Explicit Model Predictive Control for Large-Scale Systems via Model Reduction

Svein Hovland* and Jan Tommy Gravdahl†

Norwegian University of Science and Technology, N-7491 Trondheim, Norway

and

Karen E. Willcox‡

Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

In this paper we present a framework for achieving constrained optimal real-time control for large-scale systems with fast dynamics. The methodology uses the explicit solution of the model predictive control (MPC) problem combined with model reduction, in an output-feedback implementation. 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-constrained optimization formulation that yields efficient models tailored to the control application at hand. The approach is illustrated on a challenging large-scale flow problem that aims to control the shock position in a supersonic diffuser.

I. Introduction

With the increasing interest in fluid flow control over the last decade, there arises a need for control methodology that can achieve constrained optimal real-time control of distributed systems with fast dynamics, such as in mechatronics, MEMS, rotating machinery and acoustics. Computational fluid dynamic (CFD) models of such systems typically have order

*PhD Student, Department of Engineering Cybernetics; svein.hovland@itk.ntnu.no.

†Professor, Department of Engineering Cybernetics; tommy.gravdahl@itk.ntnu.no.

‡Associate Professor, Department of Aeronautics and Astronautics; kwillcox@mit.edu. Associate Fellow AIAA.

exceeding 10^4, which is prohibitive for model-based controller design. In order to achieve real-time control, the control structure must be capable of computing the control input faster than the sampling rate of the system. Therefore, we need approximate simulation models that are of sufficiently low order for control design, and a framework for coupling the controller with the plant based on the approximate models, while accounting for the error inherent in the approximate model.

While considerable progress has been made in the development of reduced-order models for flow control applications, their application in the constrained optimal control setting has remained out of reach for complex flow applications, due to a need for models of very low order that target the control problem at hand. Here, we present a new framework for achieving real-time constrained optimal control for large-scale systems with fast dynamics that exploits recent advances in a goal-oriented model reduction methodology and explicit model predictive control (eMPC). Demonstrating the feasibility of such control problems is essential if reduced-order modeling methods are to be adopted onboard actual aerospace systems.

Model predictive control (MPC) is a control strategy that has been widely adopted in the industrial process control community and implemented successfully in many 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, since an optimization problem (often a constrained quadratic program) is solved at each sampling time. This has limited the use of these controllers to processes with relatively slow dynamics.

It has recently been shown that a great deal of the computational effort in traditional MPC can be done offline. Algorithms have been presented for solving multiparametric quadratic programs (mpQPs) that are used to obtain explicit solutions to the MPC problem.[1,2] Thus, the explicit model predictive controller 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, which makes eMPC attractive for the fast systems discussed above, 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. However, the use of eMPC is critically dependent on having a system model of low order, typically with a maximum of ten states. For CFD applications, this motivates a need for model order reduction methodology that can provide reduced models of very low order, that at the same time are suitable for control.

Model reduction for control of large-scale systems has been considered in a number of settings.[3–8] MPC based on a linear reduced model derived from a CFD model using proper orthogonal decomposition (POD) has been demonstrated to perform well for the control of an industrial glass-feeder.[9] In that work, it is stated that in order to use the reduced-order simulation models for the glass-feeder control in practice, 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. One recently proposed approach to achieve this is the OS-POD method, which generates reduced models for control by iteratively computing a POD basis that targets the closed-loop optimality system.[10] Another recently proposed approach is to determine the reduced model by solving a goal-oriented optimization problem.[11]

The contribution of this paper is twofold: 1) We propose an approach for achieving constrained optimal control in applications that are described by models of high order, while being characterized by fast sampling rates, by combining a goal-oriented model reduction method with the explicit solution to the MPC problem. We attach the control structure to the plant with a Kalman filter that accounts for the error introduced in the model approximation process. 2) We demonstrate the performance of reduced models obtained by goal-oriented optimization in control system design.

The paper is organized as follows. Section II describes MPC and the explicit solution via multiparametric quadratic programming. In Section III we describe the goal-oriented reduction method, and discuss reduced-order output-feedback control, closed-loop issues and state estimation. The proposed methodology is then demonstrated for a realistic example in Section IV. Throughout the paper, positive (semi) definiteness of matrices is indicated by $\succ 0 (\succeq 0)$.

II. MPC and the Explicit Solution

A brief outline of the standard MPC formulation will be given before we address the explicit solution. For further reading on MPC, there exists a number of books [12, page 36-246], [13, page 3-219] and tutorials.[14]

A. A Standard MPC Formulation

Model predictive control is formulated for a discrete-time state-space model

$$x_{k+1} = A_d x_k + B_d u_k, \tag{1a}$$

$$y_k = C_d x_k, \tag{1b}$$

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 constant matrices A_d, B_d and C_d are of appropriate dimensions, and (A_d, B_d) is a controllable pair. For the regulator problem, the model predictive controller solves at time step k the optimization problem

$$\min_{U_k} \Big\{ x_{k+\mathcal{N}|k}^T P x_{k+\mathcal{N}|k} \tag{2a}$$

$$+ \sum_{i=0}^{\mathcal{N}-1} \left(x_{k+i|k}^T Q x_{k+i|k} + u_{k+i}^T R u_{k+i} \right) \Big\}$$

subject to:

$$u_{\min} \leq u_{k+i} \leq u_{\max}, \; i = 0, \ldots, \mathcal{M}-1 \tag{2b}$$

$$y_{\min} \leq y_{k+i} \leq y_{\max}, \; i = 1, \ldots, \mathcal{N} \tag{2c}$$

$$u_{k+1} = K x_{k+i|k} \; , \; \mathcal{M} \leq i \leq \mathcal{N}-1 \tag{2d}$$

$$x_{k|k} = x_k \tag{2e}$$

$$x_{k+i+1|k} = A_d x_{k+i|k} + B_d u_{k+i}, \; i \geq 0 \tag{2f}$$

$$y_{k+i|k} = C_d x_{k+i|k}, \; k \geq 0, \tag{2g}$$

where P and Q are design weighting matrices of appropriate dimensions that penalize deviation from zero of the states x_{k+i} at the end of the prediction horizon $\mathcal{N}$ and over the entire horizon, respectively. In this work, the final cost matrix P and gain K are calculated from the algebraic Riccati equation, under the assumption that the constraints are not active for $k \geq \mathcal{N}$. The weight R penalizes use of control action u. The notation $(\cdot)_{k+i|k}$ is used to emphasize that the predictions $(\cdot)_{k+i}$ are made based on the value at step k. $\mathcal{M}$ defines the control horizon, which is the number of future control moves to be optimized. In this work, we set $\mathcal{M} = \mathcal{N}$, for convenience. The sequence $U_k = \begin{bmatrix} u_k^T & u_{k+1}^T & \ldots & u_{k+\mathcal{M}-1}^T \end{bmatrix}^T$ contains the future control inputs that yield the best predicted output with respect to the performance criterion on the prediction horizon. Once this set has been found, the first control input u_k is applied to the process, before the whole optimization problem is re-solved 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.

B. Explicit MPC via Multiparametric Quadratic Programming

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 [15, page 1-2].

It is well established that implementing a linear model predictive controller requires solving a quadratic program (QP) in U_k at each time step.[12] With some manipulations, the problem in (2) can be written

$$\min_{U_k} \left\{ \frac{1}{2} U_k^T H U_k + x_k^T F U_k \right\} \tag{3a}$$

$$\text{subject to: } G U_k \le W + E x_k, \tag{3b}$$

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 the weighting matrices in (2a) satisfy $P \succeq 0$, $R \succ 0$ and $Q \succeq 0$, then $H \succ 0$ and the problem is strictly convex. The Karush-Kuhn-Tucker conditions (KKT) are then sufficient conditions for optimality [16, page 333], and the solution U_k can be shown to be unique.[1] The assumptions on Q and R are usually met by choosing Q and R to be diagonal matrices that appropriately penalize the relative importance of state or input values. The problem (3) can be viewed as an mpQP in U_k, where x_k is a vector of parameters.

By defining

$$z \triangleq U_k + H^{-1} F^T x_k, \tag{4}$$

the problem in (3) can be transformed into

$$\min_{z} \left\{ \frac{1}{2} z^T H z \right\} \tag{5a}$$

$$\text{subject to: } G z \le W + S x_k, \tag{5b}$$

which is an mpQP in z, parameterized by x_k. The matrix S is found as $S = E + G H^{-1} F^T$. By considering the KKT conditions of this 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).[1] 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,

$$U_k = K_i x_k + k_i, \ i = 1, \ldots, \mathcal{N}_p \tag{6}$$

where $\mathcal{N}_p$ is the number of polyhedral regions and the subscript i denotes the ith affine function. K_i and k_i 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 to evaluating a piecewise affine function of the current state, by determining the region i in which the current state x_k resides. By implementing the piecewise affine function as a binary search tree, the online computational time is logarithmic in the number of polyhedra in the state space partition.[17]

III. 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 therefore use reduced-order control, where reduced-order models are used to design output-feedback explicit model predictive controllers for the high-fidelity model.

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 and passivity. In the control community, algorithms such as optimal Hankel model reduction[18–20] and balanced truncation[21] are known to have strong guarantees on the quality of the reduced model by providing upper bounds for the approximation error. Although recent and ongoing research address the extension of these algorithms to large-scale settings,[22–25] model reduction of very large-scale models with rigorous guarantees on quality remains a challenge. In addition, balanced truncation is limited to linear systems.

Model reduction for *control* is somewhat different from model reduction for simulation purposes. A reduced 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,[6,10,26,27] 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.

POD has been used with success in control of large-scale models, including some nonlinear applications. However, there are several limitations associated with using the 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 LTI systems, reduction via POD can lead to undesirable and unpredictable results, such as unstable reduced models. A recently proposed goal-oriented model-constrained reduction algorithm[11] is targeted at large-scale applications in optimal control and optimal design. This approach retains applicability to nonlinear systems, but addresses some of the limitations of the POD 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.

Formulation of the problem of determining the basis as an optimal control problem has also been considered for distributed parameter systems.[28] In this section, we briefly present the model reduction methodology and then describe the reduced-order control framework that uses eMPC.

A. Goal-Oriented Model-Constrained Reduction

Order reduction of the discretized LTI system (1) can be achieved using a projection framework, which assumes that the state x_k is approximated by a linear combination of r basis vectors

$$x_k \approx \Phi x_{r_k}, \tag{7}$$

where $x_{r_k} \in \mathbb{R}^r$ is the reduced state at step k 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 (7) into (1), and requiring the resulting residual to be orthogonal to the space spanned by Φ gives the reduced model

$$x_{r_{k+1}} = A_r x_{r_k} + B_r u \tag{8a}$$

$$y_{r_k} = C_r x_{r_k}, \tag{8b}$$

where $A_r = \Phi^T A_d \Phi \in \mathbb{R}^{r\times r}$, $B_r = \Phi^T B_d \in \mathbb{R}^{r\times m}$, $C_r = C_d \Phi \in \mathbb{R}^{p\times r}$ and y_{r_k} is the output of the reduced model.

In the goal-oriented model-constrained algorithm, an optimization problem is solved to determine the basis. The optimization problem seeks to find the rth-order basis $\Phi \in \mathbb{R}^{n\times r}$ and the corresponding reduced-order state solution $x_{r_k} \in \mathbb{R}^r$, $k = 1, 2, \ldots, \mathcal{T}$ so that the square of the $\mathcal{L}_2$-norm of the error between the full-order and reduced-order output is minimized for a selected set of inputs, over some time horizon $\mathcal{T}$. This can be formulated as

$$\min_{\Phi, x_r} \frac{1}{2} \sum_{\ell=1}^{\mathcal{S}} \sum_{k=1}^{\mathcal{T}} \left(y_k^\ell - y_{r_k}^\ell\right)^T \left(y_k^\ell - y_{r_k}^\ell\right) + \frac{\beta}{2} \left[\sum_{j=1}^{r} \left(1 - \phi_j^T \phi_j\right)^2 + \sum_{i,j=1, i\neq j}^{r} \left(\phi_i^T \phi_j\right)^2 \right] \tag{9a}$$

subject to:

$$\Phi^T \Phi x_{r_{k+1}}^\ell = \Phi^T A_d^\ell \Phi x_{r_k}^\ell + \Phi^T B_d^\ell u_k^\ell, \; \ell = 1, \ldots, \mathcal{S}, \; k = 1, \ldots, \mathcal{T}, \tag{9b}$$

$$\Phi x_{r_1}^\ell = x_0^\ell, \; \ell = 1, \ldots, \mathcal{S}, \tag{9c}$$

$$y_{r_k}^\ell = C_d^\ell \Phi x_{r_k}^\ell, \; \ell = 1, \ldots, \mathcal{S}, \; k = 1, \ldots, \mathcal{T}. \tag{9d}$$

The summation over ℓ in the objective function permits one to consider a finite set of $\mathcal{S}$ instantiations of the governing equations (1) that could arise from variations in the coefficient matrices A_d, B_d and C_d, the input u, or the initial state x_0. The superscript ℓ thus denotes the ℓth instance of the system, which has corresponding state x^ℓ, input u^ℓ, and output y^ℓ. For example, where (1) represents a (spatially and temporally) discretized partial differential equation (PDE), these variations could stem from changes in the domain shape, boundary conditions, coefficients, initial conditions or sources of the underlying PDEs.

The key differences between the formulation (9) and the POD are that (1) the model-constrained optimization approach enforces the reduced-order governing equations as constraints, and (2) the cost is targeted to minimize the *output* error, while the POD minimizes the error of state prediction over the entire domain. Thus, using the model-constrained optimization approach, reduced models are obtained that are more suitable for use in an output-feedback implementation. This is achieved by overcoming reduced model stability problems such as those observed for the POD,[11] and by yielding accurate models of very low order through targeted prediction of system outputs.

The second term in the cost function (9a) is a regularization term to yield orthonormal basis vectors, with β as a regularization parameter. This regularization acts only in the null space of the projected Hessian matrix of the first term of (9a). Therefore, the reduced output approximation, y_r, 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 (9) should therefore be tolerant of singular projected Hessian matrices.

The formulation defined by equations (9) provides a mathematical definition of the desired optimal basis; however, in practice this optimization problem may not be tractable for large-scale problems. In a computationally efficient implementation of the method,[11] the basis functions are assumed to be a linear combination of the collection of full-state snapshots. Then, the number of optimization variables is reduced to Mr, where M is the number of snapshots and r is the dimension of the reduced state. As a consequence, neither the gradient computation nor the optimization step computation (which dominate the cost of an optimization iteration) scale with the full system size n.

Determining the basis via the optimization procedure will in general be more computationally demanding than using POD. However, this additional offline cost is a tradeoff that we make in order to achieve sufficiently low order in the reduced models so that eMPC becomes tractable for challenging flow control problems.

B. eMPC Control Setup

The control setup of the eMPC framework that uses the reduced-order model is illustrated in Figure 1.

The complexity of the proposed control scheme is given by the offline model reduction

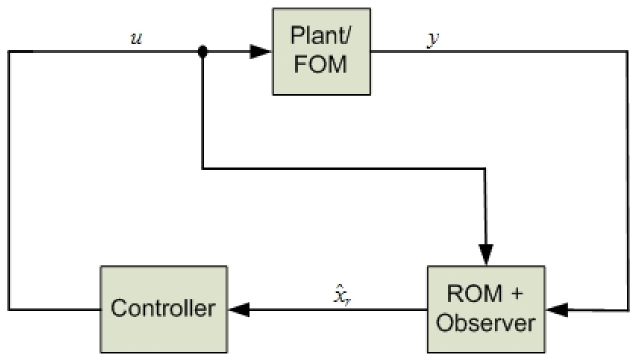

Figure 1. Block diagram of the output-feedback setup. In this case $\hat{x}_r$ is an estimate of the reduced state x_r based on an observer using the reduced model (ROM) and measurements from the CFD model (FOM).

cost plus the cost of solving the eMPC problem offline for the reduced model. The former is determined by the number of optimization variables in the optimization problem (9), which is Mr, as well as the cost of solving the high-fidelity model (to generate the snapshots and to compute the gradient information required by the optimizer). The cost of solving the eMPC problem is problem dependent, but increases rapidly with the number of parameters, the length of the control horizon $\mathcal{M}$, and the number of constraints in the mpQP. For problems whose solutions consist of a large number of regions, one can easily run into numerical problems. Also, the memory required to store the eMPC solution online increases rapidly as the size of the solution grows. A large number of polyhedra in the online solution requires a large search tree with many nodes, which entails a longer searching process which might compromise real-time requirements. The scheme is therefore limited to cases where the reduced models can be made reasonably small, typically with around ten states.

Further complexity reduction techniques, such as input blocking, can be used to make the eMPC procedure more tractable in cases where the problem is large.

C. Reduced State Estimation

The eMPC control input is computed based on the reduced state vector at every iteration, and x_r must therefore be estimated by an observer, based on the output of the CFD model. In systems with output constraints, it is particularly important that the output of the reduced model is a good estimate of the plant output. The observer should therefore account for the approximation error in the reduced model.

A basic linear observer, such as the Luenberger observer, does not account explicitly for uncertainties that are amplified by the observer gain matrices. Consequently, the state estimate may not be accurate enough in the presence of model perturbation. We therefore follow common practice in the literature,[29,30] and use a Kalman filter, which is known to have desirable properties for systems with noise in outputs and state equations. The Kalman filter can be defined in terms of the discretized reduced model

$$x_{r_{k+1}} = A_r x_{r_k} + B_r u_k + \Gamma w_k \tag{10a}$$

$$y_{r_k} = C_r x_{r_k} + v_k, \tag{10b}$$

where v_k and w_k are assumed to be zero mean white noise processes with covariance matrices $R_k = R_k^T \succ 0$ and $Q_k = Q_k^T \succ 0$, respectively, and where Γ defines the mapping between w_k and the states. In this setup, the noise processes account for uncertainty in the state equations through Γw_k, and the uncertainty in the output through v_k.

D. Stability, Feasibility and Constraint Fulfillment

A number of questions regarding robust stability, feasibility and robust constraint fulfillment arises when the reduced model is used to control the high-order model. Such robustness analysis is an active research subject *per se*, and it is outside the scope of the current paper. We therefore use the nominal model (the reduced model) for controller design, and address certain robustness issues during the design stage in Section IV. In this section we will point to other possible solutions to the robustness problems.

Since the explicit MPC solution is equivalent to the standard MPC solution obtained by solving (2), many methods for robust stability analysis techniques developed for standard MPC[31] can be used to conclude stability for the reduced-order eMPC in the presence of the uncertainty introduced through the model reduction process.

There are in general two approaches for addressing robustness in MPC. In the first approach, the plant uncertainty is expressed by allowing the state-space matrices to be arbitrarily time-varying and belonging to a polytope.[32] Recent contributions include triple mode MPC algorithms that allow large feasibility regions.[33] In the second approach, the plant is assumed to be known, and a bounded unmeasured disturbance is introduced in the state equations (1). MPC stability in the presence of model uncertainty has been addressed in Refs. 34–36, and tests for robust MPC stability of input-constrained systems with unstructured uncertainty have recently been established.[37] The applicability of these methods to establish robustness in the context of MPC with reduced-order models remains a challenging open research question.

Given the uncertainty introduced through the model reduction process, one cannot guarantee that feasibility of the underlying optimization problem is maintained and that the constraints on the states/outputs are fulfilled. This problem is handled through the use of soft constraints in Section IV-C. Relaxing the state constraints in effect removes the feasibility problem, at least for stable systems.[31]

IV. Numerical Results

The following example is a challenging model reduction problem with $n = 11,730$ states, where the objective is to control the position of a shock in a supersonic inlet. The problem is based on an unsteady CFD formulation to simulate subsonic and supersonic flows through a jet engine inlet that is designed to provide a compressor with air at the required conditions.[38]

A. Control Problem Setup

The control setup is shown in Figure 2. In nominal flow conditions, a strong shock sits downstream of the inlet throat. In order to stabilize the shock position in the presence of incoming flow disturbances, and thus prevent engine unstart, active flow control is effected through flow bleeding upstream of the throat. The case considered has a steady-state Mach

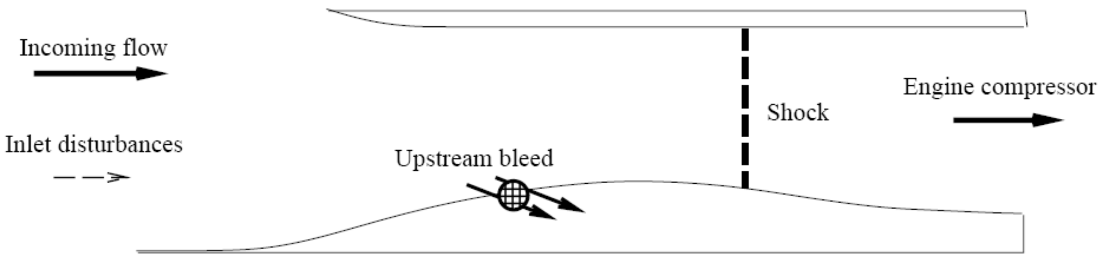

Figure 2. Active flow control setup for the supersonic inlet.[38]

number of 2.2. The flow is assumed inviscid and is modeled by the Euler equations. The underlying CFD code is nonlinear, and the model is linearized about a steady-state solution, giving a stable continuous-time model of the form[a]

$$E\dot{x} = Ax + Bu, \tag{11a}$$

$$y = Cx, \tag{11b}$$

where the continuous-time state $x(t)$ contains the n unknown perturbation flow quantities at each point in the computational grid, and the matrices A, B, C and E result from the CFD spatial discretization of the Euler equations. The vector $u \in \mathbb{R}^2$ is the input to the system and $y \in \mathbb{R}$ contains the system output. In this case, the flow state quantities are density, flow velocity components and enthalpy, and the output y is the average Mach number at the throat. There are $3,078$ grid points in the computational domain, giving a total of $n = 11,730$ unknowns. The descriptor matrix E is sparse, and some rows contain only zeros; consequently, E is singular and the inlet model represents a general differential algebraic

[a]The system matrices are available in the Oberwolfach Model Reduction Benchmark Collection.

equation system. The input u contains bleed actuation b (manipulated variable) and an incoming density disturbance d, i.e.

$$u \triangleq \begin{bmatrix} b \\ d \end{bmatrix}. \tag{12}$$

A discrete-time system is obtaining by applying a backward Euler time integration method to (11). The general projection framework for model reduction described in Section III-A can then be applied.

The high order of the model (11) is prohibitive for optimal and model-based control, which motivates the use of model reduction. It should be noted that this benchmark is relatively difficult to approximate. Various model reduction methods have been applied to this problem with varying degrees of success. As shown in Ref. 39, POD and Krylov-based methods yield reduced models that are unstable, unless the sampled frequencies are chosen very carefully. One reason for this may be that there are inverse responses from the inputs to the output, suggesting non-minimum phase. Non-minimum phase systems are harder to approximate than minimum phase systems.[40] Balanced truncation is guaranteed to produce stable models, but is difficult to apply in this case due to the singular descriptor matrix E. Good results were shown using the Fourier model reduction approach;[39] however, that method is applicable only to linear systems.

The optimization methodology described in Section III-A is extendable to nonlinear systems, and also is likely to yield stable reduced models, since the objective function includes the actual error between full and reduced models. Therefore, the optimization approach is used here to approximate the input/output relationship between the two inputs u and the output y for the inlet example. The model reduction procedure handles reduction of MIMO system models in a straightforward manner through the framework for parametric variations described in Section III-A.

The eMPC framework can be extended naturally to handle disturbances such as the density disturbance. In the controller, we obtain a reduced-order prediction model of the form

$$\hat{x}_{r_{k+i+1}} = A_r \hat{x}_{r_{k+i}} + B_r^b b_{k+i} + B_r^d d_{k+i|k} \tag{13a}$$

$$\hat{y}_{r_{k+i}} = C_r \hat{x}_{r_{k+i}}; \; i \geq 0, \tag{13b}$$

where B_r^b and B_r^d are the columns of B_r corresponding to the inputs b and d, respectively, and $i = 1, \ldots, \mathcal{N}$ is the ith step on the prediction horizon. We assume that the disturbance d_k is measured, and we use the notation $d_{k+i|k}$ to emphasize that the disturbance d_{k+i}, given the measured value at time step k, is predicted based on an assumption on the future behavior of the disturbance. If we assume that the disturbance is constant over the prediction horizon, one straightforward way to implement the prediction model (13) is to augment the state vector and the system matrices as follows:

$$\hat{x}_{r_k} \leftarrow \begin{bmatrix} \hat{x}_{r_k} \\ d_k \end{bmatrix}, \tag{14}$$

$$A_r \leftarrow \begin{bmatrix} A_r & B_r^d \\ 0 & 1 \end{bmatrix}, \tag{15}$$

and

$$C_r \leftarrow \begin{bmatrix} C_r & 0 \end{bmatrix}. \tag{16}$$

To avoid numerical difficulties (the augmented system is marginally stable if we set $d_{k+1} = d_k$), we replace the 1 in equation (15) with a scalar δ, and typically choose $\delta = 0.99$.

Now, the control structure can be summarized as follows:

- The Mach number is measured using the output equation

$$y_k = Cx_k. \tag{17}$$

- The reduced state is estimated using a Kalman filter based on the reduced-order model and the output of the CFD model.
- The reduced state estimate is fed to the explicit model predictive controller along with the measured disturbance, where the bleed input b_k is found as an explicit function of the augmented state (14).
- Control is effected through upstream bleed.

For all results presented in the following, the inlet model is discretized with a time step of $\Delta_t = 0.025\,\mathrm{s}$. The controllers are verified to be sufficiently fast for this example.

B. Model Reduction Results

In order to test the model reduction algorithm, we compare time-domain and frequency-domain responses for the CFD model and models of reduced order. We consider a reduced model with 10 states, which was the lowest order that gave sufficient approximation quality. The optimized basis is found by minimizing the output error for 200 samples in the interval $t \in (0, 2)\,\mathrm{s}$ in response to a step in each of the two inputs. That is, first we set $b \equiv 1$ and $d \equiv 0$ and collect 200 samples in the time interval, and then we re-initialize the model, set $b \equiv 0$ and $d \equiv 1$ and collect another 200 samples in the same time interval. We use the

POD basis vectors generated from the snapshot data as an initial guess for the optimization algorithm. Comparisons with POD reduced models themselves are not given, since for this example instability of POD reduced models is observed over a wide range of reduced-order state dimensions, prohibiting the use of POD for the reduced-order controller design.

The transfer functions from bleed b to output y are shown in Figure 3 for the CFD model and the reduced model obtained with an optimized basis. Figure 4 illustrates the same

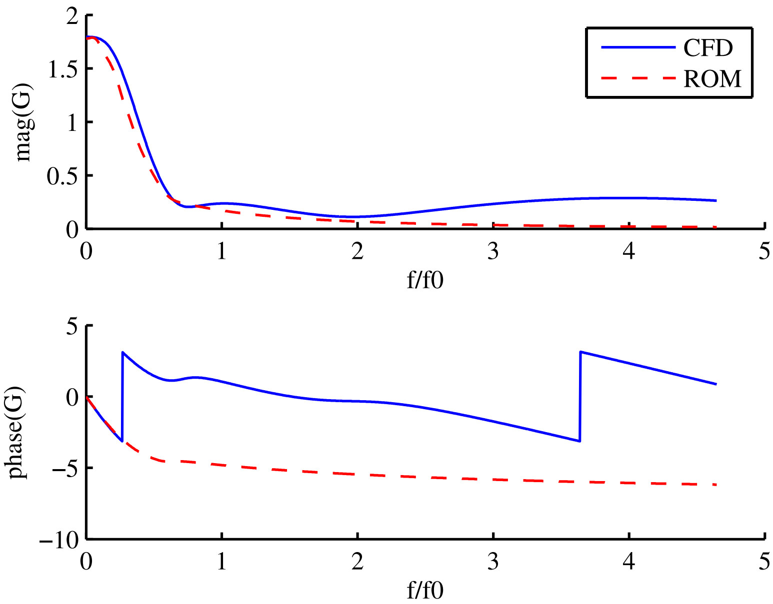

Figure 3. Comparison of transfer function from bleed b to Mach number y for the CFD model (11,730 states) and the reduced model of order $r = 10$.

comparison for the transfer function from disturbance input d to output y. The transfer function from the disturbance to the output shows that the dynamics contain a delay, and are consequently more difficult for the reduced-order model to approximate. The reduced-order model is accurate for lower frequencies, but does not capture the disturbance response at higher frequencies. However, these higher frequencies are unlikely to occur in typical atmospheric disturbances;[39] thus, the reduced model performance shown in Figures 3 and 4 is deemed acceptable for the purposes of controller design. Figure 5 shows the time-domain responses to a step in bleed actuation and a Gaussian density disturbance input. The frequency content of this disturbance input is representative of that expected in practical flight conditions. It can be seen that the reduced model accurately predicts the time-domain response, confirming its suitability for conditions of practical interest.

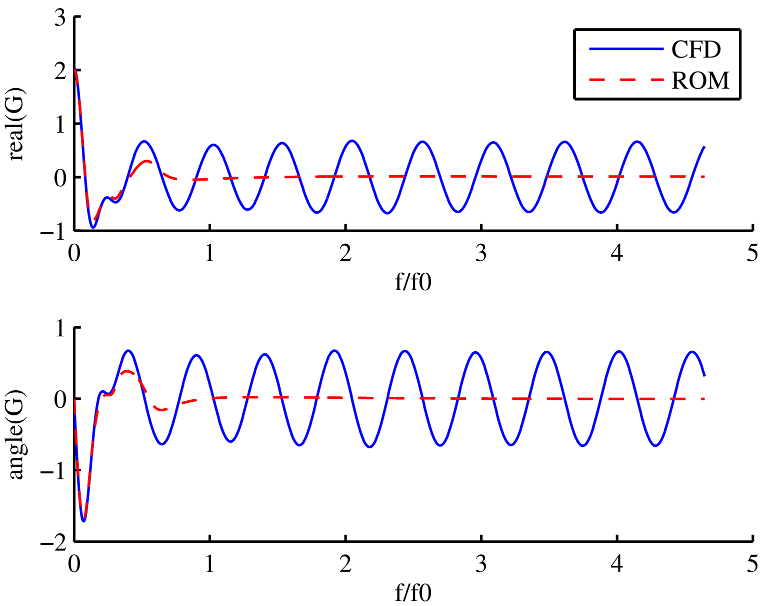

Figure 4. Comparison of transfer function from disturbance d to Mach number y for the CFD model (11,730 states) and the reduced model of order $r = 10$.

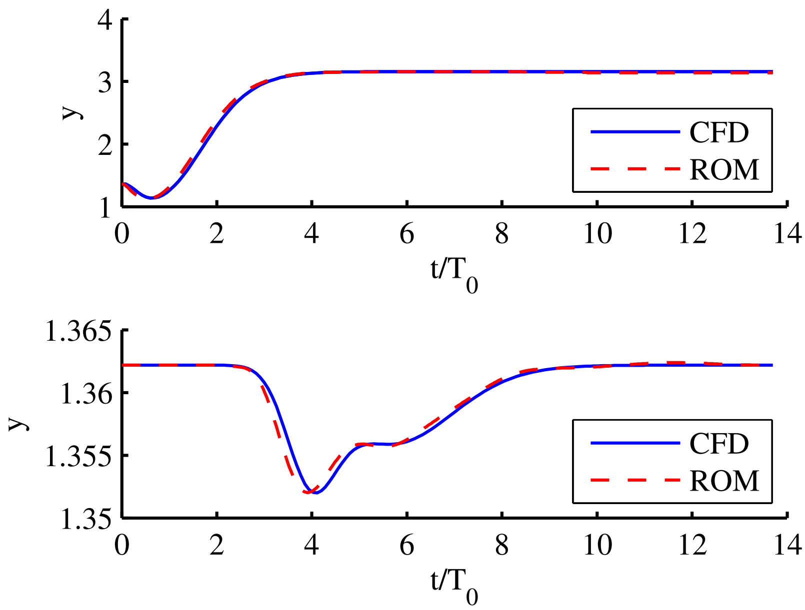

Figure 5. Top: Response in Mach number y to step in bleed input b for the CFD model (11,730 states) and the reduced model of order $r = 10$. Bottom: Response in Mach number y to Gaussian disturbance input d for the CFD model and a reduced model of order $r = 10$.

Case	Λ	α	t_p
1	0.01	$2f_0^2$	5
2	0.02	$2f_0^2$	5
3	0.04	$2f_0^2$	5

Table 1. Disturbance parameter values for different simulation cases.

C. Closed Loop Results

The disturbance input is set to be a Gaussian distribution, which is described by its amplitude Λ, rise time α and peak time t_p through the relation

$$d(t) = -\Lambda \rho_0 e^{-\alpha(t-t_p)^2}. \tag{18}$$

In the following, we address the controller robustness by tuning its performance for a set of disturbances for which the linear model is a good representation of the nonlinear CFD model. (Note that the linearized CFD model is only valid for small perturbations from steady-state conditions.) Subsequently, we add measurement noise to account for errors in the Mach number measurements. The parameter values for the disturbance inputs are shown in Table 1, and the different disturbance cases are shown in Figure 6.

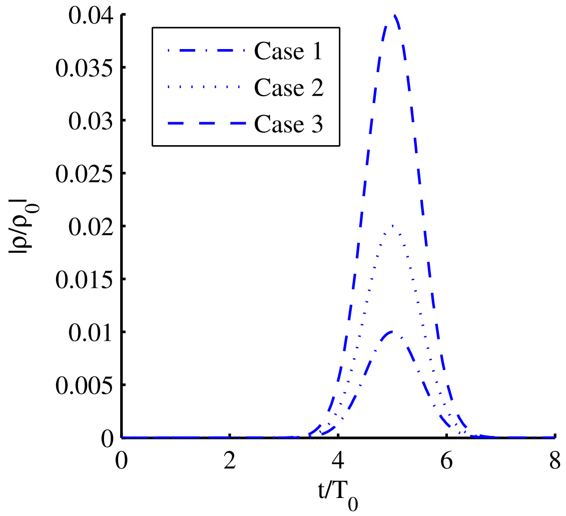

Figure 6. Magnitude of disturbance inputs used in Cases 1-3.

The computed control input b_k is in fact a perturbation about the nominal steady state bleed b^{ss} of 1% of the inlet mass flow,

$$b^{total} = b^{ss} + b_k. \tag{19}$$

We therefore require that the total bleed b^{total} is non-negative, i.e.

$$b_{k+i} \geq -0.01; \; i \geq 0. \tag{20}$$

We also put an upper bound on the control action,

$$b_{k+i} < b_{\max}; \; i \geq 0, \tag{21}$$

and we bound the Mach number at the throat

$$y_{\min} < y_{r_{k+i}} < y_{\max}; \; i \geq 0. \tag{22}$$

Since our objective is to prevent the shock from moving upstream causing engine unstart, we will set $y_{\min} > 1$, e.g. $y_{\min} = 1.1$. The controller tuning parameters are the weighting matrices, the prediction horizon, and the control horizon in the MPC formulation. Good performance is obtained by setting $\mathcal{M} = \mathcal{N} = 10$, $Q = C_r^T C_r$, $R = 0.05$ and P to the solution to the algebraic Riccati equation. The resulting closed-loop performance is shown for the different disturbance cases in Figure 7. It is seen that the controller gives good performance

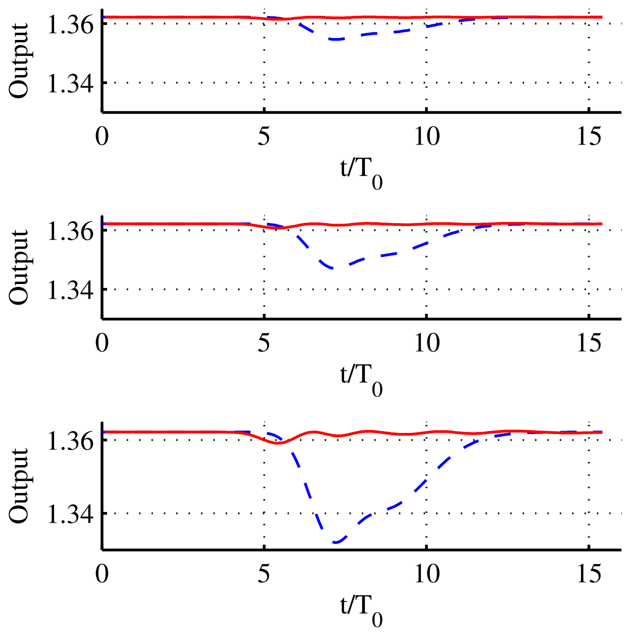

Figure 7. Uncontrolled (dashed) and controlled (solid) Mach number for Case 1 (top), Case 2 (middle) and Case 3 (bottom).

in all three cases. There are, however, some minor oscillations in the closed-loop response, which are attributed to full model/reduced model mismatch and inexact modeling of the disturbance in the prediction model. Recall that we assume that the disturbance is constant over the prediction horizon, while it in fact increases or decreases, corresponding to the shape of the Gaussian distribution. Also, the horizon $\mathcal{M} = \mathcal{N} = 10$ is somewhat short, especially since there is an inverse response from inputs to output.

Constraints on the states/outputs often represent operational desirables rather than fundamental operational constraints. In addition, from a practical point of view it does not make sense to use tight state constraints because of the presence of noise, disturbances and numerical errors. In order to guarantee feasibility of the MPC problem, we "soften" the constraints on the outputs. Soft constraints represent bounds that are allowed to be violated if necessary, with the violation being penalized in the cost function. The soft constraints are typically implemented by introducing so-called slack variables, s, in the constraint formulation. For instance, if we want a soft lower bound on y, we require

$$y_{\min} \leq y_{r_{k+i}} + s; \; i \geq 0 \tag{23}$$

$$s \geq 0, \tag{24}$$

and add a penalty term $f(s)$ to the cost function (2a). $f(s)$ is typically chosen as

$$f(s) = w \, \|s\|, \tag{25}$$

where w is a scalar weight and $\|s\|$ is some norm of s. Penalty functions that lead to constraint violation and use of slack only if the original problem is left infeasible are called *exact* penalty functions. Consequently, the constraints will not be violated unnecessarily if the penalty function is exact. In order to achieve an exact penalty function, the 1-norm or the ∞-norm must be used in (25), and the weight w must be sufficiently large.[41,42]

If we again consider disturbance Case 3, we see from Figure 7 that the controlled Mach number falls below 1.36. Now, we set $y_{\min} = 1.36$ as a soft constraint, and penalize constraint violation with an exact penalty function. The resulting Mach number is compared to the simulation from Figure 7 which has a hard constraint $y_{\min} = 1.1$ in Figure 8. The corresponding control inputs are shown in Figure 9.

To further address the question of robustness, we add noise to the measured Mach number y. For that purpose we add Gaussian white noise of different intensities to the output of the CFD model during the simulation, and study the effect in closed loop. Figure 10 shows a simulation run without noise, compared to three simulation runs with Gaussian white noise. It can be seen that in the presence of noise, particularly at the two lower levels, the controller performance remains good.

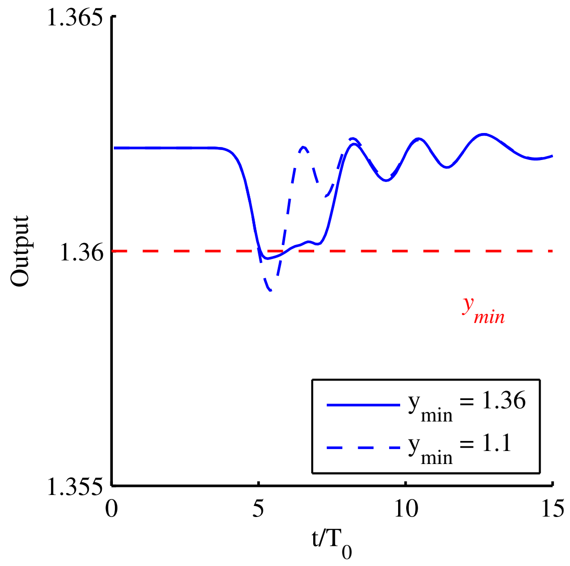

Figure 8. Mach number at inlet throat for two simulations with disturbance Case 3, with a soft constraint $y_{k+i} > 1.36$ and a hard constraint $y_{k+i} > 1.1$. The horizontal line indicates the soft lower bound for the soft-constrained case.

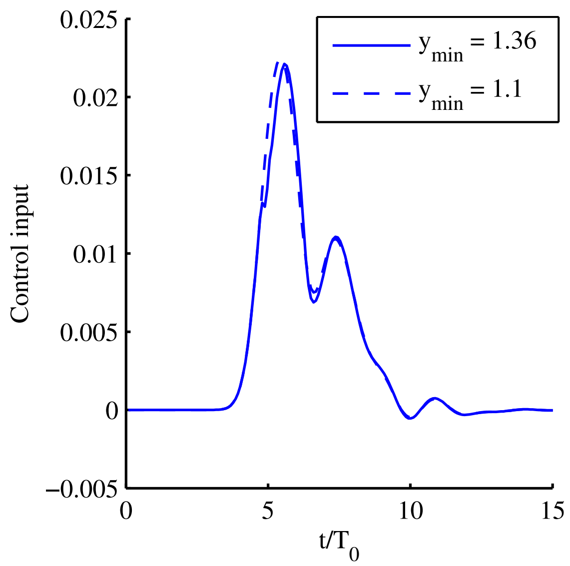

Figure 9. Control input for two simulations with disturbance Case 3, with a soft constraint $y_{\text{min}} = 1.36$ and a hard constraint $y_{\text{min}} = 1.1$.

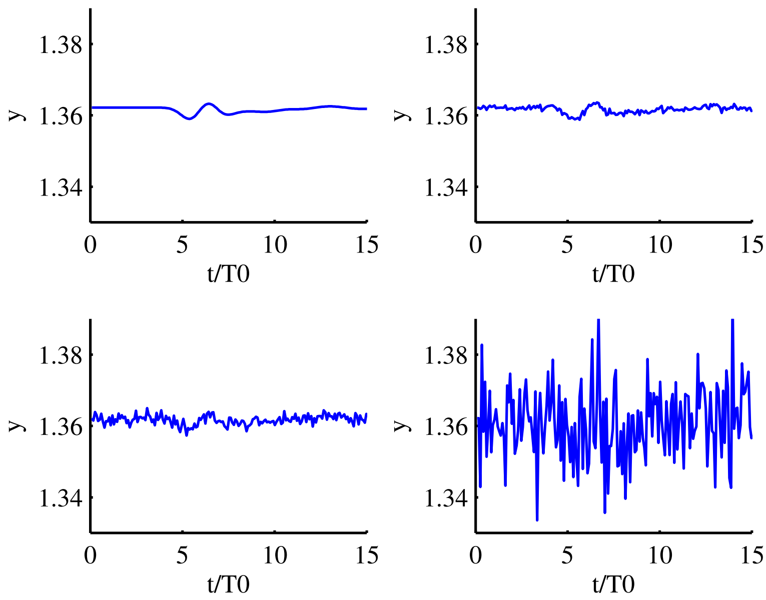

Figure 10. Controlled Mach number with measurement noise. Top left: No noise. Top right: Gaussian white noise of intensity 2.5×10^{-7}. Bottom left: Gaussian white noise of intensity 10^{-6}. Bottom right: Gaussian white noise of intensity 10^{-4}, corresponding to Mach number measurement accuracy within $\pm 0.01M$.

V. Conclusions

This paper presents a new framework for achieving real-time constrained optimal control for large-scale systems, such as those arising in aerospace flow control applications. The methodology, which combines eMPC with model reduction, is demonstrated for an example that considers control of a supersonic inlet. This example presents a significant challenge to model reduction methods. First, POD reduced models suffer from instability and thus cannot be used in a control setting. Further, obtaining models of very low dimensional is critical in order for the eMPC scheme to be viable for real-time control. Using a goal-oriented reduction methodology, we were able to derive a reduced model with ten states that yields acceptable approximation quality and is within the capacity of the eMPC scheme.

While we have not explicitly analyzed the robustness of the reduced model predictive controller, good performance is achieved by tuning based on exhaustive simulations for ranges of operating conditions. In many cases this approach leads to better performance than using robust MPC techniques. Choosing the right robust MPC technique is an art, and much experience is necessary to make it work.

The proposed methodology is also applicable for more complicated control tasks, such as nonlinear MPC and reference tracking, for which the explicit solution of the MPC problem can still be found (although approximately, in some cases).

Acknowledgments

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

References

[1]Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E. N., "The explicit linear quadratic regulator for constrained systems," *Automatica*, Vol. 38, No. 1, 2002, pp. 3–20.

[2]Tøndel, P., Johansen, T. A., and Bemporad, A., "An algorithm for multi-parametric quadratic programming and explicit MPC solutions," *Automatica*, Vol. 39, No. 3, 2003, pp. 489–497.

[3]Kunisch, K. and Volkwein, S., "Control of the Burgers Equation by a Reduced-Order Approach Using Proper Orthogonal Decomposition," *Journal of Optimization Theory and Applications*, Vol. 102, No. 2, August 1999, pp. 345–371.

[4]Ravindran, S. S., "A Reduced-Order Approach for Optimal Control of Fluids Using Proper Orthogonal Decomposition," *International Journal for Numerical Methods in Fluids*, Vol. 34, 2000, pp. 425–448.

[5]Atwell, J. A., Borggaard, J. T., and King, B. B., "Reduced Order Controllers for Burgers' Equation with a Nonlinear Observer," *Applied Mathematics and Computational Science*, Vol. 11, No. 6, 2001, pp. 1311–1330.

[6]Afanasiev, K. and Hinze, M., "Adaptive control of a wake flow using proper orthogonal decomposition," *Lecture Notes in Pure and Applied Mathematics*, edited by M. Dekker, Shape Optimization and Optimal Design, 2001, pp. 317–332.

[7]Ahuja, S., Rowley, C. W., Kevrekidis, I. G., Wei, M., Colonius, T., and Tadmor, G., "Low-dimensional models for control of leading-edge vortices: Equilibria and linearized models," *45th AIAA Aerospace Sciences Meeting and Exhibit*, January 2007, AIAA Paper 2007-709.

[8]Cohen, K., Siegel, S., Seidel, J., and McLaughlin, T., "Reduced Order Modeling for Closed-Loop Control of Three Dimensional Wakes," *3rd AIAA Flow Control Conference*, San Francisco, California, June 2006, AIAA Paper 2006-3356.

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

[10]Kunisch, K. and Volkwein, S., "Proper orthogonal decomposition for optimality systems," Tech. Rep. 10, Institute for Mathematics and Scientific Computing, University of Graz, 2006.

[11]Bui-Thanh, T., Willcox, K. E., Ghattas, O., and van Bloemen Waanders, B., "Goal-Oriented, Model-Constrained Optimization for Reduction of Large-Scale Systems," *Journal of Computational Physics*, Vol. 224, June 2007, pp. 880–896.

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

[13]Allgöwer, F. and Zheng, A., editors, *Nonlinear Model Predictive Control*, Birkhäuser, 2000.

[14]Rawlings, J. B., "Tutorial overview of model predictive control," *IEEE Control Systems Magazine*, Vol. 20, No. 3, 2000, pp. 38–52.

[15]Tøndel, P., *Constrained Optimal Control via Multiparametric Quadratic Programming*, Ph.D. thesis, Department of Engineering Cybernetics, Norw. University of Sci. and Techn., 2003.

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

[17]Tøndel, P., Johansen, T. A., and Bemporad, A., "Computation of piecewise affine control via binary search tree," *Automatica*, Vol. 39, No. 5, 2003, pp. 945–950.

[18]Adamjan, V., Arov, D., and Krein, M., "Analytic properties of Schmidt pairs for a Hankel operator and the generalized Schur-Takagi problem," *Math. USSR Sbornik*, Vol. 15, 1971, pp. 31–73.

[19]Bettayeb, M., Silverman, L., and Safonov, M., "Optimal Approximation of Continuous-Time Systems," *Proc. of the 19th IEEE Conf. on Decision and Control*, Vol. 1, 1980, pp. 195–198.

[20]Kung, S.-Y. and Lin, D., "Optimal Hankel-norm model reductions: Multivariable systems," *IEEE Trans. on Automatic Control*, Vol. AC-26, No. 1, 1971, pp. 832–852.

[21]Moore, B. C., "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Trans. on Automatic Control*, Vol. 26, No. 1, 1981, pp. 17–32.

[22]Sorensen, D. and Antoulas, A., "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.*, 2002, pp. 260–445.

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

[24]Li, J. and White, J., "Low rank solution of Lyapunov equations," *Journal on Matrix Analysis and Applications*, Vol. 24, No. 1, 2002, pp. 260–280.

[25]Benner, P., Quintana-Ortí, E. S., and Quintana-Ortí, G., "Balanced Truncation Model Reduction of Large-Scale Dense Systems on Parallel Computers," *Mathematical and Computer Modelling of Dynamical Systems*, Vol. 6, No. 4, 2000, pp. 383–405.

[26]Arian, E., Fahl, M., and Sachs, E., "Trust-region proper orthogonal decomposition for flow control," Tech. rep., Universitt Trier, 2000.

[27]Wortelboer, P. M. R., Steinbuch, M., and Bosgra, O. H., "Iterative model and controller reduction using closed-loop balancing, with application to a compact disc mechanism," *Int. J. Robust Nonlinear Control*, Vol. 9, No. 3, 1999, pp. 123–142.

[28]Borggaard, J., "Optimal reduced-order modeling for nonlinear distributed parameter systems," *American Control Conference*, 2006.

[29]Astrid, P., Huisman, L., Weiland, S., and Backx, A. C. P. M., "Reduction and Predictive Control Design for a Computational Fluid Dynamics Model," *Proc. 41st IEEE Conf. on Decision and Control*, Vol. 3, Las Vegas, NV, 2002, pp. 3378–3383.

[30]Muske, K. R. and Rawlings, J. B., "Model predictive control with linear models," *AIChE Journal*, Vol. 39, No. 2, 1993, pp. 262–287.

[31]Bemporad, A. and Morari, M., "Robust Model Predictive control: A survey," *Robustness in Identification and Control*, edited by A. Garulli, A. Tesi, and A. Vicino, No. 245 in Lecture Notes in Control and Information Sciences, Springer, 1999, pp. 207–226.

[32]Kothare, M., Balakrishnan, V., and Morari, M., "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, Vol. 32, No. 10, 1996, pp. 1361–1379.

[33]Imsland, L., Rossiter, J., Pluymers, B., and Suykens, J., "Robust Triple Mode MPC," *American Control Conference, 2006*, 2006, pp. 6.

[34]Heath, W., Wills, A., and Akkermans, J., "A sufficient stability condition for optimizing controllers with saturating actuators," *International Journal for Robust and Nonlinear Control*, Vol. 15, 2005, pp. 515–529.

[35]Heath, W. and Wills, A., "The inherent robustness of constrained linear model predictive control," *16th IFAC World Congress*, Prague, 2005.

[36]Heath, W., Li, G., Wills, A., and Lennox, B., "IQC analysis of linear constrained MPC," *IEEE sponsored Colloquium on Predictive Control*, University of Sheffield, 2005.

[37]Løvaas, C., Seron, M. M., and Goodwin, G. C., "Robust Model Predictive Control of Input-Constrained Stable Systems with Unstructured Uncertainty," *Proc. European Contr. Conf.*, 2007.

[38]Willcox, K. E. and Lassaux, G., "Model Reduction of an Actively Controlled Supersonic Diffuser," *Dimension Reduction of Large-Scale Systems*, edited by P. Benner, V. Mehrmann, and D. Sorensen, Vol. 45 of *Lecture Notes in Computational Science and Engineering*, Springer-Verlag, Berlin/Heidelberg, 2005, pp. 357–361.

[39]Willcox, K. E. and Megretski, A., "Fourier Series for Accurate, Stable, Reduced-Order Models in Large-Scale Applications," *SIAM Journal for Scientific Computing*, Vol. 26, No. 3, 2005, pp. 944–962.

[40]Antoulas, A. C., Sorensen, D. C., and Zhou, Y., "On the decay rate of Hankel singular values and related issues," *Systems & Control Letters*, Vol. 46, No. 5, 2002, pp. 323–342.

[41]Kerrigan, E. C. and Maciejowski, J., "Soft constraints and exact penalty functions in model predictive control," *Proc. UKACC International Conference (Control 2000)*, 2000.

[42]Hovland, S., *Soft Constraints in Explicit MPC*, Master's thesis, Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, 2004.