\cdot z+d_1 & z\in Z_1 \\ \quad\cdots & \\ k_{n_{k+}}\cdot z+d_{n_{k+}} & z\in Z_{n_{k+}} \end{cases}, \qquad (5)$$

where we assume without loss of generality that the interval Z_0 starts at 0. If this condition is fulfilled, then the quadratic cost function for the tracking of y can be transformed to a multi-quadratic cost function for the tracking of z, *i.e.* several quadratic cost functions are superposed to generate an asymmetric cost function. In that case, a symmetric quadratic function is joined with asymmetric parts as shown in Fig. 4.

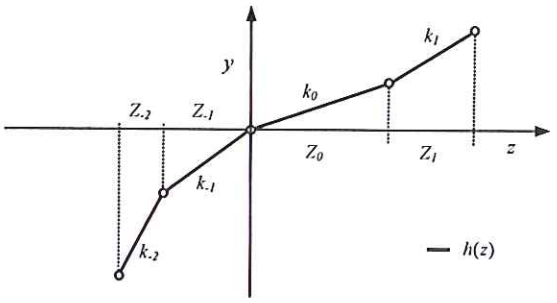


Fig. 3: Piecewise affine function $h(z)$

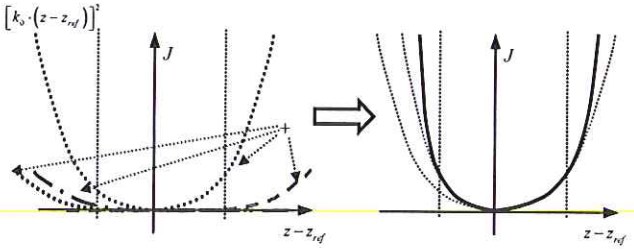


Fig. 4: Asymmetric cost function

Consequently, the quadratic nature is retained; however by applying different weightings the cost function becomes asymmetric. In other words, Wiener systems comprising such a piecewise affine output mapping can be exactly transferred to a convex QP with an asymmetric cost function. It has to be underlined that both descriptions are equivalent.

If the output nonlinearity $\tilde{h}(z)$ is not piecewise affine, it can often be approximated by a suitable piecewise affine function $h(z)$ as depicted in Fig. 5. Although by doing so the transformation does not yield the exact original problem, it can lead to an improved tracking behavior in practical applications like our application example. In the next section the corresponding mathematical description is presented

3. MATHEMATICAL FORMULATION

To make the approach suitable for a standard QP solver, a few steps are required. In the following we concentrate on the part of the cost function penalizing the output only

$$J_y=\frac{1}{2}\sum_{i=0}^{n_{FH}}\left(y_i-y_{ref,i}\right)^T Q_y\left(y_i-y_{ref,i}\right), \qquad (6)$$

since the penalization of the input remains the same independent on the output nonlinearity. The conditions of Lemma 1 imply that one of the function parts that meet at the origin (*i.e.* those that are defined over the intervals Z_{-1} and Z_0 in (5)) has the lowest absolute gain value of the piecewise affine function $h(z)$. Without loss of generality let it be k_0, then the symmetric quadratic part $\left[k_0\cdot\left(z-z_{ref}\right)\right]^2$ builds the basis of the QP formulation (see Fig. 4). Taking asymmetric quadratic terms into account (in the QP) requires the introduction of slack variables s which are weighted by ρ.

$$J_z = \frac{1}{2}\left[\sum_{i=0}^{n_{PH}} \left(z_i - z_{ref,i}\right)^T Q_z \left(z_i - z_{ref,i}\right) + \sum_{i=0}^{n_{SH}} \sum_{\substack{j=-n_{k-} \\ j\neq 0}}^{n_{k+}} s_{i,j}{}^T \rho_j s_{i,j}\right] \quad (7)$$

The contribution of the slacks over the slack horizon (n_{SH}) has to be chosen such that J_z becomes equal to J_y. This condition inherently includes that the convexity of the problem is retained. In the QP framework conditions have to be implemented by inequalities. Hence, the according inequalities for the slack variables are (for the case $k_j \geq 0$, the other case works analogously):

$$\left.\begin{aligned} & s_{i,j} \geq 0 \\ & s_{i,j} \geq \left(k_j - \sum_{w=0}^{j-1} k_w\right)\cdot\left(z_i - z_{ref,i}\right) + d_j - \sum_{w=0}^{j-1} d_w \end{aligned}\right\} j > 0$$

$$\left.\begin{aligned} & s_{i,j} \geq 0 \\ & s_{i,j} \geq \left(k_j - \sum_{w=j+1}^{0} k_w\right)\cdot\left(z_i - z_{ref,i}\right) + d_j - \sum_{w=j+1}^{0} d_w \end{aligned}\right\} j < 0 \quad (8)$$

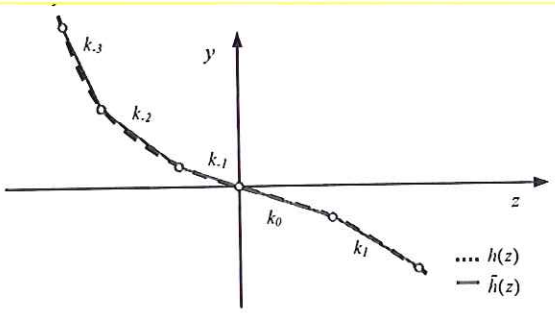


Fig. 5: Piecewise affine approximation $h(z)$ of the nonlinear function $\tilde{h}(z)$

The slacks in the optimization problem are treated as optimization variables. Since the initial QP (3) is simply extended with a diagonal matrix

$$\min_{\Delta u, s} \frac{1}{2}\begin{bmatrix}\Delta u^T & s^T\end{bmatrix}\begin{bmatrix}H & 0 \\ 0 & \rho\end{bmatrix}\begin{bmatrix}\Delta u \\ s\end{bmatrix} + \begin{bmatrix}\Delta u^T & s^T\end{bmatrix}\begin{bmatrix}g \\ 0\end{bmatrix}, \quad (9)$$

the determination of the slacks is computationally less expensive than *e.g.* an increase of n_{CH} by the same dimension. However, the QP dimension increases and so the computation time, typically quadratic to cubic in the number of slacks. Ideally the slack extension is done for every time instant of the prediction horizon. However, the number of slacks can quickly become high if many affine pieces are used and more than one output needs to be weighted asymmetrically. Thus, for practical applications the size of n_{SH} is a tuning parameter similar to n_{PH} and n_{CH}. The weightings Q_z and ρ_j in (7) are defined by

$$\begin{aligned} & Q_z = k_0{}^2 \cdot Q_y \\ & \rho_j = Q_y, \quad j = -n_{k-} \ldots n_{k+}, \; j \neq 0 \end{aligned} \quad (10)$$

Obviously, if the piecewise affine function h is defined for absolute values of z and not for the deviation to the reference z-z_{ref}, (7)-(10) need to be recalculated in case of a setpoint change.

4. APPLICATION EXAMPLE

To show the potential of the approach, it was applied to a simulation model of a real plant. In the next subsections the system is briefly described as well as the target is presented. Then the results of the new approach are compared to a standard MPC for the linear system.

4.1 System Description

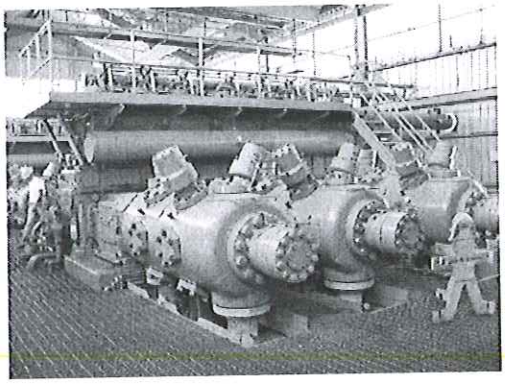


Fig. 6: Integral Engine

The considered system (Fig. 6) consists of a turbocharged 2 stroke reciprocating gas engine with counterflow scavenging and a reciprocating compressor both acting on the same crankshaft, also called "integral engine". For the current analysis a model of a Clark TLA 6 was used, with 6 combustion and 3 compressor cylinders. Typically the engine is operated in a narrow range and has four main controls:

- The engine speed is controlled by the injected fuel amount (through the injection pressure), GFC (governor fuel command) represents the valve position of the fuel supply
- The fuel to air ratio ϕ is controlled with the turbocharger wastegate (WG)
- The ignition timing is usually kept constant
- The power output is adjusted with the compressor load by adding or removing discrete amounts of compressor clearances (pockets).

The pockets can be seen in Fig. 6, where 2 small pockets are mounted in "V-shape" at each compressor cylinder and one big at the front. In standard operation the pockets are used to maintain almost constant engine power according to the slowly changing pipeline conditions. The considered integral engine has 6 small and 3 big pockets which results in 28 load stages, also called load steps (LS). The steady state values of two adjacent LS differ by approximately 3%. However, due to imperfect synchronization of the pocket switching, substantial peaks arise during the transition at certain LS changes. The standard control configuration uses simple SISO PIDs, although in particular ϕ has a strong coupling with the injected fuel amount. Former work showed that a model based MIMO control increases the control performance significantly. For more details see (Alberer, *et al.*, 2008; Huschenbett, *et al.*, 2007).

4.2 Implementation

A linear MIMO grey box system was designed – GFC, WG and the compressor torque (T_{Comp}) were used as inputs and engine speed as well as ϕ as outputs. Physical knowledge (*e.g.* the crank shaft is an integrator) was combined with data based identification (*e.g.* for the time constant of the fuel valve) to design a combined MIMO model. Although the several dynamics were modeled either with first or second order elements, due to delay times the model order increased to 22 states. Adding the nonlinear relation between ϕ and NO_x (Fig. 1) to the output of the linear system finally results in a Wiener representation.

The target is to keep NO_x below a given threshold and to stabilize the engine speed. Since no catalytic aftertreatment devices are applied, the engines are always operated in the lean range (see Fig. 1), *i.e.* only a narrow part of the NO_x-ϕ relation needs to be considered. The nonlinear function $NO_x = \tilde{h}(\phi)$ had to be approximated by a piecewise affine function $h(\phi)$. Unfortunately $\frac{\partial|\tilde{h}(\phi)|}{\partial\phi}$ does not fulfill the necessary condition to be monotonically increasing.

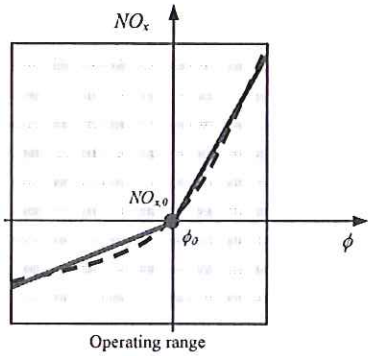


Fig. 7: Multi linear approximation of $NO_x = \tilde{h}(\phi) \approx h(\phi)$

However, in the real application we use $h(\phi)$ instead of $\tilde{h}(\phi)$, thus the condition can be relaxed to $\frac{\partial|h(\phi)|}{\partial\phi}$ monotonically increasing. The consequence of this condition was that for negative deviations with respect to the setpoint we were restricted to apply only one affine function. A single affine function for positive deviations proved sufficient too (see Fig. 7).

4.3 Results

The online MPC was designed with the linear system plant, a prediction horizon of 300 and a control horizon of 7 (t_s = *0.1s*). In Fig. 8 the effect of the slack extension on the objective J_{NOx} can be seen as well as the increase of the computation time. Obviously, the maximum improvement of the cost function depends on the settings of n_{CH} and n_{PH} too, whereas the computation time typically grows quadratically with n_{SH}. For our application $n_{SH} = 50$ was considered to be the best compromise. In Fig. 9 and Fig. 10 a MPC with symmetric cost function for ϕ is compared to the asymmetric version.

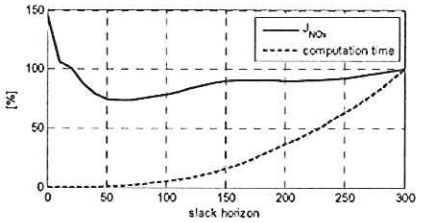


Fig. 8: NO_x objective and computation time as function of n_{SH}

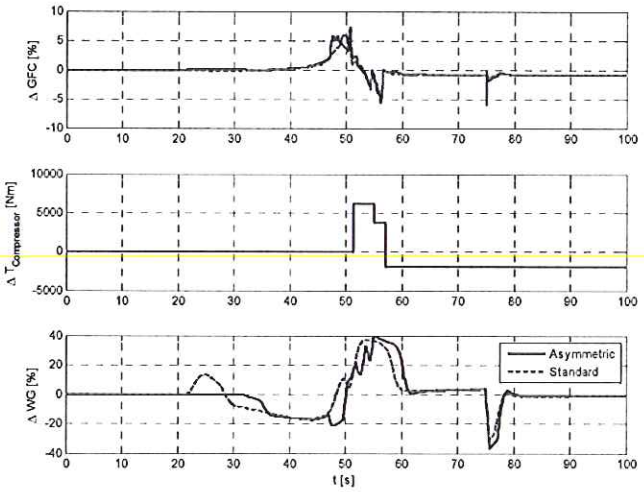


Fig. 9: Comparison of system inputs, MPC with symmetrical and asymmetrical cost function for ϕ

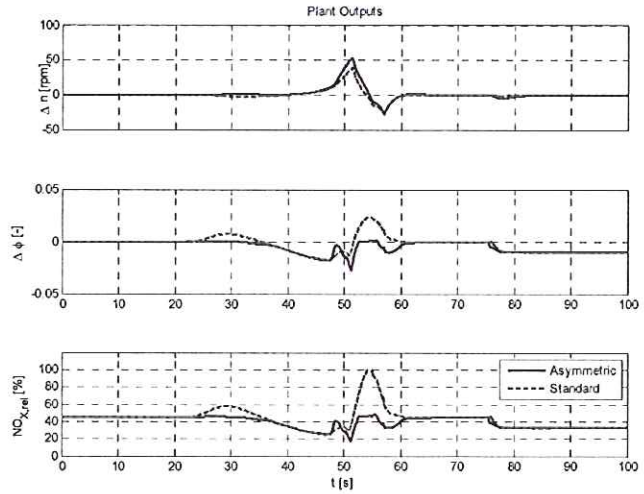


Fig. 10: Comparison of system outputs, MPC with symmetrical and asymmetrical cost function for ϕ

At t=50s a load step change 7 to 8 was done with high amplitudes on the compressor torque. Due to the future measured disturbance implementation, the MPC already began to react on the LS change 30s in advance. A perfect regulation of both outputs was not possible, thus the objective was adjusted such that the focus primarily lied on ϕ. Obviously, the standard MPC with symmetrical weighting on ϕ shows an almost symmetric distribution of the ϕ-deviations.

On the other hand the modified objective causes the engine to run tendentiously lean and thus reduces the NO_x peaks significantly.

5. CONCLUSIONS

For a certain class of nonlinear functions of Wiener systems, namely piecewise affine functions with some restrictions, the control problem of the output y could be restated to a control of the linear system output z with a modified objective, *i.e.* controlling z is equivalent to controlling y provided that the convex symmetric cost function for y is changed to a convex asymmetric cost function for z. The main advantage over solving the general nonlinear problem is that efficient numerical solvers can be applied. For a certain weighting of the nonlinear problem, the slack weights are determined directly by the gains of the piecewise affine function $h(z)$.

If the general Wiener nonlinearity $\tilde{h}(z)$ can be approximated by a suitable piecewise affine function $h(z)$, then the control can be reformulated too. Hence, for many practical applications the approximate solution might be sufficient to increase the control performance compared to a purely linear control without using a general nonlinear solver.

In the practical example it was shown that the procedure apparently works for nonlinear functions that do not fulfill the necessary condition, too. Since the emission values were reduced significantly in simulation, the approach is going to be tested on the real plant against the standard approach. The satisfactory simulation results let us be confident that the implementation on the real system is going to bring valuable results, too.

Obviously, the extension with slack variables increases the dimension of the problem. However, the simulations showed that this extension is not as computationally expensive than increasing the control horizon. As already presented in the simulation example, the slack horizon n_{SH} is a tuning parameter and needs to be optimized for every application with respect to the computation time.

6. ACKNOWLEDGEMENTS

The authors gratefully thank the ACCM Austrian Center of Competence in Mechatronics, Hoerbiger Compression Technology and the Pipeline Research Council International (PRCI), which financially supported the work. The fourth author holds a PhD fellowship of the research foundation – Flanders (FWO). Research of the last two authors has also been supported by Research Council KUL: CoE EF/05/006 Optimization in Engineering Center (OPTEC), and by Belgian Federal Science Policy Office: UAP P6/04 (Dynamical systems, control and optimization, 2007-2011).

7. REFERENCES

Alberer, D., M. Ranzmaier, L. del Re and M. Huschenbett (2008). MIMO Model Predictive Control for Integral Gas Engines under Switching Disturbances. In: *17th IEEE International Conference on Control Applications*, San Antonio, Texas, USA.

Azhmyakov, V. and J. Raisch (2008). Convex Control Systems and Convex Optimal Control Problems With Constraints. *IEEE Transaction on Automatic Control*, 53, 993-998.

Bloemen, H. H. J., T. J. J. v. d. Boom and H. B. Verbruggen (2001). Model-based predictive control for Hammerstein–Wiener systems. *International Journal of Control*, 74, 482-495.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*, Cambridge University Press, Cambridge, United Kingdom.

Camacho, E. F. and C. Bordons (2004). *Model Predicitve Control*, Springer, London.

Falcone, P., F. Borrelli, H. E. Tseng, J. Asgari and D. Hrovat (2008). Linear time-varying model predictive control and its application to active steering systems: Stability analysis and experimental validation. *International Journal of Robust Nonlinear Control*, 18, 862-875.

Ferreau, H. J., H. G. Bock and M. Diehl (2008). An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, 18, 816-830.

Guo, F. (2003). A New Identification Method for Wiener and Hammerstein Systems. In: *Fakultät für Maschinenbau*, Karlsruhe, Karlsruhe.

Huschenbett, M., D. Alberer, G. Beshouri and M. Richter (2007). Emission Reduction & Stability Improvement by Model Based Predictive Control of Legacy Gas Engines. In: *Gas Machinery Conference 2007*, Dallas, Texas.

Norquay, S. J., A. Palazoglu and J. A. Romagnoli (1999). Application of Wiener Model Predictive Control (WMPC) to a pH Neutralization Experiment. *IEEE Transactions on Control Systems Technology*, 7, 437-445.

Ortner, P. and L. del Re (2007). Predictive Multivariable Control of a Diesel Engine Air Path. *IEEE Transactions on Control Systems Technology (Control Applications in Automotive Engineering)*, 15, 449-456.

Pearson, R. K. and M. Pottmann (2000). Gray-box identication of block-oriented nonlinear models. *Journal of Process Control*, 10, 301-315.

Pérez, E., X. Blasco, S. García-Nieto and J. Sanchis (2006). Diesel Engine Identification and Predictive Control using Wiener and Hammerstein Models. *IEEE, International Conference on Control Applications*. Munich, Germany.

Rawlings, J. B. (1999). Tutorial: Model Predictive Control Technology. *American Control Conference*, 662-676. San Diego, California.

Scokaert, P. O. M. and D. Q. Mayne (1998). Min-max feedback model predictive control for constrained linear systems. *IEEE Transaction on Automatic Control*, 43, 1136 - 1142.

Vandersteen, G. and J. Schoukens (1997). Measurements and Identification of Nonlinear Systems Consisting of Linear Dynamic Blocks and one Static Nonlinearity. *IEEE Transactions on Automatic Control*, 44, 1266-1271.