Proceedings of the 45th IEEE Conference on Decision & Control
Manchester Grand Hyatt Hotel
San Diego, CA, USA, December 13-15, 2006

MPC for Large-Scale Systems via Model Reduction and Multiparametric Quadratic Programming

S. Hovland, K. Willcox and J. T. Gravdahl

Abstract— **In this paper we present a methodology for achieving real-time control of systems modeled by partial differential equations. The methodology uses the explicit solution of the model predictive control (MPC) problem combined with model reduction. The explicit solution of the MPC problem leads to online MPC functionality without having to solve an optimization problem at each time step. Reduced-order models are derived using a goal-oriented, model-based optimization formulation that yields efficient models tailored to the application at hand. The approach is demonstrated for reduced-order output feedback control of a large-scale linear time invariant state space model of the discretized heat equation.**

I. Introduction

Distributed systems such as those representing flow dynamics are modeled mathematically by a set of partial differential equations (PDEs). In order to simulate such systems, the PDEs are usually discretized using a set of tools known as computational fluid dynamics (CFD). For many applications, CFD models are often linearized around steady-state operating points, leading to linear systems of potentially very high order. Although often acceptable for simulation purposes, the high order of the discretized (and linearized) models may be prohibitive for controller design. Model-order reduction has emerged as a powerful tool to address such issues. The recent textbook [3] provides a comprehensive overview of the field.

Model predictive control (MPC) is a control strategy that has been widely accepted in the industrial process control community and implemented successfully in many commercial applications. The greatest strength of MPC is the intuitive way in which constraints can be incorporated in a multivariable control problem formulation. However, the traditional MPC strategy demands a great amount of online computation, limiting the use of these kinds of controllers to processes with relatively slow dynamics, since an optimization problem (often a constrained quadratic program) is solved at each samplingtime step. It has recently been shown that much of the computational effort in traditional MPC can be done offline. In [7] and [26] the authors present algorithms for solving multiparametric quadratic programs (mpQPs) that are used to obtain explicit solutions to the MPC problem. Thus, the explicit model predictive controller (eMPC) accomplishes online MPC functionality without solving an optimization problem at each time step. This has several advantages: 1) The online computational time can be reduced to the microsecond-millisecond range, and 2) MPC functionality is achieved with low complexity, easily verifiable real-time code, justifying the employment of eMPC in embedded and safety-critical systems. In [13], the authors use eMPC for attitude control for spacecraft.

Several authors have considered model-order reduction for control of large-scale systems, among them [16], [5], [21], [6], [2]. In [14] the authors demonstrate that model predictive control based on a linear reduced-order model derived from a CFD model using proper orthogonal decomposition performs well for the control of an industrial glass-feeder. It is stated that in order to use the reduced-order simulation models for that purpose, the models should be at least 100 times faster than real time. In order to achieve this challenging goal, very low order models that target the control problem at hand are needed. The methodology presented in this paper deals with this problem in two ways. First, our model reduction methodology yields reduced models that are appropriate for use in controller design. Second, eMPC gives a further reduction in computational requirements compared to MPC. Thus, eMPC facilitates implementation of constrained optimal control in applications that are described by models of high order, while being characterized by fast sampling or low cost, such as mechatronics, MEMS, rotating machinery and acoustics; for instance for active control of combustion instability. Here, we demonstrate the methodology for output regulation of the discretized heat equation.

Throughout the paper, positive (semi-) definiteness for matrices is indicated by $\succ 0$ ($\succeq 0$) and superscript * indicates complex conjugate transpose. The $\mathcal{H}_2$-norm of a linear time-invariant system is denoted by $\|\cdot\|_{\mathcal{H}_2}$.

II. Model-Order Reduction

The goal of model reduction is to derive a model of low order that preserves the input-output behavior of the high-fidelity model. In addition, one may wish to preserve specific properties of the high-fidelity model, such as stability, passivity etc. In the control community, algorithms such as optimal Hankel model reduction [1], [9], [15] and balanced truncation [19] are known to have strong guarantees on the quality of the reduced model by providing upper bounds for the approximation error. Although attempts have been made to extend these algorithms to large-scale settings ([24], [12], [17], [8]), model reduction of very large-scale models with rigorous guarantees on quality remains a challenge.

S. Hovland and J. T. Gravdahl are with the Department of Engineering Cybernetics, the Norwegian University of Science and Technology, 7491 Trondheim, Norway {svein.hovland, tommy.gravdahl}@itk.ntnu.no

K. Willcox is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA kwillcox@mit.edu

1-4244-0171-2/06/$20.00 ©2006 IEEE.

One model reduction method that has been used with considerable success in large-scale settings is the proper orthogonal decomposition (POD). POD is popular due to its simplicity and because it is applicable to very large-scale and nonlinear models. However, there are several limitations associated with using POD; in particular, POD-based reduced models lack the quality guarantees of those derived using more rigorous methods such as balanced truncation. Even in the case of stable linear time invariant (LTI) systems, reduction via POD can lead to undesirable and unpredictable results, such as unstable reduced models.

In [28], a goal-oriented model-based reduction algorithm was proposed. This methodology is targeted at applications in optimal control and optimal design, and addresses some of the limitations of POD. In this paper we compare the performance of eMPC using reduced-order models from these two projection-based model reduction algorithms; the POD method of snapshots [23] and the goal-oriented model-based reduction algorithm [28].

A. Model reduction by projection

For the order reduction of an LTI system,

$$\dot{x} = Ax + Bu \tag{1a}$$

$$y = Cx, \tag{1b}$$

where $x \in \mathbb{R}^n$ is the system state, $u \in \mathbb{R}^m$ contains the m inputs to the system, $y \in \mathbb{R}^p$ contains the p outputs and the constant coefficient matrices A, B, C are of appropriate dimensions, model reduction by projection works as follows. It is assumed that the state $x(t)$ can be approximated as a linear combination of r basis vectors

$$x \approx \Phi\hat{x}, \tag{2}$$

where $\hat{x} \in \mathbb{R}^r$ is the reduced state and $\Phi \in \mathbb{R}^{n\times r}$ is a projection matrix containing as columns the r basis vectors $\phi_1, \phi_2, \ldots, \phi_r$. Substituting $\Phi\hat{x}$ for x into (1), and requiring the resulting residual to be orthogonal to the space spanned by Φ, gives the reduced LTI state-space model

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u \tag{3a}$$

$$\hat{y} = \hat{C}\hat{x}, \tag{3b}$$

where $\hat{A} = \Phi^T A\Phi$, $\hat{B} = \Phi^T B$, $\hat{C} = C\Phi$ and $\hat{y}$ is the output of the reduced model. The POD and the goal-oriented model-based methodology both use the general projection framework just described; however, the two methods differ in the computation of the projection matrix Φ. In the method of snapshots the POD basis is found based on a set of snapshots $x(t_i)$, $i = 1, 2, \ldots M$, where M is the number of snapshots and $x(t_i)$ is the solution of (1a) at time t_i. The POD basis is computed as the set of left singular vectors of the snapshot matrix

$$X = [x(t_1)\ x(t_2)\ \cdots\ x(t_M)]\,. \tag{4}$$

For a given number of basis vectors, the POD basis minimizes the error $\mathcal{E}$ between the original snapshots and their representation in the reduced space, defined by

$$\mathcal{E} = \sum_{i=1}^{M} \left[x(t_i) - \tilde{x}(t_i)\right]^T \left[x(t_i) - \tilde{x}(t_i)\right], \tag{5}$$

where $\tilde{x}(t_i) = \Phi\Phi^T x(t_i)$. Consequently, POD only minimizes the approximation error for a set of data (defined by the snapshots) and no quality guarantees can be made regarding the reduced-order model that results from projection onto the POD basis.

B. Goal-Oriented Model-Based Reduction

In [28], a cost similar to that minimized in the POD procedure is used as an objective function in an optimization formulation. The key difference in that formulation is that the model-based optimization approach enforces the reduced-order governing equations as constraints. In addition, the cost is targeted to minimize the *output* error, while the POD minimizes the error of state prediction over the entire domain. The use of output error as a metric is motivated by a need for reduced models for optimal control and optimal design applications and, intuitively, might lead to different properties in output-feedback implementations.

The optimization problem in [28] can be formulated as

$$\min_{\Phi,\hat{x}} \frac{1}{2}\int_0^T (y-\hat{y})^T(y-\hat{y})\,dt + \frac{\beta}{2}\left(1-\phi_j^T\phi_j\right)^2 \tag{6}$$

subject to:

$$\begin{array}{rcl} \Phi^T\Phi\dot{\hat{x}} &=& \Phi^T A\Phi\hat{x} + \Phi^T Bu \\ \hat{y} &=& C\Phi\hat{x}, \end{array} \tag{7}$$

where the optimization problem seeks to find the rth-order basis $\Phi \in \mathbb{R}^{n\times r}$, and the corresponding reduced-order state solution $\hat{x} \in \mathbb{R}^r$ so that the 2-norm of the error between the full-order and reduced-order output is minimized. The full-order output $y(t)$ is obtained from simulating the high-fidelity model over a selected set of inputs and the interval $t \in [0,\ T\rangle$. The second term in the cost function (6) is a regularization term that penalizes the deviation of the length of the basis vectors from unity, with $\beta > 0$ as a regularization parameter. This regularization acts only in the null space of the projected Hessian matrix of the first term of (6). Therefore, the reduced output approximation, $\hat{y}$, is unaffected by the regularization term, yet the conditioning of the optimization problem is improved. Note, however, that there remains a null space of the projected Hessian matrix that admits arbitrary rotations of the basis vectors; the optimization method chosen to solve (6)–(7) should therefore be tolerant of singular projected Hessian matrices.

As described in [28] in a computationally efficient implementation of the method, the basis functions are assumed to be a linear combination of a finite collection of full-state snapshots. In this case, the number of optimization variables becomes Mr, where M is the number of snapshots and r is the dimension of the reduced state.

Model reduction for control is a challenging task, which is somewhat different from model reduction for simulation purposes. A reduced-order model that yields a good approximation of the high-fidelity model in open loop may not necessarily provide a good approximation in the closed loop, since the system dynamics change once the feedback loop is closed. One way to address this problem is to perform model reduction and control design iteratively, as was done in [4], in an attempt to approximate the closed-loop dynamics of the high-order model. Another common approach is to use frequency weighting in order to emphasize the importance of approximation quality in the bandwidth of the closed-loop system. Our goal-oriented model-based approach aims to create reduced models suitable for control applications, by targeting the projection basis to output functionals of interest, and by bringing additional knowledge of the reduced-order governing equations into the construction of the basis.

III. MPC VIA MULTIPARAMETRIC QUADRATIC PROGRAMMING

A brief outline of the standard MPC formulation will be given before we address the explicit solution. For further reading on MPC, the reader is advised to consult the textbook [18] or [22] for a tutorial.

A. A standard MPC formulation

The plant under consideration is modeled by a discrete-time LTI state space model

$$x_{k+1} = A_d x_k + B_d u_k, \tag{8}$$
$$y_k = C_d x_k, \tag{9}$$

where $k \in \mathbb{Z}$, and $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^m$ and $y_k \in \mathbb{R}^p$ denote the state, input and output, respectively, at step k. The matrices A_d, B_d and C_d are of appropriate dimension. The input u and output y are constrained by

$$y_{min} \leq y_k \leq y_{max}, \; u_{min} \leq u_k \leq u_{max} \tag{10}$$

for all $k > 0$. For the regulator problem the model predictive controller solves at time k the following optimization problem on the time horizon k to $k+N$:

$$\min_{U_k} \mathcal{J}_{mpc}(U_k, x_k), \text{ subject to} \tag{11}$$
$$u_{min} \leq u_{k+i} \leq u_{max}, i = 0, 1, \ldots, N-1 \tag{12}$$
$$y_{min} \leq y_{k+i} \leq y_{max} \;, \; i = 1, \ldots, N, \tag{13}$$

where $U_k = \begin{bmatrix} u_k & u_{k+1} & \ldots & u_{k+N-1} \end{bmatrix}^T$ is the sequence of future control inputs that yields the best predicted output with respect to the performance criterion on the prediction horizon N. The cost function is given by

$$\mathcal{J}_{mpc}(U_k, x_k) = x_{k+N}^T P x_{k+N} + \sum_{i=0}^{N-1} \left(x_{k+i}^T Q x_{k+i} + u_{k+i}^T R u_{k+i} \right), \tag{14}$$

where P, Q and R are weighting matrices of appropriate dimension. P and Q penalize deviation of the states x_{k+i} at the end of the prediction horizon and over the entire horizon, respectively, and R penalizes use of control action u. The first control input is applied to the process, before the whole optimization is repeated at the next sample. The optimization problem is then slightly different, having been updated by a new process measurement, a new starting point and an additional time slice at the end of the time horizon. The following assumptions are made.

Assumption 1: The cost matrices in (14) satisfy $P \succeq 0$, $R \succ 0$ and $Q \succeq 0$.

Assumption 2: (A_d, B_d) form a stabilizable pair, that is, all the unstable modes are controllable through $B_d u_k$.

B. Computing the control input

Sensitivity analysis is a technique used to describe how the solution to a mathematical program changes with small changes in the problem parameters. Closely related is parametric programming, where the solution is found explicitly for a range of parameter values. Mathematical programs that contain more than a single parameter are commonly referred to as multiparametric programs [25].

It is well established that implementing a model predictive controller requires solving a quadratic program (QP) in U_k at each time step, see e.g. [18]. With some manipulations, the problem in (11)-(13) can be written

$$\min_{U_k} \left\{ \frac{1}{2} U_k^T H U_k + x_k^T F U_k \right\} \tag{15}$$
$$\text{subject to: } G U_k \leq W + E x_k, \tag{16}$$

where the matrices H, F, G, W and E are functions of the weighting matrices P, Q, R and the bounds $u_{\min}$, $u_{\max}$, $y_{\min}$ and $y_{\max}$. If Assumption 1 is met, then $H \succ 0$ and the problem is strictly convex. The Karush-Kuhn-Tucker conditions (KKT) are then sufficient conditions for optimality, giving a unique solution U_k [20]. The problem (15)-(16) can be viewed as a multiparametric program in U_k, where x_k is a vector of parameters.

C. The explicit MPC formulation

By defining

$$z \triangleq U_k + H^{-1} F^T x_k, \tag{17}$$

the problem in (15)-(16) can be transformed into

$$\min_{z} \left\{ \frac{1}{2} z^T H z \right\} \tag{18}$$
$$\text{subject to: } G z \leq W + S x_k, \tag{19}$$

which is a multiparametric quadratic program in z, parametrized by x_k. The matrix S is found as $S = E + G H^{-1} F^T$. By considering the KKT conditions for the quadratic program in z, the solution z^* is seen to remain optimal in a neighborhood of x_k where the active set remains optimal. The region in which this active set remains optimal can be shown to be a polyhedron in the parameter space (that is, the state space) [7]. The mpQP in z can be solved offline for the state space area of interest. Computing the control input at a time step k then becomes a straightforward task: Given the system state x_k, the optimal control inputs U_k are obtained through an affine mapping,

$$u_k = K_i x_k + k_i, \tag{20}$$

where the subscript i denotes the ith affine function. K and k are constant within each polyhedral region in the parameter space. The online effort is thus reduced from solving a potentially large optimization problem at each time step for traditional MPC, to evaluation of a piecewise affine function of the current state. By implementing the piecewise affine function as a binary search tree, the online computational time is logarithmic in the number of regions in the state space partition [27].

IV. Reduced-order control

Implementing MPC or eMPC directly on the high-fidelity model is infeasible in large-scale settings, for instance when working with models obtained from CFD analysis. We will therefore use reduced-order control, where the reduced-order models will be used to design output-feedback controllers for the high-fidelity model.

The eMPC control input is computed based on the current state vector at every iteration. The reduced-order state must therefore be estimated in an observer or a Kalman filter. Here, we use a linear observer of the form[1]

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u + L\left(y - \hat{y}\right) \tag{21}$$

$$\hat{y} = \hat{C}\hat{x} \tag{22}$$

to estimate the reduced-order state. The linear observer should be designed such that the matrix $\left(\hat{A} - L\hat{C}\right)$ is Hurwitz.

The overall complexity of the proposed control scheme is given by the offline model reduction cost plus the cost of solving the eMPC problem offline for the reduced-order model. The former is determined by the number of optimization variables in the optimization problem (6)-(7), which is Mr. The cost of solving the eMPC problem increases with the number of parameters in the mpQP problem. Also, the memory required to store the eMPC solution online increases rapidly as the size of the solution grows. The scheme is therefore limited to cases where the reduced-order models can be made reasonably small, i.e. in the order of 10. Further complexity reduction techniques, such as input blocking, can be used to make the eMPC procedure more tractable in cases where the number of parameters (reduced-order states) is large.

V. Numerical Results

To investigate the implementation of the reduced-order control setup described above, we consider the ´heatcont' benchmark described in [10]. The benchmark is a single-input single-output LTI of the form (1) with 200 states, which are the temperatures at different locations in a thin rod, and it is based on spatial discretization of the one-dimensional heat diffusion equation. The input u is a heat source located at $1/3$ of the rod length, and the output y is the temperature recorded at $2/3$ of the length.

The procedure of finding the optimal basis that minimizes the criterion (6) is described in detail in [28], and is just mentioned briefly here. To solve the constrained optimization problem (6)-(7), we choose to eliminate the state variables $\hat{x}$ and state equations (7) and solve an equivalent unconstrained optimization problem in the Φ variables. The analytic gradient can be found through basic calculus of variations and use of adjoint variables, and an unconstrained optimization algorithm that uses a trust-region-based Newton method [11] can be used to determine the optimal basis. Since the optimization problem is nonlinear and nonconvex, it is important to generate a good initial guess. One possibility is to pick the POD basis as an initial guess. Alternatively, the initial guess for the case of r basis vectors can be chosen to be the solution of the optimization problem for $r-1$ basis vectors plus an arbitrary rth vector. This iterative procedure can be initialized at any value $r \geq 1$ with the POD basis vectors as an initial guess on the first iteration.

r	$\mathcal{H}_2^e$ for GOMBR	$\mathcal{H}_2^e$ for POD
1	0.6213	0.7959
2	0.0647	0.5023
3	0.0230	0.0692
4	0.0217	0.0627
5	0.02087	0.0841
6	0.02085	0.0742
7	0.0207	0.0468
8	0.0020	0.0020
9	0.0012	0.0012
10	8.6236×10^{-4}	38×10^{-4}

TABLE I

Assessment of reduced-order models of order 1 to 10. The reduced-order models with the optimized basis give a significant reduction in the relativ 2-norm of the error system, especially for low orders.

A. Open loop evaluation

Reduced-order models of order 1 to 10 are compared in Table I in terms of the relative $\mathcal{H}_2$ norm of the corresponding error systems, defined as

$$\mathcal{H}_2^e \triangleq \frac{\left\|\mathcal{G}\left(s\right) - \mathcal{G}_r\left(s\right)\right\|_{\mathcal{H}_2}}{\left\|\mathcal{G}\left(s\right)\right\|_{\mathcal{H}_2}}, \tag{23}$$

where $\mathcal{G}\left(s\right)$ and $\mathcal{G}_r\left(s\right)$ are the transfer functions of the full-order and reduced-order models, respectively. The reduced-order models are generated by comparing snapshots of the step response of the high-fidelity model at 20 time instants. It is seen that the goal-oriented model based reduction algorithm from [28] (labeled GOMBR in Table I) leads to a significant increase in approximation quality from POD in most cases for this metric, especially for low r. The goal-oriented basis is optimized with the POD basis as the initial guess in most cases. For $r = 4$, 5, 6 and 7 we use the iterative procedure discussed in Section II-A.

B. Closed loop evaluation

We consider the objective of regulating the output of the large-scale system to zero based on the reduced-order models. The controller weights are chosen to reflect this objective, by setting $Q = C^T\tilde{Q}C$, where $\tilde{Q} \in \mathbb{R}^{p\times p}$ is the weight on the output.

To compare the performance of the reduced-order models in closed loop, we first implement an output-feedback infinite horizon LQ-regulator based on the reduced-order models. The input from the LQ regulator is given by $u = -K\hat{x}$, where K is a constant feedback matrix, and $\hat{x}$ is the reduced state, in this case. The results are shown in Figure 1 and 2 for simulation of an optimized and a POD reduced-order model, respectively, with the same weights and $r = 3$. The figures clearly illustrate that the reduced-order model obtained with an optimized basis performs much better in closed loop than

[1] With a slight abuse of notation, we let $\hat{x}$ denote both the reduced state and its estimate.

the one with a POD basis, and emphasizes the observation from Table I, that the optimized reduced-order models give a better approximation, particularly for small r.

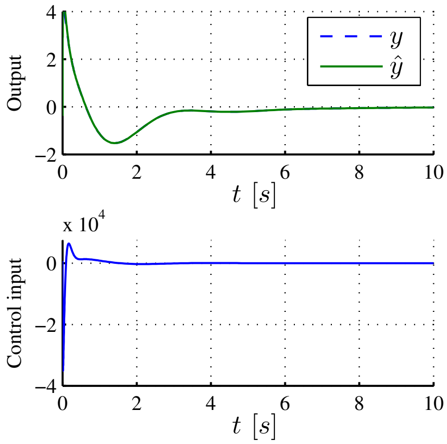

Fig. 1. Output-feedback LQ regulator for the high-fidelity model based on a reduced-order model with optimized basis for $r = 3$. Top: Estimated output from the reduced-order model $\hat{y}$ vs. output from the high-fidelity model y. Bottom: Control input.

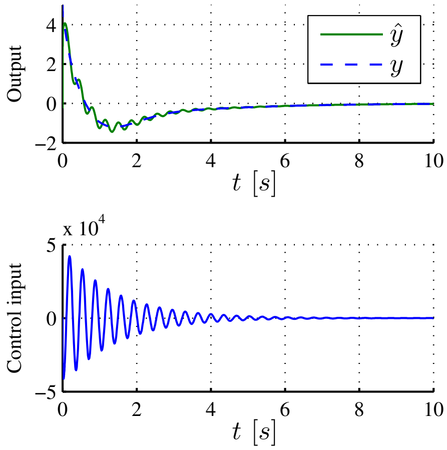

Fig. 2. Output-feedback LQ regulator for the high-fidelity model based on a reduced-order model with POD basis for $r = 3$. Top: Estimated output from the reduced-order model $\hat{y}$ vs. output from the high-fidelity model y. Bottom: Control input.

In real-world control problems there will always be some constraints on the state, input and/or output variables. To handle this, eMPC is a better choice than the unconstrained LQ regulator. To illustrate and visualize the setup, we first consider the case where $r = 2$, that is we have only 2 states in the reduced-order model. We set the prediction (and control) horizon $N = 2$. To demonstrate the controller's ability to enforce constraints, we constrain the control input such that $|u| < 1000$. First, the explicit solution to the MPC problem is solved in an offline phase for the relevant area of the reduced-order state space. This solution is used to control the high-fidelity model in the output-feedback setup described in Section IV. The system is initialized with a non-zero output. The resulting response is shown in Figure 3 for an optimized basis, where it is seen that the bound constraint on the control input is active during the first half second or so. It can also be observed that the output from the reduced-order model converges relatively slowly (compared to Figure 1) to the output of the high-fidelity model, after about $0.5\,\mathrm{s}$. The partition of the state space into regions with constant (K_i, k_i) is shown in Figure 4, with the phase plane trajectory of the reduced state $\hat{x}$ for the simulation in Figure 3 indicated by the dotted line.

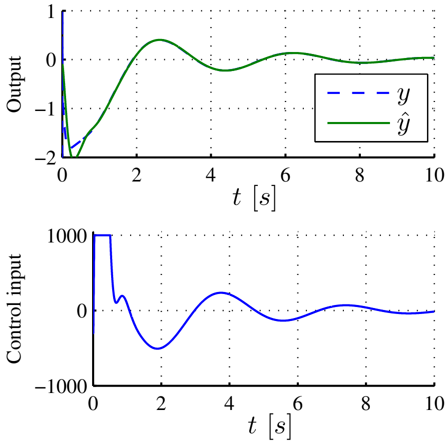

Fig. 3. Closed-loop performance with $r = 2$ and an optimized basis. Top: High-fidelity y and estimated $\hat{y}$. Bottom: eMPC control input, constrained such that $|u| < 1000$. The input constraint is seen to be active during the first half second.

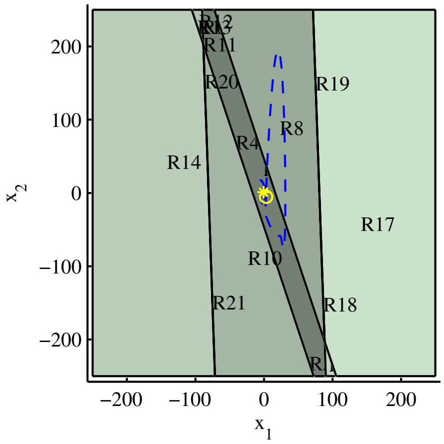

Fig. 4. Example of state-space partition for a reduced-order model with $r = 2$, with state trajectory starting in o and ending in $*$. The different color shades indicate the 21 regions $\mathbf{R}_i$ in the state space. The controller feedback matrices (K_i, k_i) are constant within each region.

Based on simulations, the reduced-order models generated with the optimized basis perform better in closed loop than the POD models. For this benchmark, they are able to handle higher controller gains, the output is regulated faster to the origin and the control action is smoother. This is illustrated by Figure 5. The difference in performance may be attributed to the way in which the goal-oriented models are targeted to give an accurate approximation of the output. For $r = 5$ it is also observed that the output from the reduced-order models converge to the ´true’ output an order of magnitude faster than for $r = 2$, resulting in a better closed-loop response. This is what one would expect; adding more states to the reduced-order model leads to better approximations.

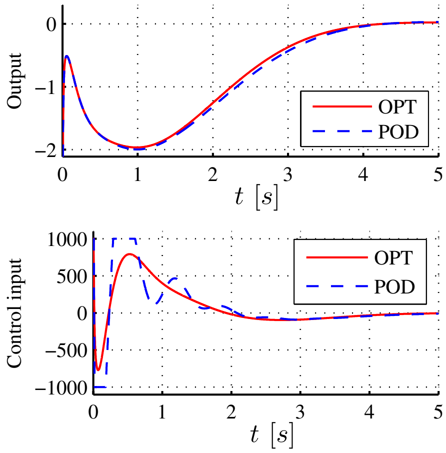

Fig. 5. Performance comparison for $r = 5$ with eMPC horizon $N = 10$. Top: Output of the full model using reduced-order control based on optimized- and POD basis. Bottom: Control input for the two different cases.

VI. Conclusions

The case study presented in this paper demonstrates the importance of reduced-order models that are appropriate for use in a controller design context. For the example presented here, reduced models derived using the goal-oriented, model-based approach led to substantially better closed-loop performance than those derived using the POD, which does not account for system outputs. The proposed methodology is also applicable to more complicated control tasks, such as nonlinear MPC and reference tracking, for which the explicit solution of the MPC problem can still be found (approximately, in some cases).

VII. Acknowledgements

The authors thank Petter Tøndel for providing mpQP software and Tan Bui-Thanh for helpful discussions. The first and third author acknowledge the financial support from The Research Council of Norway through the strategic university program Computational Methods in Nonlinear Motion Control. The second author acknowledges the support of AFOSR grant number FA9550-06-0271, technical monitor Dr. Fahroo.

References

[1] V. Adamjan, D. Arov, and M. Krein. Analytic properties of schmidt pairs for a Hankel operator and the generalized schur-takagi problem. *Math. USSR Sbornik*, 15:31–73, 1971.

[2] K. Afanasiev and M. Hinze. Adaptive control of a wake flow using proper orthogonal decomposition. In M. Dekker, editor, *Lecture Notes in Pure and Applied Mathematics*, Shape Optimization and Optimal Design, 2001.

[3] A. C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. Advances in Design and Control, SIAM, Philadelphia, 2005.

[4] E. Arian, M. Fahl, and E. Sachs. Trust-region proper orthogonal decomposition for flow control. Technical report, Universitt Trier, 2000.

[5] P. Astrid, L. Huisman, S. Weiland, and A. C. P. M. Backx. Reduction and predictive control design for a computational fluid dynamics model. In *Proc. 41st IEEE Conf. on Decision and Control*, volume 3, pages 3378–3383, Las Vegas, NV, 2002.

[6] J. A. Atwell, J. T. Borggaard, and B. B. King. Reduced order controllers for burgers’ equation with a nonlinear observer. *Applied Mathematics and Computational Science*, 11(6):1311–1330, 2001.

[7] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[8] P. Benner, E. S. Quintana-Ort, and G. Quintana-Ort. Balanced truncation model reduction of large-scale dense systems on parallel computers. *Mathematical and Computer Modelling of Dynamical Systems*, 6(4):383–405, 2000.

[9] M. Bettayeb, L. Silverman, and M. Safonov. Optimal approximation of continuous-time systems. In *Proc. of the 19th IEEE Conf. on Decision and Control*, volume 1, 1980.

[10] Y. Chahlaoui and P. van Dooren. A collection of benchmarks examples for model reduction of linear time invariant dynamical systems. SLICOT Working Note, 2002.

[11] T. Coleman and Y. Li. An interior, trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6:418–445, 1996.

[12] S. Gugercin and A. Antoulas. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77:748–766, 2004.

[13] Ø. Hegrenæs, J. Gravdahl, and P. Tøndel. Spacecraft attitude control using explicit model predictive control. *Automatica*, 41(12):2107–2114, 2005.

[14] L. Huisman and S. Weiland. Identification and model predictive control of an industrial glass feeder. In *Preprints of the 13th IFAC Symposium on System Identification*, pages 1685–1689, Rotterdam, The Netherlands, 2003.

[15] S.-Y. Kung and D. Lin. Optimal Hankel-norm model reductions: Multivariable systems. *IEEE Trans. on Automatic Control*, AC-26(1):832–852, 1971.

[16] K. Kunisch and S. Volkwein. Control of the Burgers Equation by a Reduced-Order Approach Using Proper Orthogonal Decomposition. *Journal of Optimization Theory and Applications*, 102(2):345–371, August 1999.

[17] J. Li and J. White. Low rank solution of Lyapunov equations. *Journal on Matrix Analysis and Applications*, 24(1):260–280, 2002.

[18] J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education, 2001.

[19] B. C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans. on Automatic Control*, 26(1):17–32, 1981.

[20] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Verlag, 1999.

[21] S. S. Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 34:425–448, 2000.

[22] J. B. Rawlings. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, 20(3):38–52, 2000.

[23] L. Sirovich. Turbulence and the dynamics of coherent structures. part 1: Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.

[24] D. Sorensen and A. Antoulas. The sylvester equation and approximate balanced reduction. *Linear Algebra and Its Applications, Fourth Special Issue on Linear Systems and Control, edited by Blondel et al.*, pages 260–445, 2002.

[25] P. Tøndel. *Constrained Optimal Control via Multiparametric Quadratic Programming*. PhD thesis, Department of Engineering Cybernetics, NTNU, 2003.

[26] P. Tøndel, T. A. Johansen, and A. Bemporad. An algorithm for multi-parametric quadratic programming and explicit mpc solutions. *Automatica*, 39(3):489–497, 2003.

[27] P. Tøndel, T. A. Johansen, and A. Bemporad. Computation of piecewise affine control via binary search tree. *Automatica*, 39(5):945–950, 2003.

[28] K. E. Willcox, O. Ghattas, B. van Bloemen Waanders, and B. Bader. An Optimization Framework for Goal-Oriented, Model-Based Reduction of Large-Scale Systems. In *Proc. 44th IEEE Conf. on Decision and Control*, Seville, 2005.