15th Triennial World Congress, Barcelona, Spain

EFFICIENT SOLUTION OF SECOND ORDER CONE PROGRAM FOR MODEL PREDICTIVE CONTROL [1]

Magnus Åkerblad and Anders Hansson

Division of Automatic Control,
Department of Electrical Engineering,
Linköping University, SE-581 83 Linköping, Sweden.
E-mail:pma@isy.liu.se, hansson@isy.liu.se

Abstract: In this paper it is shown how to efficiently solve an optimal control problem with applications to model predictive control. The objective is quadratic and the constraints can be both linear and quadratic. The key to an efficient implementation is to rewrite the optimization problem as a second order cone program. This can be done in many different ways. However, done carefully, it is possible to use both very efficient scalings as well as Riccati recursions for computing the search directions.

Keywords: Optimization, Second Order Cone Programming, Model Predictive Control

1. INTRODUCTION

Optimal control of discrete-time systems with constraints is used in many applications. Among others is Model Predictive Control (MPC), (Morari and Lee, 1999), where such a control problem is solved at each sampling instant. MPC has its origin in Dynamic Matrix Control (DMC) which was invented by engineers at Shell Oil in the early 1970's and presented in (Cutler and Ramaker, 1979). How DMC evolved to become what today is called MPC together with the current status of research in MPC is described in numerous surveys, e.g. (Garcia *et al.*, 1989; Morari and Lee, 1991; Morari and Lee, 1999; Muske and Rawlings, 1993; Qin and Badgwell, 1996; Ricker, 1991).

Recently specially tailored Interior Point (IP) methods applicable to MPC have appeared, (Gopal and Biegler, 1998). These algorithms solve the resulting Quadratic Program (QP) by utilizing the special structure of the control problem. By ordering the equations and variables in a certain way the linear system of equations that has to be solved for the search direction becomes block-diagonal, (Wright, 1993; Wright, 1996). By further examining this structure it is possible to solve the equations using a Riccati recursion. This makes the computational burden to grow only linearly with the time horizon, (Rao *et al.*, 1997). A similar approach is used by Steinbach, (Steinbach, 1994). Riccati recursions have also been used together with active set methods for solving the optimal control problem, (Arnold and Puta, 1994; Glad and Jonson, 1984). Comparisons between active set methods and IP methods have been done by several authors, (Albuquerque *et al.*, 1997; Biegler, 1997; Wright, 1996).

Recently, it has been shown that stability of MPCs can be established less conservatively if the optimization problem considered at each sampling interval is a Quadratically Constrained Quadratic Program (QCQP), (Lee, 2000; Lee and Kouvaritakis, 1999; Scokaert and Rawlings, 1998). This facilitates e.g. ellipsoidal terminal state domains. In addition piecewise quadratic terminal state weights can be used, (Löfberg, 2001). Robustness can also be addressed with QCQPs, (Hansson, 2000). The idea of using Riccati-recursions work also for QCQPs. However, it is not possible to use feasible IP-methods (Wright, 1997). Because of this no proof of polynomial complexity is available.

[1] Research supported by the Swedish Research Council for Engineering Sciences under contract No. 1999-169

A way to overcome this is to reformulate the QCQPs as Second Order Cone Programs (SOCPs), (Lobo *et al.*, 1998). The objective of this paper is to show how Riccati recursions can be used also in this context.

The remaining part of the paper is organized as follows. Section 2 presents the control problem and formulates it as an SOCP. In Section 3 the Karush-Kuhn-Tucker (KKT) conditions are stated and transformed by using a logarithmic barrier. In Section 4 it is discussed how to compute the search directions and how to scale them. In Section 5 it is shown how to efficiently solve the equations for the search directions by using a Riccati recursion. Then, in Section 6 feasibility is discussed. Finally, in Section 7 some concluding remarks are given.

2. CONTROL PROBLEM

In this section the control problem is described. First the model is presented and some constraints are presented. Then the performance measure is introduced. The optimization problem is reformulated as an SOCP.

Consider the following model and constraints for $k = 0, \dots, N-1$:

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k u_k, \\ z_k &= C_k x_k + D_k u_k, \\ b_k &\geq E_k x_k + F_k u_k, \\ 1 &\geq x_N^T W x_N, \end{aligned} \tag{1}$$

where $x_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^m$ is the control signal, $z_k \in \mathbb{R}^q$ is the output and where $A_k \in \mathbb{R}^{n \times n}$, $B_k \in \mathbb{R}^{n \times m}$, $C_k \in \mathbb{R}^{p \times n}$, $D_k \in \mathbb{R}^{p \times m}$, $E_k \in \mathbb{R}^{q \times n}$, $F_k \in \mathbb{R}^{q \times m}$, $b_k \in \mathbb{R}^q$ and $0 < W \in \mathbb{R}^{n \times n}$. The inequality in (1) should be interpreted as componentwise inequality. With abuse of notation we will denote both the given initial value and the state variable at time zero with x_0. The performance criterion to minimize is defined as

$$\phi = \sum_{k=0}^{N-1} z_k^T z_k + x_N^T P x_N,$$

where P is positive semidefinite. The performance criterion can easily be extended to handle piecewise quadratic end-point penalties by replacing $x_N^T P x_N$ with

$$\max_i \,(x_N^T P_i x_N - \rho_i).$$

This can be used to show stability for larger sets of initial values of x_0, see (Löfberg, 2001), but this will not be discussed further in this paper. The optimization problem can be reformulated as

$$\begin{aligned} \min_{t} \quad & \sum_{k=0}^{N} t_k \\ \text{subject to} \quad & Hy = d, \\ & My \leq b, \\ & z_k^T z_k \leq t_k, \; k = 0, \dots, N-1, \\ & x_N^T P x_N \leq t_N, \\ & x_N^T W x_N \leq 1, \end{aligned} \tag{2}$$

where $y^T = \begin{bmatrix} x_0^T \; u_0^T \; \dots \; x_{N-1}^T \; u_{N-1}^T \; x_N^T \end{bmatrix}$, $d^T = \begin{bmatrix} x_0^T \; 0 \; \dots \; 0 \end{bmatrix}$ and

$$H = \begin{bmatrix} I & 0 & 0 & \dots & 0 \\ -A_0 & -B_0 & I & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & -A_{N-1} & -B_{N-1} & I \end{bmatrix},$$

$$M = \begin{bmatrix} \oplus_{k=0}^{N-1} \begin{bmatrix} E_k \; F_k \end{bmatrix} \; 0 \end{bmatrix}.$$

The reason for having one quadratic constraint for each k instead of one for all k is explained in Section 5. To be able to solve the optimization problem efficiently we will formulate it as an SOCP. To this end define

$$\hat{L}_k = \begin{bmatrix} -1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & -2C_k & -2D_k \end{bmatrix},$$

$$c_k^T = \begin{bmatrix} 1 \; 1 \; 0 \end{bmatrix}, \quad k = 0, \dots, N, \quad c_{N+1}^T = \begin{bmatrix} 1 \; 0 \end{bmatrix},$$

$$\hat{P} = \begin{bmatrix} -1 & 0 \\ 1 & 0 \\ 0 & -2P^{\frac{1}{2}} \end{bmatrix}, \; \hat{W} = \begin{bmatrix} 0 & 0 \\ 0 & -W^{\frac{1}{2}} \end{bmatrix}, \; f = \begin{bmatrix} e_0^T \\ \vdots \\ e_0^T \end{bmatrix},$$

where $e_0 \in \mathbb{R}^{n+m+1}$ is the first unit vector. Then the optimization problem can be written as the SOCP

$$\begin{aligned} \min_{y,\,s} \quad & f^T y \\ \text{subject to} \quad & s_k = c_k - \hat{L}_k \hat{y}_k, \quad k = 0, \dots, N-1, \\ & s_N = c_N - \hat{P} \hat{y}_N, \\ & s_{N+1} = c_{N+1} - \hat{W} \hat{y}_N, \\ & r = b - \hat{M} \hat{y}, \\ & 0 = d - \hat{H} \hat{y}, \\ & \|s_{k,1}\| \leq s_{k,0}, \quad k = 0, \dots, N+1, \\ & 0 \leq r, \end{aligned} \tag{3}$$

where $\hat{y}_k^T = \begin{bmatrix} t_k \; x_k^T \; u_k^T \end{bmatrix}$, $\hat{y}_N^T = \begin{bmatrix} t_N \; x_N^T \end{bmatrix}$ and where $\hat{M}$ and $\hat{H}$ are modified versions of H and M with zero columns where t_k comes in. We use the notation $s_k = \begin{bmatrix} s_{k,0} \\ s_{k,1} \end{bmatrix}$, where $s_{k,0} \in \mathbb{R}$.

3. KARUSH-KUHN-TUCKER CONDITIONS

In this section we will discuss how to solve the SOCP that was presented in the previous section. The KKT conditions for the SOCP are

$$\begin{bmatrix} 0 & 0 & 0 & \hat{H} & 0 & 0 \\ 0 & 0 & 0 & \hat{L} & I & 0 \\ 0 & 0 & 0 & \hat{M} & 0 & I \\ \hat{H}^T & \hat{L}^T & \hat{M}^T & 0 & 0 & 0 \\ 0 & S & 0 & 0 & \Lambda & 0 \\ 0 & 0 & R & 0 & 0 & \Gamma \end{bmatrix} \begin{bmatrix} \rho \\ \lambda \\ \gamma \\ \hat{y} \\ s \\ r \end{bmatrix} = \begin{bmatrix} d \\ \hat{c} \\ b \\ f \\ 0 \\ 0 \end{bmatrix}, \qquad (4)$$

together with the cone constraints

$$\begin{aligned} s_k \geq_{K_k} 0, \quad \lambda_k \geq_{K_k} 0, \quad k = 0, \ldots, N+1, \\ r \ \geq \ 0, \quad \gamma \geq 0, \end{aligned}$$

where $\hat{y}$, r and s are the primal variables and λ, γ and ρ are the associated dual variables. The notation $s_k \geq_{K_k} 0$ is equivalent to $\|s_{k,1}\| \leq s_{k,0}$. The matrices S, Λ and $\hat{L}$ are defined as

$$\begin{aligned} \hat{L} &= \begin{bmatrix} \oplus_{k=0}^{N-1} \hat{L}_k \oplus \hat{P} \\ [0 \ \hat{W}] \end{bmatrix}, \quad \hat{c}^T = \begin{bmatrix} c_0^T \ \ldots \ c_{N+1}^T \end{bmatrix}, \\ \Lambda &= \text{block diag}_{k=0,\ldots,N+1} \ \Lambda_k, \\ S &= \text{block diag}_{k=0,\ldots,N+1} \ S_k, \end{aligned}$$

where S_k and Λ_k are arrow-matrices defined as

$$S_k = \begin{bmatrix} s_{k0} & s_{k1}^T \\ s_{k1} & s_{k0} I \end{bmatrix}, \quad \Lambda_k = \begin{bmatrix} \lambda_{k0} & \lambda_{k1}^T \\ \lambda_{k1} & \lambda_{k0} I \end{bmatrix}.$$

Furthermore $R = \text{diag}(r)$, $\Gamma = \text{diag}(\gamma)$ and $e_0 \in \mathbb{R}^{p+1}$. For a unified treatment of quadratic and linear constraints, see (Alizadeh and Schmieta, 1997). Under certain regularity condition the KKT conditions are necessary and sufficient for the SOCP. Specifically a solution to the KKT condition is also a solution to the SOCP.

Had it not been for the cone constraints, a straight forward application of Newton's method to solve the equality condition would have been sufficient. The key idea in IP methods is to remove the cone constraints by using a logarithmic barrier transformation, see (Alizadeh and Schmieta, 1997). This will result in a relaxation of the complementary slackness condition

$$\begin{aligned} S_k \Lambda_k e_0 &= \mu e_0, \quad k = 0, \ldots, N+1, \\ R\Gamma &= \mu \mathbf{1}, \end{aligned}$$

where $\mu > 0$, and hence modify the righthandside of (4) to

$$\begin{bmatrix} d^T \ \hat{c}^T \ b^T \ f^T \ 2\mu\hat{e}_0^T \ \mu\mathbf{1}^T \end{bmatrix}^T, \qquad (5)$$

where $\hat{e}_0^T = \begin{bmatrix} e_0^T \ \ldots \ e_0^T \end{bmatrix}$, and where $e_0 \in \mathbb{R}^{p+1}$. Now the cone constraints are only implicitly present, and we can apply Newton's method, assuming that the initial point is strictly feasible, i.e. satisfies the cone constraints with strict inequality. By gradually decreasing μ the original complementary slackness condition will be recovered. The step-length in Newton's method is chosen such that the cone constraints are never violated. In this paper a potential-reduction method is used to determined the step-length, see (Vandenberghe and Boyd, 1995).

4. SEARCH DIRECTION

In this section the primal-dual potential function will be introduced and the resulting equations for the search direction will be obtained followed by a discussion on how to obtain the Nesterov-Todd search direction for the SOCP.

First define the barrier functions as

$$\begin{aligned} \Upsilon_L(r_k) &= -\ln(r_k), \\ \Upsilon_S(s_k) &= -\ln(s_{k0}^2 - \|s_{k,1}\|), \end{aligned}$$

and then define the primal-dual potential function as

$$\begin{aligned} \Psi = \nu \ln(\lambda^T s + \gamma^T r) + \sum_{k=0}^{N+1} \Upsilon_S(s_k) + \Upsilon_S(\lambda_k) \\ + \sum_{k=1}^{Nq} \Upsilon_L(r_k) + \Upsilon_L(\gamma_k) \\ - (2(N+2) + Nq) \ln(2(N+2) + Nq), \qquad (6) \end{aligned}$$

where ν is a design parameter which determines the relative weight between duality gap and centrality. Since (6) has a concave term a modification of Newton's method is applied to (6), where the second derivative of the concave term is ignored. This leads to the following equation for the search direction

$$\begin{bmatrix} 0 & 0 & 0 & \hat{H} & 0 & 0 \\ 0 & 0 & 0 & \hat{L} & I & 0 \\ 0 & 0 & 0 & \hat{M} & 0 & I \\ \hat{H}^T & \hat{L}^T & \hat{M}^T & 0 & 0 & 0 \\ 0 & \Theta^2 & 0 & 0 & I & 0 \\ 0 & 0 & \Xi^2 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \Delta\rho \\ \Delta\lambda \\ \Delta\gamma \\ \Delta\hat{y} \\ \Delta s \\ \Delta r \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\nabla_\lambda \Psi \\ -\nabla_\gamma \Psi \end{bmatrix}, \quad (7)$$

where Θ^2 and Ξ^2 are the modified Hessians of Ψ with respect to λ and γ, respectively. To be able to solve (3) efficiently a scaling can be introduced. When introducing a scaling the problem must be kept scaling invariant. A popular scaling is the Nesterov-Todd(NT) scaling, see (Tsuchiya, 1998). It has been shown that primal-dual algorithms using the NT scaling have polynomial complexity. Let the NT scaling matrix for the cone constraints be defined as

$$\begin{aligned} G_k^{-1}(\lambda_k, s_k) &= \begin{bmatrix} \bar{\alpha}_k & -\bar{\beta}_{k,0} & -\bar{\beta}_{k,1}^T \\ -\bar{\beta}_{k,0} & \Sigma_{k,11} & \Sigma_{k,12} \\ -\bar{\beta}_{k,1} & \Sigma_{k,21} & \Sigma_{k,22} \end{bmatrix} \\ &= \frac{1}{\omega_k} \begin{bmatrix} \alpha_k & -\beta_k^T \\ -\beta_k & I + \dfrac{\beta_k \beta_k^T}{1 + \alpha_k} \end{bmatrix}, \end{aligned}$$

$$\begin{aligned}
\alpha_k &= \frac{\zeta_{k0}}{\sqrt{\zeta_{k0}^2 - \zeta_{k1}^T \zeta_{k1}}}, \quad \beta_k = \frac{\zeta_{k1}}{\sqrt{\zeta_{k0}^2 - \zeta_{k1}^T \zeta_{k1}}}, \\
\zeta_k &= (\zeta_{k0}, \zeta_{k1}) = (\bar{s}_{k0} + \bar{\lambda}_{k0}, \bar{s}_{k1} - \bar{\lambda}_{k1}), \\
\bar{\lambda_k} &= (\bar{\lambda}_{k0}, \bar{\lambda}_{k1}) = \omega_k^{-1}(\lambda_{k0}, \lambda_{k1}), \\
\bar{s_k} &= (\bar{s}_{k0}, \bar{s}_{k1}) = \omega(s_{k0}, s_{k1}), \\
\omega_k &= \left[\frac{s_{k0}^2 - \| s_{k1} \|^2}{\lambda_{k0}^2 - \| \lambda_{k1} \|^2} \right]^{1/4} .
\end{aligned}$$

For the linear inequalities the scaling matrix is defined as

$$\Sigma_2 = \operatorname{diag} \left(\frac{r_k}{\gamma_k} \right)^{1/2} .$$

Let G_k and Σ_2 operate on the unscaled problem as follows

$$\begin{aligned}
\tilde{L}_k &= G_k^{-1} \hat{L}_k, \quad k = 0, \ldots, N-1, \\
\tilde{P} &= G_N^{-1} \hat{P}, \quad \tilde{W} = G_{N+1}^{-1} \hat{W}, \\
\tilde{M} &= \Sigma_2^{-1} \hat{M}, \\
\tilde{\lambda}_k &= G_k^T \lambda_k, \quad \tilde{s}_k = G_k^{-1} s_k, \quad k = 0, \ldots, N+1, \\
\tilde{\gamma} &= \Sigma_2^T \gamma, \quad \tilde{r} = \Sigma_2^{-1} r, \\
\tilde{L} &= \begin{bmatrix} \oplus_{k=0}^{N-1} \tilde{L}_k \oplus \tilde{P} \\ [0 \ \tilde{W}] \end{bmatrix} .
\end{aligned}$$

The NT scaling matrix maps the primal and dual variable to the same vector v, which is given by

$$\begin{aligned}
v_k &= G_k^T \lambda_k = G_k^{-1} s_k, \quad k = 0, \ldots, N+1, \\
v_{N+2} &= \Sigma_2^T \gamma = \Sigma_2^{-1} r.
\end{aligned}$$

With this scaling the following equations for the search direction is obtained

$$\begin{bmatrix} 0 & 0 & 0 & \hat{H} & 0 & 0 \\ 0 & 0 & 0 & \tilde{L} & I & 0 \\ 0 & 0 & 0 & \tilde{M} & 0 & I \\ \hat{H}^T & \tilde{L}^T & \tilde{M}^T & 0 & 0 & 0 \\ 0 & I & 0 & 0 & I & 0 \\ 0 & 0 & I & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \Delta\rho \\ \Delta\tilde{\lambda} \\ \Delta\tilde{\gamma} \\ \Delta\hat{y} \\ \Delta\tilde{s} \\ \Delta\tilde{r} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ w_1 \\ w_2 \end{bmatrix}, \qquad (8)$$

where $w_1 = (\nu + 2N(1 - \ln 2) + Nq)v - 2V^{-1}\hat{e}_0$, $w_2 = (\nu + 2N(1 - \ln 2) + Nq)v_{N+2} - V_{N+2}^{-1}\mathbf{1}$,

$$V = \text{block diag}_{k=0,\ldots,N+1} V_k,$$

and where V_k are arrow-matrices defined by v_k for $k = 0, \ldots, N+1$. Finally, $V_{N+2} = \operatorname{diag}(v_{N+2})$.

5. EFFICIENT SOLUTION OF EQUATIONS FOR SEARCH DIRECTION

In this section it is shown how a Riccati recursion efficiently can be used to solve the equations for the search direction obtained in Section 4. From the two last block-rows of (8) it follows that $\Delta\tilde{s} = -\Delta\tilde{\lambda} + w_1$ and $\Delta\tilde{r} = -\Delta\tilde{\gamma} + w_2$. By substituting this back into (8) the following equation is obtained

$$\begin{bmatrix} 0 & 0 & 0 & \hat{H} \\ 0 & -I & 0 & \tilde{L} \\ 0 & 0 & -I & \tilde{M} \\ \hat{H}^T & \tilde{L}^T & \tilde{M}^T & 0 \end{bmatrix} \begin{bmatrix} \Delta\rho \\ \Delta\tilde{\lambda} \\ \Delta\tilde{\gamma} \\ \Delta\hat{y} \end{bmatrix} = \begin{bmatrix} 0 \\ -w_1 \\ -w_2 \\ 0 \end{bmatrix} . \qquad (9)$$

Reorder the variables in Δy as

$$\begin{bmatrix} \Delta t_0 \ \ldots \ \Delta t_N \ \Delta x_0^T \ \Delta u_0^T \ \ldots \ \Delta x_{N-1}^T \ \Delta u_{N-1}^T \ \Delta x_N^T \end{bmatrix}$$

and partition the dual variable as $\Delta\tilde{\lambda}_k^T = \begin{bmatrix} \Delta\tilde{\lambda}_{kt}^T \ \Delta\tilde{\lambda}_{ky}^T \end{bmatrix}$, where $\Delta\tilde{\lambda}_{kt}$ contains the first two elements of $\Delta\tilde{\lambda}_k$. Now define

$$\Delta\tilde{\lambda}_t = \begin{bmatrix} \Delta\tilde{\lambda}_{0,t} \\ \vdots \\ \Delta\tilde{\lambda}_{N+1,t} \end{bmatrix}, \quad \Delta\hat{\lambda} = \begin{bmatrix} \Delta\tilde{\lambda}_{0,y} \\ \vdots \\ \Delta\tilde{\lambda}_{N+1,y} \end{bmatrix} .$$

Then with some reordering of rows, (9) can be reformulated as

$$\begin{bmatrix} \bar{\alpha} & -I & 0 & 0 & 0 & \Omega L \\ 0 & \bar{\alpha}^T & 0 & \bar{\beta}^T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & H \\ \bar{\beta} & 0 & 0 & -I & 0 & \Sigma L \\ 0 & 0 & 0 & 0 & -I & \Sigma_2 M \\ 0 & L^T\Omega^T & H^T & L^T\Sigma^T & M^T\Sigma_2^T & 0 \end{bmatrix} \begin{bmatrix} \Delta t \\ \Delta\tilde{\lambda}_t \\ \Delta\rho \\ \Delta\hat{\lambda} \\ \Delta\gamma \\ \Delta y \end{bmatrix} = \begin{bmatrix} -w_t \\ 0 \\ 0 \\ -w_q \\ -w_2 \\ 0 \end{bmatrix}, \qquad (10)$$

where $\begin{bmatrix} w_t^T \ w_q^T \end{bmatrix} = w_1^T$, $\bar{\beta} = \text{block diag}(\bar{\beta}_{k,1} + \Sigma_{k,21})$, $\Sigma = \text{block diag}(\Sigma_{k,22})$, and where

$$\begin{aligned}
\bar{\alpha} &= \text{blockdiag} \left(\begin{bmatrix} -\bar{\alpha}_k - \bar{\beta}_{k,0} \\ \bar{\beta}_{k,0} + \Sigma_{k,11} \end{bmatrix} \right), \\
\Omega &= \text{block diag} \left(\begin{bmatrix} -\bar{\beta}_k \\ \Sigma_{k,12} \end{bmatrix} \right), \\
L &= \begin{bmatrix} -2C_0 & -2D_0 & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ 0 & \ldots & -2C_{N-1} & -2D_{N-1} & 0 \\ 0 & \ldots & 0 & 0 & -2P^{1/2} \\ 0 & \ldots & 0 & 0 & -W^{1/2} \end{bmatrix},
\end{aligned}$$

From block-rows 4 and 5 of (10) it follows that

$$\begin{aligned}
\Delta\hat{\lambda} &= \Sigma L \Delta y + \bar{\beta}\Delta t + w_q, \\
\Delta\hat{\gamma} &= \Sigma_2 M \Delta y + w_2 .
\end{aligned}$$

By substituting this back into (10) the following equation is obtained

$$\begin{bmatrix} \bar{\beta}^T\bar{\beta} & \bar{\alpha}^T & \bar{\beta}^T\Sigma L & 0 \\ \bar{\alpha} & -I & \Omega L & 0 \\ L^T\Sigma^T\bar{\beta} & L^T\Omega^T & Q & H^T \\ 0 & 0 & H & 0 \end{bmatrix} \begin{bmatrix} \Delta t \\ \Delta\lambda_t \\ \Delta y \\ \Delta\rho \end{bmatrix} = \begin{bmatrix} -\bar{\beta}^T w_q \\ -w_t \\ -\hat{w} \\ 0 \end{bmatrix}, \qquad (11)$$

where

$$\begin{aligned}
Q &= M^T\Sigma_2^T\Sigma_2 M + L^T\Sigma^T\Sigma L, \\
\hat{w} &= L^T\Sigma^T w_q + M^T\Sigma_2^T w_2 .
\end{aligned}$$

Now partition (11) as

$$\begin{bmatrix}\Phi_1 & \Phi_{12}\\ \Phi_{12}^T & \Phi_2\end{bmatrix}\begin{bmatrix}\Delta_1\\ \Delta_2\end{bmatrix} = \begin{bmatrix}U_1\\ U_2\end{bmatrix}. \qquad (12)$$

The partitioning is done so that Φ_1 is a $(3(N+1)+1)\times(3(N+1)+1)$ block matrix which easily can be inverted because of the block diagonal structure of $\bar{\alpha}$ and $\bar{\beta}$. Then (12) can be written

$$\begin{aligned}\Delta_1 &= (\Phi_1 - \Phi_{12}\Phi_2^{-1}\Phi_{12}^T)^{-1}(U_1 - \Phi_{12}\Phi_2^{-1}U_2), \quad (13)\\ \Delta_2 &= \Phi_2^{-1}(U_2 - \Phi_{12}^T\Delta_1).\end{aligned}$$

Since Q has a block-diagonal structure and H is built from the dynamic system with a very specific structure, $\Phi_2^{-1}U_2$ can be efficiently computed using a Riccati recursion approach as in (Wright, 1993). Had we replaced the N quadratic constraints $z_k^T z_k \leq t_k$ with one constraint $\sum_{k=0}^{N-1} z_k^T z_k \leq t$, then Q would not have had the desired structure and the Riccati recursion could not have been applied. By using the matrix inversion lemma it holds that

$$\begin{aligned}(\Phi_1 - \Phi_{12}\Phi_2^{-1}\Phi_{12}^T)^{-1} &= \Phi_1^{-1} - \Phi_1^{-1}\Phi_{12}\\ &\times(\Phi_{12}^T\Phi_1^{-1}\Phi_{12} - \Phi_2)^{-1}\Phi_{12}^T\Phi_1^{-1}. \quad (14)\end{aligned}$$

It can be shown that $(\Phi_{12}^T\Phi_1^{-1}\Phi_{12} - \Phi_2)$ has the same structure as Φ_2 and thus can be factorized by a Riccati recursion as well. Substituting (14) into (13) we realize that the rest of the algorithm can be divided into the following steps

1. calculate $\eta_1 = \Phi_2^{-1}U_2$ using a Riccati recursion
2. let $a = U_1 - \Phi_{12}\eta_1$
3. let $\xi = \Phi_{12}^T\Phi_1^{-1}a$
4. solve $(\Phi_{12}^T\Phi_1^{-1}\Phi_{12} - \Phi_2)\eta_2 = \xi$ using a Riccati recursion
5. now $\Delta_1 = \Phi_1^{-1}a - \Phi_1^{-1}\Phi_{12}\eta_2$
6. then $\Delta_2 = \eta_1 - \Phi_2^{-1}\Phi_{12}\Delta_1$ where the second term also can be computed using a Riccati recursion

Notice that in Step 6 parts of the Riccati recursion used in Step 1 can be reused. For detailed information on the Riccati factorization approach, see (Rao *et al.*, 1997). The computational complexity grows linearly with the time horizon N.

6. STRICTLY FEASIBLE INITIAL POINTS

In this section it will be shown how to obtain strictly feasible initial primal and dual points. It will also be shown when there exist solutions to (1). To simplify, we will look at a special case of constraints

$$\varpi_k \leq \mathcal{E}_k x_k + \mathcal{F}_k u_k \leq \varsigma_k,$$

where $\mathcal{F}_k$ has full column rank. This can be written as in (1) with

$$E_k = \begin{bmatrix}\mathcal{E}_k\\ -\mathcal{E}_k\end{bmatrix}, \quad F_k = \begin{bmatrix}\mathcal{F}_k\\ -\mathcal{F}_k\end{bmatrix}, b_k = \begin{bmatrix}\varpi_k\\ -\varsigma_k\end{bmatrix} \quad (15)$$

6.1 Dual Initial Point

A strictly feasible dual point (ρ, λ, γ) has to satisfy block row 4 of (4), $\lambda_k >_{K_k} 0$ and $\gamma > 0$. By choosing $\lambda_{k1} = 0$ and $\lambda_{k0} = 1$, $N+1$ rows of block row 4 in (4) will be satisfied. The remaining rows are trivially satisfied by $\rho = 0$ and $\gamma = \mathbf{1}$.

6.2 Primal Initial Point

To look for a strictly feasible primal point consider the following problem

$$\begin{aligned}\min_{\tau} \quad & \tau \qquad (16)\\ \text{subject to } \quad & Hy = d,\\ & My \leq b + \tau,\\ & x_N^T W x_N \leq 1 + \tau.\end{aligned}$$

When (16) has a solution $\tau < 0$ a strictly feasible primal point to the original problem can be found. The corresponding dual problem can be stated as follows

$$\begin{aligned}\max_{\rho,\gamma,\lambda} \quad & -d^T\rho - b^T\gamma - c^T\lambda \qquad (17)\\ \text{subject to } \quad & \hat{W}^T\lambda + \hat{M}^T\gamma + \hat{H}^T\rho = -f,\\ & \|\ \lambda_1\ \| \leq \lambda_0,\\ & 0 \leq \gamma,\end{aligned}$$

where $\hat{H} = [0\ H]$ and where

$$\begin{aligned}\hat{M} &= \left[\oplus_{k=0}^{N-1}\left[-1\ E_k\ F_k\right]\ 0\ 0\right],\\ \hat{W} &= \begin{bmatrix}-1 & 0 & 0\\ -1 & 0 & 0\\ 0 & 0 & -2W^{\frac{1}{2}}\end{bmatrix},\\ c^T &= \begin{bmatrix}2 & 0 & 0\end{bmatrix}, \quad f^T = \begin{bmatrix}1 & 0 & \dots & 0\end{bmatrix}.\end{aligned}$$

A primal-dual algorithm can be used to determine if the original problem is feasible or not. More precisely there exist three different cases of solutions to (16) and (17)

- $-d^T\rho - b^T\gamma - c^T\lambda > 0$. There exists no strictly feasible solution to the original problem
- $\tau < 0$. There exists a strictly feasible solution to the original problem
- $\tau = -d^T\rho - b^T\gamma - c^T\lambda = 0$. There exist a feasible solution to the original problem, however not strictly feasible

Notice that this so called Phase 1 problem also can be reformulated as an SOCP as in Section 2, and thus it also can be solved using a similarly approach.

7. CONCLUSIONS

In this paper it has been shown how to efficiently solve an optimal control problem with applications to model predictive control. Special attention has been given to formulate the problem as an SOCP while still preserving certain structure under scalings. Preserving this structure makes it possible to efficiently solve the equations for the search directions using Riccati-recursions. The computational complexity grows linearly with the time horizon. It has also been shown that the algorithm in an easy way can determine when a problem is feasible.

REFERENCES

Albuquerque, J. S., V. Gopal, G. H. Staus, L. T. Biegler and B. E. Ydstie (1997). Interior point SQP strategies for structured process optimization problems. *Computers Chem. Engng.* **21, Suppl.**, S853–S859.

Alizadeh, F. and S. Schmieta (1997). Optimization with semidefinite quadratic and linear constraints. Technical report RRR 23-97, RUTCOR Research Report. Rutgers University, NJ, USA.

Arnold, E. and H. Puta (1994). An SQP-type solution method for constrained discrete-time optimal control problems. In: *Computational Optimal Control* (R. Bulirsch and D. Kraft, Eds.). Vol. 115 of *International Series of Numerical Mathematics*. pp. 127–136. Birkhäuser Verlag. Basel.

Biegler, L. T. (1997). Advances in nonlinear programming concepts for process control. *Journal of Process Control*.

Cutler, C. R. and B. L. Ramaker (1979). Dynamic matrix control—a computer control algorithm. In: *Proceedings of the AIChE National Meeting*. Huston, Texas.

Garcia, C. E., D. M. Prett and M. Morari (1989). Model predictive control: Theory and practice—a survey. *Automatica* **3**, 335–348.

Glad, T. and H. Jonson (1984). A method for state and control constrained linear quadratic control problems. In: *Proceedings of the 9th IFAC World Congress*. Budapest, Hungary.

Gopal, V. and L. T. Biegler (1998). Large scale inequality constrained optimization and control. *IEEE Control Systems Magazine* **18**(6), 59–68.

Hansson, A. (2000). A primal-dual interior-point method for robust optimal control of linear discrete-time systems. *IEEE Transactions on Automatic Control* **45**(9), 1639–1655.

Lee, J-W. (2000). Exponential stability of constrained receding horizon control with terminal ellipsiod constraints. *IEEE Transactions on Automatic Control* **45**(1), 83–88.

Lee, Y.I. and B. Kouvaritakis (1999). Stabilizable regions of receding horizon predictive control with input constraints. *Systems and Control Letters* **38**, 13–20.

Lobo, M., L. Vandenberghe, S. Boyd and H. Lebret (1998). Applications of second-order cone programming. *Linear Algebra and its Applications* **284**, 193–228.

Löfberg, J. (2001). Linear model predictive control stability and robustness. Licentiate Thesis No. 866, Linköpings universitet.

Morari, M. and J. H. Lee (1991). Model predictive control: The good, the bad, and the ugly. In: *Chemical Process Control—CPC IV* (Y. Arkun and W. H. Ray, Eds.). Elsevier. Amsterdam. pp. 419–444.

Morari, M. and J. H. Lee (1999). Model predictive control: Past, present and future. *Computers and Chemical Engineering* **23**, 667–682.

Muske, K. R. and J. B. Rawlings (1993). Model predictive control with linear models. *Process Systems Engineering, AIChE Journal* **39**(2), 261–287.

Qin, S. J. and T. A. Badgwell (1996). An overview of industrial predictive control technology. In: *Chemical Process Control–V, Assessment and New Directions for Research*. Tahoe City, CA.

Rao, C. V., S. J. Wright and J. B. Rawlings (1997). Application of interior-point methods to model predictive control. Preprint ANL/MCS-P664-0597. Mathematics and Computer Science Division, Argonne National Laboratory.

Ricker, N. L. (1991). Model predictive control: State of the art. In: *Chemical Process Control—CPC IV* (Y. Arkun and W. H. Ray, Eds.). Elsevier. Amsterdam. pp. 271–296.

Scokaert, P.O.M. and J.B. Rawlings (1998). Constained linear quadratic control. *IEEE Transactions on Automatic Control* **43**(8), 1163–1169.

Steinbach, M. C. (1994). A structured interior point SQP method for nonlinear optimal control problems. In: *Computational Optimal Control* (R. Bulirsch and D. Kraft, Eds.). Vol. 115 of *International Series of Numerical Mathematics*. pp. 213–222. Birkhäuser Verlag. Basel.

Tsuchiya, T. (1998). A convergence analysis of the scaling-invariant primal-dual path-following algorithms for second-order cone programming. Technical report. The Institute of Statistical Mathematics, Tokyo, Japan.

Vandenberghe, L. and S. Boyd (1995). A primal-dual potential reduction method for probelms involving matrix inequalities. *Mathematical Programming* **69**, 205–236.

Wright, S. J. (1993). Interior-point methods for optimal control of discrete-time systems. *J. Optim. Theory Appls.* **77**, 161–187.

Wright, S. J. (1996). Applying new optimization algorithms to model predictive control. *Chemical Process Control-V*.

Wright, S. J. (1997). *Primal-Dual Interior-Point Methods*. SIAM.