Mathematical Programming manuscript No.
(will be inserted by the editor)

Paul J. Goulart · Eric C. Kerrigan ·
Daniel Ralph

Efficient Robust Optimization for Robust Control with Constraints

Received: 30 September 2005 / Accepted: 4 December 2006

Abstract This paper proposes an efficient computational technique for the optimal control of linear discrete-time systems subject to bounded disturbances with mixed linear constraints on the states and inputs. The problem of computing an optimal state feedback control policy, given the current state, is non-convex. A recent breakthrough has been the application of robust optimization techniques to reparameterize this problem as a convex program. While the reparameterized problem is theoretically tractable, the number of variables is quadratic in the number of stages or horizon length N and has no apparent exploitable structure, leading to computational time of $\mathcal{O}(N^6)$ per iteration of an interior-point method. We focus on the case when the disturbance set is ∞-norm bounded or the linear map of a hypercube, and the cost function involves the minimization of a quadratic cost. Here we make use of state variables to regain a sparse problem structure that is related to the structure of the original problem, that is, the policy optimization problem may be decomposed into a set of coupled finite horizon control problems. This decomposition can then be formulated as a highly structured quadratic program, solvable by primal-dual interior-point methods in which each iteration requires $\mathcal{O}(N^3)$ time. This cubic iteration time can be guaranteed using a Riccati-based block factorization technique, which is standard in discrete-time optimal control. Numerical results are presented, using a standard sparse primal-dual interior point solver, that illustrate the efficiency of this approach.

P. Goulart
Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, United Kingdom. E-mail: pgoulart@alum.mit.edu

E. Kerrigan
Department of Aeronautics and Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Road, London SW7 2AZ, United Kingdom. E-mail: e.kerrigan@imperial.ac.uk

D. Ralph
Judge Business School, University of Cambridge, Trumpington Street, Cambridge CB2 1AG, United Kingdom. E-mail: d.ralph@jbs.cam.ac.uk

Keywords Constrained control · robust optimization · optimal control · robust control · receding horizon control · predictive control

1 Introduction

Robust and predictive control

This paper is concerned with the efficient computation of optimal control policies for constrained discrete-time linear systems subject to bounded disturbances on the state. In particular, we consider the problem of finding, over a finite horizon of length N, a *feedback policy*

$$\pi := \{\mu_0(\cdot), \dots, \mu_{N-1}(\cdot)\} \tag{1}$$

for a discrete-time linear dynamical system of the form

$$x_{i+1} = Ax_i + Bu_i + w_i \tag{2}$$

$$u_i = \mu_i(x_0, \dots, x_i) \tag{3}$$

which guarantees satisfaction of a set of mixed constraints on the states and inputs at each time, for all possible realizations of the disturbances w_i, while minimizing a given cost function.

The states x_i and inputs u_i are constrained to lie in a compact and convex set $\mathcal{Z}$, i.e.

$$(x_i, u_i) \in \mathcal{Z}, \quad \forall i \in \{0, 1, \dots, N-1\} \tag{4}$$

with an additional terminal constraint $x_N \in X_f$. We assume nothing about the disturbances other than that they lie in a given compact set W.

The above, rather abstract problem is motivated by the fact that for many real-life control applications, optimal operation nearly always occurs on or close to some constraints [41]. These constraints typically arise, for example, due to actuator limitations, safe regions of operation, or performance specifications. For safety-critical applications, it is crucial that some or all of these constraints are met despite the presence of unknown disturbances.

Because of its importance, the above problem and derivations of it have been studied for some time now, with a large body of literature that falls under the broad banner of “robust control” (see [7, 53] for some seminal work on the subject). The field of *linear* robust control, which is mainly motivated by frequency-domain performance criteria [57] and does *not* explicitly consider time-domain constraints as in the above problem formulation, is considered to be mature and a number of excellent references are available on the subject [19, 29, 58]. In contrast, there are few tractable, non-conservative solutions to the above constrained problem, even if all the constraint sets are considered to be polytopes or ellipsoids; see, for example, the literature on set invariance theory [9] or ℓ_1 optimal control [14, 21, 49, 52].

A control design method that is particularly suitable for the synthesis of controllers for systems with constraints is predictive control [12, 41]. Predictive control is a family of optimal control techniques where, at each time instant, a finite-horizon constrained optimal control problem is solved using tools from mathematical programming. The solution to this optimization problem is usually implemented in a receding horizon fashion, i.e. at each time instant, a measurement of the system is obtained, the associated optimization problem is solved and only the first control input in the optimal policy is implemented. Because of this ability to solve a sequence of complicated, *constrained* optimal control problems in real-time, predictive control is synonymous with “advanced control” in the chemical process industries [45].

The theory on predictive control *without* disturbances is relatively mature and most of the fundamental problems are well-understood. However, despite recent advances, there are many open questions remaining in the area of *robust* predictive control [4, 42, 43]. In particular, efficient optimization methods have to be developed for solving the above problem before robust predictive control methods can be applied to unstable or safety-critical applications in areas such as aerospace and automotive applications [50].

Robust control models

The core difficulty with the problem (1)–(4) is that optimizing the feedback policy π over *arbitrary* nonlinear functions is extremely difficult, in general. Proposals which take this approach, such as those based on robust dynamic programming [3, 15], or those based on enumeration of extreme disturbance sequences generated from the set W, as in [47], are typically intractable for all but the smallest problems. Conversely, optimization over *open-loop* control sequences, while tractable, is considered unacceptable since problems of infeasibility or instability may easily arise [43].

An obvious sub-optimal proposal is to parameterize the control policy π in terms of *affine* functions of the sequence of states, i.e. to parameterize the control sequence as

$$u_i = g_i + \sum_{j=0}^{i} L_{i,j} x_j \tag{5}$$

where the matrices $L_{i,j}$ and vectors g_i are decision variables. However, the set of constraint admissible policies of this form is easily shown to be *non-convex* in general. As a result, most proposals that take this approach [2, 13, 37, 38, 44] fix a stabilizing feedback gain K, then parameterize the control sequence as $u_i = Kx_i + g_i$ and optimize the design parameters g_i. Though tractable, this approach is problematic since it is unclear how one should select the gain K to minimize conservativeness.

A recent discovery [5, 28] showed that the problem of optimizing over state feedback policies of the form (5) is equivalent to the problem of optimizing over disturbance feedback policies of the form

$$u_i = v_i + \sum_{j=0}^{i-1} M_{i,j} w_j. \tag{6}$$

The particular advantage of the parameterization (6) is that the set of constraint admissible policy parameters $\{\{M_{i,j}\}, \{v_i\}\}$ is guaranteed to be convex when the constraint sets $\mathcal{Z}$ and X_f are convex.

The parameterization (6) has been proposed as a means for finding solutions to a general class of robust optimization problems, called affinely adjustable robust counterpart (AARC) problems [6, 30]. The same parameterization has also appeared specifically in application to robust model predictive control problems in [31, 32, 39, 40], and appears to have originally been suggested within the context of stochastic programs with recourse [23]. In the particular control context considered here, the reparameterization of (5) to the convex form (6) may be considered a special case of the well-known Youla parameterization in linear system theory [56] [58, Ch. 12].

Using the parameterization (6), robust optimization modelling techniques [5, 6, 30] are used to eliminate the unknown disturbances w_j and formulate the admissible set of control policies with $O(N^2mn)$ variables, where N is the horizon length as above, and m and n are the respective dimensions of the controls u_i and states x_i at each stage. This implies that, given a suitable objective function, an optimal affine state feedback policy (5) can be found in time that is polynomially bounded in the size of the problem data.

Efficient computation in robust optimal control

In the present paper we demonstrate that an optimal policy of the form (6), equivalently (5), can be efficiently calculated in practice, given suitable polytopic assumptions on the constraint sets W, $\mathcal{Z}$ and X_f. This result is critical for practical applications, since one would generally implement a controller in a receding horizon fashion by calculating, on-line and at each time instant, an admissible control policy (5), given the current state x. Such a control strategy has been shown to allow for the synthesis of stabilizing, nonlinear time-invariant control laws that guarantee satisfaction of the constraints $\mathcal{Z}$ for all time, for all possible disturbance sequences generated from W [28].

While convexity of the robust optimal problem arising out of (6) is key, the resulting optimization problem is a dense convex quadratic program with $O(N^2)$ variables (see Section 2.3, cf. [28]), assuming N dominates the dimension of controls m and states n at each stage. Hence each iteration of an interior-point method will require the solution of a dense linear system and thus require $O(N^6)$ time. This situation is common, for example, in the rapidly growing number of aerospace and automotive applications of predictive control [41, Sec. 3.3] [45]. We show that when the disturbance set is ∞-norm bounded or the linear map of a hypercube, the special structure of the robust optimal control problem can be exploited to devise a sparse formulation of the problem, thereby realizing a substantial reduction in computational effort to $O(N^3)$ work per interior-point iteration.

We demonstrate that the cubic-time performance of interior-point algorithms at each step can be guaranteed when using a factorization technique based on Riccati recursion and block elimination. Numerical results are presented that demonstrate that the technique is computationally feasible for systems of appreciable complexity using the standard sparse linear system solver MA27 [34] within the primal-dual interior-point solver OOQP [24]. We compare this primal-dual interior-point approach to the sparse active-set method PATH [17] on both the dense and sparse problem formulations. Our results suggest that the interior-point method applied to the sparse formulation is the most practical method for solving robust optimal control problems, at least in the “cold start” situation when optimal active set information is unavailable.

A final remark is that the sparse formulation of robust optimal control results from a decomposition technique that can be used to separate the problem into a set of coupled finite horizon control problems. This reduction of effort is the analogue, for robust control, to the situation in classical unconstrained optimal control in which Linear Quadratic Regulator (LQR) problems can be solved in $O(N)$ time, using a Riccati [1, Sec. 2.4] or Differential Dynamic Programming [35] technique in which the state feedback equation $x^+ = Ax + Bu$ is explicit in every stage, compared to $O(N^3)$ time for the more compact formulation in which states are eliminated from the system. More direct motivation for our work comes from [8, 16, 46, 51, 54], which describe efficient implementations of optimization methods for solving optimal control problems with state and control constraints, though without disturbances.

Contents

The paper is organized as follows: Section 2 introduces the optimal control problem considered throughout the paper, and shows how the class of affine disturbance feedback policies described in [28] may be used to design a receding horizon control (RHC) law which can be implemented via the solution of a quadratic program (QP) at each time step. Section 3 gives an equivalent formulation for this QP that can be decomposed into a highly structured, singly-bordered block-diagonal quadratic program through reintroduction of appropriate state variables. Section 4 demonstrates that, when using a primal-dual interior-point solution technique, the decomposed quadratic program can always be solved in an amount of time which is cubic in the horizon length at each interior-point iteration. Section 5 demonstrates through numerical examples that the proposed decomposition can be solved much more efficiently than the equivalent original formulation. The paper concludes in Section 6 with suggestions for further research.

Notation: The set of integers $\{i, \ldots, j\}$ is denoted $\mathbb{Z}_{[i,j]}$. Given vectors x and y and matrices A and B, $A \succ 0$ ($\succeq 0$) means that A is positive (semi)definite, $A \otimes B$ is the Kronecker product of A and B, x' is the transpose of x, $\mathrm{vec}(x, y) := [x'\ y']'$ and $\|x\|_A := \sqrt{x'Ax}$. For scalar q, $\lfloor q \rfloor$ is the largest integer less than or equal to q. The vector $\mathbf{1}$ is an appropriately sized column vector with all entries equal to 1.

2 Problem Statement

Consider the following discrete-time linear time-invariant system:

$$x^+ = Ax + Bu + w, \tag{7}$$

where $x \in \mathbb{R}^n$ is the system state at the current time instant, x^+ is the state at the next time instant, $u \in \mathbb{R}^m$ is the control input and $w \in \mathbb{R}^n$ is the disturbance. It is assumed that (A, B) is stabilizable and that at each sample instant a measurement of the state is available. It is further assumed that the current and future values of the disturbance are unknown and may change unpredictably from one time instant to the next, but are contained in a convex and compact (closed and bounded) set W, which contains the origin.

The system is subject to mixed constraints on the state and input:

$$\mathcal{Z} := \{(x,u) \in \mathbb{R}^n \times \mathbb{R}^m \mid Cx + Du \leq b\}, \tag{8}$$

where the matrices $C \in \mathbb{R}^{s\times n}$, $D \in \mathbb{R}^{s\times m}$ and the vector $b \in \mathbb{R}^s$; s is the number of affine inequality constraints that define $\mathcal{Z}$. A design goal is to guarantee that the state and input of the closed-loop system remain in $\mathcal{Z}$ for all time and for all allowable disturbance sequences.

In addition to $\mathcal{Z}$, a target/terminal constraint set X_f is given by

$$X_f := \{x \in \mathbb{R}^n \mid Yx \leq z\}, \tag{9}$$

where the matrix $Y \in \mathbb{R}^{r\times n}$ and the vector $z \in \mathbb{R}^r$; r is the number of affine inequality constraints that define X_f. The set X_f can, for example, be used as a target set in time-optimal control or, if defined to be robust positively invariant, to design a receding horizon controller with guaranteed invariance and stability properties [28].

Before proceeding, we define some additional notation. In the sequel, predictions of the system's evolution over a finite control/planning horizon will be used to define a number of suitable control policies. Let the length N of this planning horizon be a positive integer and define stacked versions of the predicted input, state and disturbance vectors $\mathbf{u} \in \mathbb{R}^{mN}$, $\mathbf{x} \in \mathbb{R}^{n(N+1)}$ and $\mathbf{w} \in \mathbb{R}^{nN}$, respectively, as

$$\begin{aligned}\mathbf{x} &:= \operatorname{vec}(x_0,\dots,x_{N-1},x_N),\\ \mathbf{u} &:= \operatorname{vec}(u_0,\dots,u_{N-1}),\\ \mathbf{w} &:= \operatorname{vec}(w_0,\dots,w_{N-1}),\end{aligned}$$

where $x_0 = x$ denotes the current measured value of the state and $x_{i+1} := Ax_i + Bu_i + w_i$, $i = 0,\dots,N-1$ denotes the prediction of the state after i time instants into the future. Finally, let the set $\mathcal{W} := W^N := W \times \cdots \times W$, so that $\mathbf{w} \in \mathcal{W}$.

2.1 Affine Disturbance Feedback Policies

As noted in the Introduction, the problem of constructing a constraint admissible finite horizon feedback policy π for the system (2), where the control input at each time is specified as an *arbitrary* function of prior states, is extremely difficult in general. We therefore choose to employ a more restricted class of *affine* feedback policies, where the control input at each time is modelled as an affine function of the sequence of past disturbances, so that

$$u_i = v_i + \sum_{j=0}^{i-1} M_{i,j} w_j, \quad \forall i \in \mathbb{Z}_{[0,N-1]}, \tag{11}$$

where each $M_{i,j} \in \mathbb{R}^{m\times n}$ and $v_i \in \mathbb{R}^m$ are decision variables to be specified in the construction of the control policy. We note that, since full state feedback is assumed, the past disturbance sequence is easily calculated as the difference between the predicted and actual states at each step, i.e.

$$w_i = x_{i+1} - Ax_i - Bu_i, \quad \forall i \in \mathbb{Z}_{[0,N-1]}. \tag{12}$$

Define the variable $\mathbf{v} \in \mathbb{R}^{mN}$ and the block lower triangular matrix $\mathbf{M} \in \mathbb{R}^{mN\times nN}$ such that

$$\mathbf{M} := \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ M_{1,0} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ M_{N-1,0} & \cdots & M_{N-1,N-2} & 0 \end{bmatrix}, \ \mathbf{v} := \begin{bmatrix} v_0 \\ \vdots \\ v_{N-1} \end{bmatrix}, \tag{13}$$

so that the control input sequence can be written as $\mathbf{u} = \mathbf{M}\mathbf{w} + \mathbf{v}$. Define the set of admissible policies $(\mathbf{M},\mathbf{v})$, for which the constraints (8) and (9) are satisfied, as:

$$\Pi_N^{df}(x) := \left\{ (\mathbf{M},\mathbf{v}) \left| \begin{array}{r} (\mathbf{M},\mathbf{v}) \text{ satisfies (13)}, x = x_0 \\ x_{i+1} = Ax_i + Bu_i + w_i \\ u_i = v_i + \sum_{j=0}^{i-1} M_{i,j} w_j \\ Cx_i + Du_i \leq b, \ Yx_N \leq z \\ \forall w_i \in W, \ \forall i \in \mathbb{Z}_{[0,N-1]} \end{array} \right. \right\}, \tag{14}$$

and define the set of initial states x for which an admissible control policy of the form (11) exists as

$$X_N^{df} := \{x \in \mathbb{R}^n \mid \Pi_N^{df}(x) \neq \emptyset\}. \tag{15}$$

Note that, as shown in [26, 28], the sets $\Pi_N^{df}(x)$ and X_N^{df} are convex since the sets $\mathcal{Z}$ and X_f are convex.

2.2 A Receding Horizon Control Law

We are chiefly interested in employing the policy parameterization (11) to construct a *receding horizon control* (RHC) law for the uncertain linear system (2), i.e. at each sample instant, given a measurement of the current state x, we would like to calculate *on-line* a constraint admissible policy $(\mathbf{M}(x), \mathbf{v}(x)) \in \Pi_N^{df}(x)$ that is optimal with respect to some cost function, and apply the first component of this policy to the system (2). We stress that, for problems of non-trivial size, this determination of control policies *must* be performed on-line, since it is generally not possible to select a single pair $(\mathbf{M},\mathbf{v})$ such that $(\mathbf{M},\mathbf{v}) \in \Pi_N^{df}(x)$ for all $x \in X_N^{df}$. The main contribution of this paper is to describe an efficient computational method by which this on-line calculation may be performed.

In particular, we define an optimal policy pair $(\mathbf{M}^*(x), \mathbf{v}^*(x)) \in \Pi_N^{df}(x)$ to be one that minimizes the value of a cost function that is quadratic in the disturbance-free state and input sequence. We thus define:

$$V_N(x,\mathbf{v}) := \frac{1}{2}\|\hat{x}_N\|_P^2 + \sum_{i=0}^{N-1}\left(\frac{1}{2}\|\hat{x}_i\|_Q^2 + \frac{1}{2}\|v_i\|_R^2\right) \tag{16}$$

where $\hat{x}_0 = x$, $\hat{x}_{i+1} = A\hat{x}_i + Bv_i$ for $i = 0,\ldots,N-1$; the matrices Q and P are assumed positive semidefinite, and R is assumed positive definite. The cost function (16) can alternatively be written in vectorized form as

$$V_N(x,\mathbf{v}) = \frac{1}{2}\|\mathbf{A}x + \mathbf{B}\mathbf{v}\|_{\mathcal{Q}}^2 + \frac{1}{2}\|\mathbf{v}\|_{\mathcal{R}}^2, \tag{17}$$

where $\mathbf{A} \in \mathbb{R}^{n(N+1)\times n}$ and $\mathbf{B} \in \mathbb{R}^{n(N+1)\times mN}$ are defined in Appendix A and where $\mathcal{Q} := \left[\begin{smallmatrix} I\otimes Q & \\ & P \end{smallmatrix}\right]$ and $\mathcal{R} := I \otimes R$. We define an optimal policy pair as

$$(\mathbf{M}^*(x), \mathbf{v}^*(x)) := \operatorname*{argmin}_{(\mathbf{M},\mathbf{v})\in \Pi_N^{df}(x)} V_N(x,\mathbf{v}). \tag{18}$$

For the receding-horizon control case, a *time-invariant* control law $\mu_N : X_N^{df} \to \mathbb{R}^m$ can be implemented by using the first part of this optimal control policy at each time instant, i.e.

$$\mu_N(x) := v_0^*(x), \tag{19}$$

where $\mathbf{v}^*(x) =: (v_0^*(x), \dots, v_{N-1}^*(x))$. We emphasize that, due to the dependence of the optimization problem (18) on the current state x, the control law $\mu_N(\cdot)$ will, in general, be a *nonlinear* function with respect to the current state, even though it may have been defined in terms of the class of affine feedback policies (11).

The control law $\mu_N(\cdot)$ has many attractive geometric and system-theoretic properties. In particular, implementation of the receding horizon control law $\mu_N(\cdot)$ renders the set X_N^{df} robust positively invariant, i.e. if $x \in X_N^{df}$, then it can be shown that $Ax + B\mu_N(x) + w \in X_N^{df}$ for all $w \in W$, subject to certain technical conditions on the terminal set X_f. Furthermore, the control law $\mu_N(\cdot)$ is uniquely defined for each x, and the closed-loop system is guaranteed to be input-to-state (ISS) stable under suitable assumptions on Q, P, R and X_f. Finally, calculation of an optimal policy in (18) requires the minimization of a convex function over a convex set, so that $\mu_N(\cdot)$ in (19) is thus practically realizable for a variety of disturbance classes. The reader is referred to [26, 28] for a proof of these results and a review of other system-theoretic properties of this parameterization.

2.3 Solution via Quadratic Programming

As shown in [28], it is possible to eliminate the universal quantifier in (14) and construct matrices $F \in \mathbb{R}^{(sN+r)\times mN}$, $G \in \mathbb{R}^{(sN+r)\times nN}$ and $T \in \mathbb{R}^{(sN+r)\times n}$, and vector $c \in \mathbb{R}^{sN+r}$ (defined in Appendix A) such that the set of feasible pairs $(\mathbf{M},\mathbf{v})$ can be written as:

$$\Pi_N^{df}(x) = \left\{ (\mathbf{M},\mathbf{v}) \;\middle|\; \begin{array}{r} (\mathbf{M},\mathbf{v}) \text{ satisfies (13)} \\ F\mathbf{v} + \max\limits_{\mathbf{w}\in\mathcal{W}} (F\mathbf{M}+G)\mathbf{w} \le c + Tx \end{array} \right\}, \tag{20}$$

where $\max_{\mathbf{w}\in\mathcal{W}}(F\mathbf{M}+G)\mathbf{w}$ denotes row-wise maximization – note that this is equivalent to evaluating the *support function* of the set $\mathcal{W}$ for each column of the matrix $(F\mathbf{M}+G)'$, and that these maxima always exist since the set $\mathcal{W}$ is assumed to be compact. In the remainder of this paper, we consider the particular case where W is generated as the linear map of a hypercube. Define

$$W = \{w \in \mathbb{R}^n \mid w = Ed,\ \|d\|_\infty \le 1\}, \tag{21}$$

where $E \in \mathbb{R}^{n\times l}$ is assumed to have full column rank, so that the stacked generating disturbance sequence $\mathbf{d} \in \mathbb{R}^{lN}$ is

$$\mathbf{d} := \operatorname{vec}(d_0, \dots, d_{N-1}), \tag{22}$$

and define the matrix $J := I_N \otimes E \in \mathbb{R}^{Nn\times Nl}$, so that $\mathbf{w} = J\mathbf{d}$. From the properties of the dual norm [33], when the generating disturbance d is an ∞-norm bounded signal given as in (21), then

$$\max_{w\in W} a'w = \|E'a\|_1 \tag{23}$$

for any vector $a \in R^n$. Straightforward application of (23) to the row-wise maximization in (20) yields

$$\Pi_N^{df}(x) = \left\{ (\mathbf{M},\mathbf{v}) \;\middle|\; \begin{array}{r} (\mathbf{M},\mathbf{v}) \text{ satisfies (13)} \\ F\mathbf{v} + \mathrm{abs}(F\mathbf{M}J + GJ)\mathbf{1} \le c + Tx \end{array} \right\}, \tag{24}$$

where $\mathrm{abs}(F\mathbf{M}J+GJ)\mathbf{1}$ is a vector formed from the 1-norms of the rows of the matrix $(F\mathbf{M}J+GJ)$. This can be written as a set of purely affine constraints by introducing slack variables and rewriting as

$$\Pi_N^{df}(x) = \left\{ (\mathbf{M},\mathbf{v}) \;\middle|\; \begin{array}{r} (\mathbf{M},\mathbf{v}) \text{ satisfies (13), } \exists \mathbf{\Lambda} \text{ s.t.} \\ F\mathbf{v} + \mathbf{\Lambda}\mathbf{1} \le c + Tx \\ -\mathbf{\Lambda} \le (F\mathbf{M}J + GJ) \le \mathbf{\Lambda} \end{array} \right\}. \tag{25}$$

The control policy optimization problem (18) can thus be solved in this case by forming a quadratic program in the variables $\mathbf{M}$, $\mathbf{\Lambda}$, and $\mathbf{v}$, i.e.

$$\min_{\mathbf{M},\mathbf{\Lambda},\mathbf{v}} \frac{1}{2}\|\mathbf{A}x + \mathbf{B}\mathbf{v}\|_{\mathcal{Q}}^2 + \frac{1}{2}\|\mathbf{v}\|_{\mathcal{R}}^2 \tag{26a}$$

subject to:

$$M_{i,j} = 0, \quad \forall i \le j \tag{26b}$$

$$F\mathbf{v} + \mathbf{\Lambda}\mathbf{1} \le c + Tx \tag{26c}$$

$$-\mathbf{\Lambda} \le (F\mathbf{M}J + GJ) \le \mathbf{\Lambda}. \tag{26d}$$

Remark 1 The total number of decision variables in (26) is $mnN(N-1)/2$ in $\mathbf{M}$, mN in $\mathbf{v}$ and $(slN^2 + rlN)$ in $\mathbf{\Lambda}$, with the number of constraints equal to $(sN+r)$ + $2(slN^2 + rlN))$, or $\mathcal{O}(N^2)$ overall. For a naive interior-point computational approach using a dense factorization method, the resulting quadratic program would thus require computation time of $\mathcal{O}(N^6)$ at each iteration.

2.3.1 Writing $\Pi_N^{df}(x)$ in Separable Form

We next define the variable transformation $\mathbf{U} := \mathbf{M}J$, such that $\mathbf{U} \in \mathbb{R}^{mN\times lN}$ has block lower triangular structure similar to that defined in (13) for $\mathbf{M}$. Note that use of this variable transformation is tantamount to parameterizing the control policy directly in terms of the generating disturbances d_i, so that $u_i = v_i + \sum_{j=0}^{i-1} U_{i,j}d_j$, or $\mathbf{u} = \mathbf{U}\mathbf{d} + \mathbf{v}$.

When the matrix E is full column rank, the QP (26) may be solved using this variable transformation by solving an equivalent QP in the variables $\mathbf{U}$, $\mathbf{\Lambda}$ and $\mathbf{v}$:

$$\min_{\mathbf{U},\mathbf{\Lambda},\mathbf{v}} \frac{1}{2}\|\mathbf{A}x + \mathbf{B}\mathbf{v}\|_{\mathcal{Q}}^2 + \frac{1}{2}\|\mathbf{v}\|_{\mathcal{R}}^2 \tag{27a}$$

subject to:

$$U_{i,j} = 0, \quad \forall i \leq j \tag{27b}$$

$$F\mathbf{v} + \mathbf{\Lambda}\mathbf{1} \leq c + Tx \tag{27c}$$

$$-\mathbf{\Lambda} \leq (F\mathbf{U} + GJ) \leq \mathbf{\Lambda}. \tag{27d}$$

The equivalence between the QPs (26) and (27) when E (and thus J) has full column rank is easily demonstrated by employing a left inverse $J^\dagger$ such that $J^\dagger J = I$, since any feasible solution $(\mathbf{M}, \mathbf{\Lambda}, \mathbf{v})$ satisfying the constraint in (27d) also satisfies the constraint (26d) with $\mathbf{M} = \mathbf{U}J^\dagger$.

Remark 2 The critical feature of the quadratic program (27) is that the columns of the variables $\mathbf{U}$ and $\mathbf{\Lambda}$ are decoupled in the constraint (27d). This allows columnwise separation of the constraint into a number of subproblems, subject to the coupling constraint (27c). The reader is referred to [28] for details on the solution of the optimization problem (18) when W is an arbitrary polytope or 2–norm bounded.

2.4 Soft Constraints and Guaranteed Feasibility

An important practical consideration for control applications is the handling of potential infeasibility of the optimization problem (18). If the RHC law $\mu_N(\cdot)$ is to be implemented on-line for a real system, it is important to guarantee reasonable controller behavior if the plant enters a state x such that $\mathit{\Pi}_N^{df}(x)$ is empty (equivalently, $x \notin X_N^{df}$). A common approach in the literature in receding horizon control is to treat some or all of the constraints in $\mathcal{Z}$ or X_f as so-called *soft constraints*, i.e. constraints that may be violated if necessary to guarantee that the optimization problem (27), and particularly the constraint (27c), is feasible for all x. Techniques for soft constraint handling are well established in the literature on linear predictive control for undisturbed systems [41, 46, 48], and we show briefly how these ideas may be extended to cover the robust control problem considered here. Without loss of generality, we consider the simplest case where every constraint is a soft constraint, and replace the hard state and input constraints in (14) with soft constraints of the form

$$Cx_i + Du_i \leq b + \xi_i, \quad \xi_i \geq 0, \quad \forall i \in \mathbb{Z}_{[0,N-1]} \tag{28a}$$

$$Yx_N \leq z + \xi_N,\ \xi_N \geq 0, \tag{28b}$$

and augment the objective function with linear-quadratic terms $(\gamma_i'\xi_i + \xi_i'\mathit{\Gamma}_i\xi_i)$ penalizing the soft constraint violations ξ_i, where $\mathit{\Gamma}_i \succeq 0$. The optimization problem (27) becomes

$$\min_{\mathbf{U},\mathbf{\Lambda},\mathbf{v},\boldsymbol{\xi}} \ \frac{1}{2}\|\mathbf{A}x + \mathbf{B}\mathbf{v}\|_{\mathcal{Q}}^2 + \frac{1}{2}\|\mathbf{v}\|_{\mathcal{R}}^2 + \sum_{i=0}^{N}(\gamma_i'\xi_i + \xi_i'\mathit{\Gamma}_i\xi_i) \tag{29a}$$

subject to:

$$U_{i,j} = 0,\ \forall i \leq j \tag{29b}$$
$$F\mathbf{v} + \mathbf{\Lambda 1} \leq c + Tx + \boldsymbol{\xi},\ \ \boldsymbol{\xi} \geq 0, \tag{29c}$$
$$-\mathbf{\Lambda} \leq (F\mathbf{U} + GJ) \leq \mathbf{\Lambda}, \tag{29d}$$

where $\boldsymbol{\xi} := \mathrm{vec}(\xi_0, \ldots, \xi_N)$. Note that the quadratic program (29) is feasible for all x, so that a receding horizon controller synthesized via repeated solution of this QP is defined everywhere on $\mathbb{R}^n$. A well-known feature of such penalty function formulations is that if, in the spirit of [22, Sec. 12.3] [41, Sec. 3.4], if one defines an exact penalty function (by choosing γ_i large enough), then solutions to (29) correspond exactly to solution of (27) for all $x \in X_N^{df}$.

3 Recovering Structure in the Robust Control Problem

The quadratic program (QP) defined in (27) can be rewritten in a more computationally attractive form by re-introducing the eliminated state variables to achieve greater structure. The re-modelling process separates the original problem into subproblems; a nominal problem, consisting of that part of the state resulting from the nominal control vector **v**, and a set of perturbation problems, each representing the components of the state resulting from each of the columns of (27d) in turn.

Nominal States and Inputs

We first define a constraint contraction vector $\delta\mathbf{c} \in \mathbb{R}^{sN+r}$ such that

$$\delta\mathbf{c} := \mathrm{vec}(\delta c_0, \ldots, \delta c_N) = \mathbf{\Lambda 1}, \tag{30}$$

so that the constraint (27c) becomes

$$F\mathbf{v} + \delta\mathbf{c} \leq c + Tx. \tag{31}$$

Recalling that the nominal states $\hat{x}_i$ are defined in (16) as the expected states given no disturbances, it is easy to show that the constraint (31) can be written explicitly in terms of the nominal controls v_i and states $\hat{x}_i$ as

$$\hat{x}_{i+1} - A\hat{x}_i - Bv_i = 0, \quad \forall i \in \mathbb{Z}_{[0,N-1]} \tag{32a}$$
$$C\hat{x}_i + Dv_i + \delta c_i \leq b, \quad \forall i \in \mathbb{Z}_{[0,N-1]} \tag{32b}$$
$$Y\hat{x}_N + \delta c_N \leq z, \tag{32c}$$

where $\hat{x}_0 = x$, which is in a form that is *exactly the same* as that in conventional receding horizon control problem with no disturbances, but with the right-hand sides of the state and input constraints at each stage i modified by the constraint contraction terms δc_i; compare (32a)–(32c) and (7)–(9) respectively.

Perturbed States and Inputs

We next consider the effects of each of the columns of $(F\mathbf{U}+GJ)$ in turn, and seek to construct a set of problems similar to that in (32). We treat each column as the output of a system subject to a unit impulse in a single element of $\mathbf{d}$, and construct a subproblem that calculates the effect of that disturbance on the nominal problem constraints (32b)–(32c) by determining its contribution to the total constraint contraction vector $\delta\mathbf{c}$.

From the original QP constraint (27d), the constraint contraction vector $\delta\mathbf{c}$ can be written as

$$\text{abs}(F\mathbf{U}+GJ)\mathbf{1} \leq \mathbf{\Lambda 1} = \delta\mathbf{c}. \tag{33}$$

The left-hand side of (33) is just a summation over the columns of the matrix $\text{abs}(F\mathbf{U}+GJ)$, so that

$$\text{abs}(F\mathbf{U}+GJ)\mathbf{1} = \sum_{p=1}^{lN} \text{abs}((F\mathbf{U}+GJ)e_p). \tag{34}$$

where $e_p \in \mathbb{R}^{lN}$ is a vector whose p^{th} element is equal to 1, with all other elements equal to zero. Define $\mathbf{y}^p \in \mathbb{R}^{sN+r}$ and $\delta\mathbf{c}^p \in \mathbb{R}^{sN+r}$ as

$$\mathbf{y}^p := (F\mathbf{U}+GJ)e_p \tag{35}$$

$$\delta\mathbf{c}^p := \text{abs}(\mathbf{y}^p). \tag{36}$$

Note that the unit vector e_p models a unit disturbance in some element j of the generating disturbance d_k at some time step k, with no disturbances at any other step[1]. If we denote the j^{th} column of E as $E_{(j)}$, then it is easy to recognize $\mathbf{y}^p$ as the stacked output vector of the system

$$(u_i^p, x_i^p, y_i^p) = 0, \quad \forall i \in \mathbb{Z}_{[0,k]} \tag{37a}$$

$$x_{k+1}^p = E_{(j)}, \tag{37b}$$

$$x_{i+1}^p - Ax_i^p - Bu_i^p = 0, \quad \forall i \in \mathbb{Z}_{[k+1,N-1]} \tag{37c}$$

$$y_i^p - Cx_i^p - Du_i^p = 0, \quad \forall i \in \mathbb{Z}_{[k+1,N-1]} \tag{37d}$$

$$y_N^p - Yx_N^p = 0, \tag{37e}$$

where $\mathbf{y}^p = \text{vec}(y_0^p, \ldots, y_N^p)$. The inputs u_i^p of this system come directly from the p^{th} column of the matrix $\mathbf{U}$, i.e. they are the columns of the sub-matrices $U_{i,k}$. If the constraint terms $\delta\mathbf{c}^p$ for each subproblem are similarly defined as $\delta\mathbf{c}^p := \text{vec}(\delta c_0^p, \ldots, \delta c_N^p)$, then each component must satisfy $\delta c_i^p = \text{abs}(y_i^p)$, or in linear inequality constraint form

$$-\delta c_i^p \leq y_i^p \leq \delta c_i^p. \tag{38}$$

Note also that for the p^{th} subproblem, representing a disturbance at stage $k = \lfloor \frac{p-1}{l} \rfloor$, the constraint contraction terms are zero prior to stage $(k+1)$.

[1] Note that this implies $p = lk + j$, $k = \lfloor \frac{p-1}{l} \rfloor$ and $j = 1 + ((p-1) \mod l)$.

By further defining

$$\bar{C} := \begin{bmatrix} +C \\ -C \end{bmatrix} \; \bar{D} := \begin{bmatrix} +D \\ -D \end{bmatrix} \; \bar{Y} := \begin{bmatrix} +Y \\ -Y \end{bmatrix} \; H := \begin{bmatrix} -I_s \\ -I_s \end{bmatrix} \; H_f := \begin{bmatrix} -I_r \\ -I_r \end{bmatrix}, \tag{39}$$

equations (37d) and (37e) can be combined with (38) to give

$$\bar{C}x_i^p + \bar{D}u_i^p + H\delta c_i^p \leq 0, \quad \forall i \in \mathbb{Z}_{[k+1,N-1]} \tag{40a}$$

$$\bar{Y}x_N^p + H_f \delta c_N^p \leq 0. \tag{40b}$$

3.1 Complete Robust Control Problem

We can now restate the complete robust optimization problem (27) as:

$$\min_{\substack{\hat{x}_1,\ldots,\hat{x}_N, v_0,\ldots v_{N-1}, \delta c_0,\ldots,\delta c_N, \\ x_0^1,\ldots,x_N^1, u_0^1,\ldots u_{N-1}^1, \delta c_0^1,\ldots,\delta c_N^1, \\ \ldots, \\ x_0^{lN},\ldots,x_N^{lN}, u_0^{lN},\ldots u_{N-1}^{lN}, \delta c_0^{lN},\ldots,\delta c_N^{lN}}} \frac{1}{2}\|\hat{x}_N\|_P^2 + \sum_{i=0}^{N-1}\left(\frac{1}{2}\|\hat{x}_i\|_Q^2 + \frac{1}{2}\|v_i\|_R^2\right) \tag{41}$$

subject to (32), (37a)–(37c) and (40), which we restate here for convenience:

$$\hat{x}_{i+1} - A\hat{x}_i - Bv_i = 0, \quad \forall i \in \mathbb{Z}_{[0,N-1]} \tag{42a}$$

$$C\hat{x}_i + Dv_i + \delta c_i \leq b, \quad \forall i \in \mathbb{Z}_{[0,N-1]} \tag{42b}$$

$$Y\hat{x}_N + \delta c_N \leq z, \tag{42c}$$

where $\hat{x}_0 = x$, and

$$\delta c_i = \sum_{p=1}^{lN} \delta c_i^p, \quad \forall i \in \mathbb{Z}_{[0,N]}, \tag{43}$$

and, for each $p \in \mathbb{Z}_{[1,lN]}$:

$$(u_i^p, x_i^p, \delta c_i^p) = 0, \quad \forall i \in \mathbb{Z}_{[0,k]} \tag{44a}$$

$$x_{k+1}^p = E_{(j)}, \tag{44b}$$

$$x_{i+1}^p - Ax_i^p - Bu_i^p = 0, \quad \forall i \in \mathbb{Z}_{[k+1,N-1]} \tag{44c}$$

$$\bar{C}x_i^p + \bar{D}u_i^p + H\delta c_i^p \leq 0, \quad \forall i \in \mathbb{Z}_{[k+1,N-1]} \tag{44d}$$

$$\bar{Y}x_N^p + H_f \delta c_N^p \leq 0, \tag{44e}$$

where $k = \lfloor \frac{p-1}{l} \rfloor$ and $j = 1 + ((p-1) \mod l)$. The decision variables in this problem are the nominal states and controls $\hat{x}_i$ and v_i at each stage (the initial state $\hat{x}_0$ is known, hence *not* a decision variable), plus the perturbed states, controls, and constraint contraction terms x_i^p, u_i^p, and δc_i^p for each subproblem at each stage.

Remark 3 Recalling the discussion of Section 2.4, soft constraints are easily incorporated into the optimization problem (41)–(44) via modification of the cost function (41) and of the constraints (42b)–(42c). The important point regarding this soft constraint inclusion is that it *does not* result in a modification of any of the perturbation constraints (44), so that the qualitative results to be presented in Section 4 relating to efficient solution of the QP (41)–(44) are not fundamentally altered by the incorporation of soft constraints.

We can now state the following key result, proof of which follows directly from the discussion of Section 2.3.1 and of this section.

Theorem 1 *The convex, tractable QP* (41)–(44) *is equivalent to the robust optimal control problems* (26) *and* (27)*. The receding horizon control law $u = \mu_N(x)$ in* (19) *can be implemented using the solution to* (41)–(44) *as $u = v_0^*(x)$.*

The importance of the re-introduction of states in (42) and (44) is that significant structure and sparsity can be revealed in the problem through an interleaving of decision variables by time index. For the nominal problem, define the stacked vector of variables:

$$\mathsf{x}_0 := \operatorname{vec}(v_0, \hat{x}_1, v_1, \dots, \hat{x}_{N-1}, v_{N-1}, \hat{x}_N). \tag{45}$$

For the p^{th} perturbation problem in (44), which models a unit disturbance at stage $k = \lfloor \frac{p-1}{l} \rfloor$, define:

$$\begin{aligned}\mathsf{x}_p := \operatorname{vec}(u^p_{k+1}, \delta c^p_{k+1}, x^p_{k+2}, u^p_{k+2}, \delta c^p_{k+2}, \dots,\\ x^p_{N-1}, u^p_{N-1}, \delta c^p_{N-1}, x^p_N, \delta c^p_N).\end{aligned} \tag{46}$$

Using this reordering, the constraints (42)–(44) can be written as a single set of linear constraints in singly-bordered block-diagonal form with considerable structure and sparsity:

$$\begin{bmatrix} \mathsf{A}_0 & & & \\ & \mathsf{A}_1 & & \\ & & \ddots & \\ & & & \mathsf{A}_{lN} \end{bmatrix} \begin{bmatrix} \mathsf{x}_0 \\ \mathsf{x}_1 \\ \vdots \\ \mathsf{x}_{lN} \end{bmatrix} = \begin{bmatrix} \mathsf{b}_0 \\ \mathsf{b}_1 \\ \vdots \\ \mathsf{b}_{lN} \end{bmatrix}, \quad \begin{bmatrix} \mathsf{C}_0 & \mathsf{J}_1 & \cdots & \mathsf{J}_{lN} \\ & \mathsf{C}_1 & & \\ & & \ddots & \\ & & & \mathsf{C}_{lN} \end{bmatrix} \begin{bmatrix} \mathsf{x}_0 \\ \mathsf{x}_1 \\ \vdots \\ \mathsf{x}_{lN} \end{bmatrix} \leq \begin{bmatrix} \mathsf{d}_0 \\ \mathsf{d}_1 \\ \vdots \\ \mathsf{d}_{lN} \end{bmatrix}. \tag{47}$$

The coefficient matrices A_0 and C_0 in (47) originate from the nominal problem constraints (42), and are defined as

$$\mathsf{A}_0 := \begin{bmatrix} B & -I & & & & \\ & A & B & -I & & \\ & & & \ddots & & \\ & & & A & B & -I \end{bmatrix}, \quad \mathsf{C}_0 := \begin{bmatrix} D & & & & & \\ & C & D & & & \\ & & & \ddots & & \\ & & & & C & D \\ & & & & & Y \end{bmatrix}, \tag{48}$$

with corresponding right hand sides

$$\mathsf{b}_0 := \operatorname{vec}(-Ax, 0, 0, \dots, 0), \quad \mathsf{d}_0 := \operatorname{vec}(b - Cx, b, \dots, b, z). \tag{49}$$

The coefficient matrices A_p and C_p in (47) originate from the constraints for the p^{th} perturbation subproblem in (44), and are defined as

$$\mathsf{A}_p := \begin{bmatrix} B & 0 & -I & & & & & & \\ & & A & B & 0 & -I & & & \\ & & & & & \ddots & & & \\ & & & & & A & B & 0 & -I & 0 \end{bmatrix}, \quad \mathsf{C}_p := \begin{bmatrix} \bar{D} & H & & & & & & \\ & & \bar{C} & \bar{D} & H & & & \\ & & & & & \ddots & & \\ & & & & & \bar{C} & \bar{D} & H & \\ & & & & & & & \bar{Y} & H_f \end{bmatrix}, \tag{50}$$

with corresponding right hand sides

$$\mathsf{b}_p := \operatorname{vec}(-AE_{(j)}, 0, \ldots, 0), \quad \mathsf{d}_p := \operatorname{vec}(0, 0, \ldots, 0, 0). \tag{51}$$

The coupling matrices J_p in (47) are then easily constructed from the coupling equation (43).

Remark 4 It is possible to define a problem structure similar to that in (41)–(44) for the more general polytopic disturbance sets discussed in [28] via introduction of states in a similar manner. However, in this case the perturbation subproblems (44) require an additional coupling constraint for the subproblems associated with each stage.

4 Interior-Point Method for Robust Control

In this section we demonstrate that, using a primal-dual interior-point solution technique, the quadratic program defined in (41)–(44) can be solved with a per-iteration computational effort that grows cubicly with the horizon length N, when $n+m$ is dominated by N; this situation is common, for example, in the rapidly growing number of aerospace and automotive applications of predictive control [41, Sec. 3.3] [45]. This is a major improvement on the $O(N^6)$ work per iteration associated with the compact (dense) formulation (26), or the equivalent problem (27); cf. Remark 1. The improvement in computational efficiency comes about due to the improved structure and sparsity of the problem. Indeed, akin to the situation in [46], we will show that each subproblem in the QP (41)–(44) has the same structure as that of an unconstrained optimal control problem without disturbances.

We first outline some of the general properties of interior-point solution methods.

4.1 General Interior-Point Methods

We consider the general constrained quadratic optimization problem

$$\min_{\theta} \frac{1}{2}\theta'\mathsf{Q}\theta \quad \text{subject to } \mathsf{A}\theta = \mathsf{b},\ \mathsf{C}\theta \leq \mathsf{d}, \tag{52}$$

where the matrix Q is positive semidefinite. A solution θ to this system exists if and only if the Karush-Kuhn-Tucker conditions are satisfied, i.e. there exist additional vectors π, λ and z satisfying the following conditions:

$$\mathsf{Q}\theta + \mathsf{A}'\pi + \mathsf{C}'\lambda = 0 \tag{53a}$$

$$\mathsf{A}\theta - \mathsf{b} = 0 \tag{53b}$$

$$-\mathsf{C}\theta + \mathsf{d} - z = 0 \tag{53c}$$

$$(\lambda, z) \geq 0 \tag{53d}$$

$$\lambda' z = 0 \tag{53e}$$

In primal-dual interior point methods [55], the *central path* is defined as the set of parameters $(\theta, \pi, \lambda, z)$ satisfying (53a)–(53d), with the complementarity condition (53e) relaxed, for each element i, to $\lambda_i z_i = \mu$, where $\mu > 0$ parameterizes the path. This guarantees that λ and z are strictly positive vectors. The central path converges to a solution of (53) as $\mu \downarrow 0$ if such a solution exists.

The constraints $\lambda_i z_i = \mu$ can be rewritten in a slightly more convenient form by defining diagonal matrices Λ and Z such that

$$\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}, \quad Z = \begin{bmatrix} z_1 & & \\ & \ddots & \\ & & z_n \end{bmatrix}, \tag{54}$$

so that the relaxed complementarity condition becomes $\Lambda Z \mathbf{1} = \mu \mathbf{1}$. Primal-dual interior-point algorithms search for a solution to the KKT conditions (53) by producing a sequence of iterates $(\theta^\kappa, \pi^\kappa, \lambda^\kappa, z^\kappa)$ which approximate the central path solution at some $\mu^\kappa > 0$. These iterates are updated via repeated solution of a set of Newton-like equations of the form

$$\begin{bmatrix} \mathsf{Q} & \mathsf{A}' & \mathsf{C}' & \\ \mathsf{A} & & & \\ \mathsf{C} & & & I \\ & & Z & \Lambda \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\pi \\ \Delta\lambda \\ \Delta z \end{bmatrix} = - \begin{bmatrix} r_Q \\ r_A \\ r_C \\ r_Z \end{bmatrix}, \tag{55}$$

where the residuals (r_Q, r_A, r_C) take the values of the left-hand sides of (53a)–(53c) respectively, evaluated at the current values $(\theta^\kappa, \pi^\kappa, \lambda^\kappa, z^\kappa)$, and the matrices (Z, Λ) are formed from the current iterates $(z^\kappa, \lambda^\kappa)$ as in (54). The vector r_Z is typically defined as $r_Z = (\Lambda Z \mathbf{1} - \mathbf{1}\bar{\mu})$, where $\bar{\mu}$ is chosen such that $\bar{\mu} \in (0, \mu^\kappa)$. Once the linear system (55) has been solved, the solution is updated as $(\theta^{\kappa+1}, \pi^{\kappa+1}, \lambda^{\kappa+1}, z^{\kappa+1}) \leftarrow (\theta^\kappa, \pi^\kappa, \lambda^\kappa, z^\kappa) + \alpha(\Delta\theta, \Delta\pi, \Delta\lambda, \Delta z)$, where $\alpha > 0$ is chosen to maintain strict positivity of λ^{k+1} and z^{k+1}, and the path parameter μ^κ is updated to some $\mu^{\kappa+1} \in (0, \mu^\kappa)$. The particular method for selecting the parameters $\bar{\mu}$ and α at each iteration depends on the specific interior-point algorithm employed; the reader is referred to [55] for a thorough review. Since all such methods maintain the strict inequalities $(\lambda, z) > 0$ at each iteration as $\mu \downarrow 0$, the matrices Λ and Z are guaranteed to remain full rank, and the system of equations in (55) can be simplified through elimination of the variables Δz to form the reduced system

$$\begin{bmatrix} \mathsf{Q} & \mathsf{A}' & \mathsf{C}' \\ \mathsf{A} & & \\ \mathsf{C} & & -\Lambda^{-1}\mathsf{Z} \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\pi \\ \Delta\lambda \end{bmatrix} = - \begin{bmatrix} r_Q \\ r_A \\ \left(r_C - \Lambda^{-1} r_Z\right) \end{bmatrix}. \tag{56}$$

Since the number of interior-point iterations required in practice is only weakly related to the number of variables [55], the principal consideration is the time required to factor the Jacobian matrix (i.e., the matrix on the left-hand-side), and solve the linear system in (56). In the remainder of the paper we focus on the development of an efficient solution procedure for this linear system when the problem data for the QP (52) is defined by the robust control problem (41)–(44).

4.2 Robust Control Formulation

For the robust optimal control problem described in (41)–(44), the system of equations in (56) can be arranged to yield a highly structured set of linear equations through appropriate ordering of the primal and dual variables and their Lagrange multipliers at each stage. As will be shown, this ordering enables the development of an efficient solution procedure for the linear system in (56).

We use λ_i and λ_N to denote the Lagrange multipliers for the constraints (42b) and (42c) in the nominal system, and z_i and z_N for the corresponding slack variables. We similarly use λ_i^p and λ_N^p to denote the multipliers in (44d) and (44e) for the p^{th} perturbation subproblem, with slack variables z_i^p and z_N^p. We use π_i and π_i^p to denote the dual variables for (42) and (44).

The linear system (56), defined for the particular robust control problem (41)–(44), can then be reordered to form a symmetric, block-bordered, banded diagonal set of equations by interleaving the primal and dual variables within the nominal and perturbed problems, while keeping the variables from each subproblem separate. If the p^{th} perturbation subproblem corresponds to a unit disturbance at some stage $k = \lfloor \frac{p-1}{l} \rfloor$, then the components of the system of equations (56) corresponding to the nominal variables and the variables for the p^{th} perturbation subproblem are coupled at all stages after k.

Considering for the moment only that part of (44) corresponding to the first perturbation problem (with $p = 1$), this reordering yields the coupled linear system

$$
\left[\begin{array}{l|l}
R \;\; D' \;\; B' & \\
D \;\; -\Sigma_0 \;\; 0 & \\
B \;\; 0 \;\; 0 \;\; -I & \\
\quad -I \;\; Q \;\; 0 \;\; C' \;\; A' & \\
\quad\quad 0 \;\; R \;\; D' \;\; B' & \\
\quad\quad C \;\; D \;\; -\Sigma_1 \;\; 0 & I \\
\quad\quad A \;\; B \;\; 0 \;\; 0 \;\; -I & \\
\quad\quad\quad -I \;\; Q \;\; 0 \;\; C' \;\; A' & \\
\quad\quad\quad\quad 0 \;\; R \;\; D' \;\; B' & \\
\quad\quad\quad\quad C \;\; D \;\; -\Sigma_2 \;\; 0 \;\; \ddots \;\; \ddots & \quad\quad I \\
\quad\quad\quad\quad A \;\; B \;\; 0 \;\; \ddots & \\
\quad\quad\quad\quad\quad\quad \ddots \;\; P \;\; Y' & \\
\quad\quad\quad\quad\quad\quad\quad Y \;\; -\Sigma_N & \quad\quad\quad\quad I \\
\hdashline
 & 0 \;\; 0 \;\; \bar{D}' \;\; B' \\
\quad I & 0 \;\; 0 \;\; H' \;\; 0 \\
 & \bar{D} \;\; H \;\; -\Sigma_1^1 \;\; 0 \\
 & B \;\; 0 \;\; 0 \;\; 0 \;\; -I \\
 & \quad -I \;\; 0 \;\; 0 \;\; 0 \;\; \bar{C}' \;\; A' \\
 & \quad\quad 0 \;\; 0 \;\; 0 \;\; \bar{D}' \;\; B' \\
\quad\quad I & \quad\quad 0 \;\; 0 \;\; 0 \;\; H' \;\; 0 \\
 & \quad\quad \bar{C} \;\; \bar{D} \;\; H \;\; -\Sigma_2^1 \;\; 0 \;\; \ddots \;\; \ddots \\
 & \quad\quad A \;\; B \;\; 0 \;\; 0 \;\; \ddots \\
 & \quad\quad\quad\quad \ddots \;\; 0 \;\; 0 \;\; \bar{Y}' \\
\quad\quad\quad\quad I & \quad\quad\quad\quad\quad 0 \;\; 0 \;\; H_f' \\
 & \quad\quad\quad\quad\quad \bar{Y} \;\; H_f \;\; -\Sigma_N^1
\end{array}\right]
\left[\begin{array}{c}
\Delta v_0 \\ \Delta\lambda_0 \\ \Delta\pi_0 \\ \Delta x_1 \\ \Delta v_1 \\ \Delta\lambda_1 \\ \Delta\pi_1 \\ \Delta x_2 \\ \Delta v_2 \\ \Delta\lambda_2 \\ \vdots \\ \Delta x_N \\ \Delta\lambda_N \\ \hdashline \Delta u_1^1 \\ \Delta\delta c_1^1 \\ \Delta\lambda_1^1 \\ \Delta\pi_1^1 \\ \Delta x_2^1 \\ \Delta u_2^1 \\ \Delta\delta c_2^1 \\ \Delta\lambda_2^1 \\ \vdots \\ \Delta x_N^1 \\ \Delta\delta c_N^1 \\ \Delta\lambda_N^1
\end{array}\right]
=
\left[\begin{array}{c}
r^{v_0} \\ r^{\lambda_0} \\ r^{\pi_0} \\ r^{x_1} \\ r^{v_1} \\ r^{\lambda_1} \\ r^{\pi_1} \\ r^{x_2} \\ r^{v_2} \\ r^{\lambda_2} \\ \vdots \\ r^{x_N} \\ r^{\lambda_N} \\ \hdashline r^{u_1^1} \\ r^{\delta c_1^1} \\ r^{\lambda_1^1} \\ r^{\pi_1^1} \\ r^{x_2^1} \\ r^{u_2^1} \\ r^{\delta c_2^1} \\ r^{\lambda_2^1} \\ \vdots \\ r^{x_N^1} \\ r^{\delta c_N^1} \\ r^{\lambda_N^1}
\end{array}\right].
\tag{57}
$$

The diagonal matrices Σ_i and Σ_i^p in (57) correspond to the matrix products $\Lambda^{-1}Z$ in (56), and are defined as

$$\Sigma_i := (\Lambda_i)^{-1}Z_i, \qquad \forall i \in \mathbb{Z}_{[0,N]} \tag{58}$$

$$\Sigma_i^p := (\Lambda_i^p)^{-1}Z_i^p, \quad \forall i \in \mathbb{Z}_{[k+1,N]}, \tag{59}$$

where the matrices Λ_i, Λ_i^p, Z_i, and Z_i^p are diagonal matrices formed from the Lagrange multipliers and slack variables λ_i, λ_i^p, z_i and z_i^p from the nominal and perturbation subproblems.

If all of the perturbation problems (44) are incorporated into a linear system of the form (57), the result is a system of equations whose coefficient matrix can be partitioned into block-bordered form as

$$
\begin{bmatrix}
\mathcal{A} & \mathcal{J}_1 & \mathcal{J}_2 & \cdots & \mathcal{J}_{lN} \\
\mathcal{J}_1' & \mathcal{B}_1 & & & \\
\mathcal{J}_2' & & \mathcal{B}_2 & & \\
\vdots & & & \ddots & \\
\mathcal{J}_{lN}' & & & & \mathcal{B}_{lN}
\end{bmatrix}
\begin{bmatrix}
\mathbf{x}_A \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{lN}
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{b}_A \\ \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_{lN}
\end{bmatrix},
\tag{60}
$$

where the banded matrix $\mathcal{A}$ is derived from the coefficients in the nominal problem (42), the banded matrices $\mathcal{B}_p$ are derived from the lN perturbation subproblems (44), and the matrices $\mathcal{J}_p$ represent the coupling between the systems. The vectors $\mathbf{b}_A$, $\mathbf{b}_p$, $\mathbf{x}_A$, and $\mathbf{x}_p$ (which should not be confused with the sequence of state vectors $\mathbf{x}$) are constructed from the primal and dual variables and residuals using the ordering in (57). The matrices $\mathcal{J}_p$ are constructed from identity matrices coupling the rows of $\mathcal{A}$ that contain the Σ_i terms with the columns of $\mathcal{B}_p$ that contain the H terms. It should of course be noted that for the matrix $\mathcal{B}_p$, corresponding to a unit disturbance at stage $k = \lfloor \frac{p-1}{l} \rfloor$, terms from stages prior to stage $k+1$ are not required.

4.3 Solving for an Interior-Point Step

We can now estimate the solution time for the robust optimization problem (41)–(44) by demonstrating that the linear system (60) can be solved in $\mathcal{O}((m+n)^3N^3)$ operations. We recall that, in practice, the number of interior-point iterations is only weakly dependent on the size of the problem [55]. Throughout this section, we make the simplifying assumption that the number of constraints s and r in (8) and (9) are $\mathcal{O}(m+n)$ and $\mathcal{O}(n)$, respectively.

We first require the following standing assumption and preliminary results:

Assumption 1 The constraint matrix D in (8) has full column rank.

Note that this assumption can always be satisfied by introducing additional input constraints with suitably large bounds. This allows us to derive the following two results, proofs for which can be found in Appendices B.1 and B.2 respectively.

Lemma 1 *For the robust control problem* (41)–(44)*, the Jacobian matrix in* (57) *has full rank.*

Lemma 2 *The sub-matrices $\mathcal{B}_p$ arising from the perturbation subproblems in* (60) *have full rank. Additionally, recalling that $k = \lfloor \frac{p-1}{l} \rfloor$,*

(i) A solution to the linear system $\mathcal{B}_p\mathbf{x}_p = \mathbf{b}_p$ can be found in $\mathcal{O}\left((m+n)^3(N-k+1)\right)$ operations.

(ii) If a solution to (i) above has been found, then a solution for each additional right hand side requires $\mathcal{O}\left((m+n)^2(N-k+1)\right)$ operations.

Note that each of the blocks $\mathcal{B}_p$ on the diagonal of (60) is banded and symmetric indefinite. Several methods exist for the stable construction of Cholesky-like decompositions of symmetric indefinite matrices into factors of the form LDL' [11], and efficient algorithms for performing this factorization for sparse matrices are freely available [18, 34]. However, it is generally not possible to guarantee that the banded structure of an indefinite matrix, such as $\mathcal{B}_p$, will be exploited using these methods if symmetry and stability of the factorization are to be preserved. Instead, the special structure of the matrices $\mathcal{B}_p$ allows us to employ a specialized technique for solution of the linear system $\mathcal{B}_p\mathbf{x}_p = \mathbf{b}_p$ based on a Riccati recursion [46, 51] in the proof of Lemma 2 in Appendix B.2.

We can now demonstrate that it is always possible to solve the linear system (60) in $\mathcal{O}((m+n)^3N^3)$ operations.

Theorem 2 *For the robust optimal control problem* (41)–(44)*, each primal-dual interior-point iteration requires no more than $\mathcal{O}((m+n)^3N^3)$ operations.*

Proof The linear system (60) can be factored and solved using a Schur complement technique, so that

$$\begin{bmatrix}\mathbf{x}_A\\ \mathbf{x}_1\\ \mathbf{x}_2\\ \vdots\\ \mathbf{x}_{lN}\end{bmatrix}=\begin{bmatrix}I & & & & \\ -\mathcal{B}_1^{-1}\mathcal{J}_1' & I & & & \\ -\mathcal{B}_2^{-1}\mathcal{J}_2' & & I & & \\ \vdots & & & \ddots & \\ -\mathcal{B}_{lN}^{-1}\mathcal{J}_{lN}' & & & & I\end{bmatrix}\begin{bmatrix}\Delta^{-1} & & & & \\ & \mathcal{B}_1^{-1} & & & \\ & & \mathcal{B}_2^{-1} & & \\ & & & \ddots & \\ & & & & \mathcal{B}_{lN}^{-1}\end{bmatrix}\begin{bmatrix}I & -\mathcal{J}_1\mathcal{B}_1^{-1} & -\mathcal{J}_1\mathcal{B}_2^{-1} & \dots & -\mathcal{J}_{lN}\mathcal{B}_{lN}^{-1}\\ & I & & & \\ & & I & & \\ & & & \ddots & \\ & & & & I\end{bmatrix}\begin{bmatrix}\mathbf{b}_A\\ \mathbf{b}_1\\ \mathbf{b}_2\\ \vdots\\ \mathbf{b}_{lN}\end{bmatrix},$$

with

$$\Delta := \mathcal{A} - \sum_{p=1}^{lN}\mathcal{J}_p\mathcal{B}_p^{-1}\mathcal{J}_p'.$$

where, by virtue of Lemma 1, the matrix Δ is always full rank [33, Thm. 0.8.5]. The $\mathcal{O}((m+n)^3N^3)$ complexity bound can then be attained by solving (60) using the following procedure:

Operation		**Complexity**	
solve: $\tilde{\mathbf{x}}_p = \mathcal{B}_p^{-1}\mathbf{b}_p$	$\forall p \in \mathbb{Z}_{[1,lN]}$	$lN \cdot \mathcal{O}((m+n)^3N)$	(61a)
$\mathcal{S}_p = \mathcal{J}_p\left(\mathcal{B}_p^{-1}\mathcal{J}_p'\right)$	$\forall p \in \mathbb{Z}_{[1,lN]}$	$lN \cdot \mathcal{O}((m+n)^3N^2)$	(61b)
factor: $\Delta = \mathcal{A} - \sum_{p=1}^{lN}\mathcal{S}_p$		$lN \cdot \mathcal{O}((m+n)N)$	(61c)
$= L_\Delta D_\Delta L_\Delta'$		$\mathcal{O}((m+n)^3N^3)$	(61d)
solve: $\mathbf{z}_A = \mathbf{b}_A - \sum_{p=1}^{lN}(\mathcal{J}_p\tilde{\mathbf{x}}_p),$		$lN \cdot \mathcal{O}((m+n)N)$	(61e)
$\mathbf{x}_A = (L_\Delta')^{-1}(D_\Delta^{-1}(L_\Delta^{-1}\mathbf{z}_A)),$		$\mathcal{O}((m+n)^2N^2)$	(61f)
$\mathbf{z}_p = \mathcal{J}_p'\mathbf{x}_A,$	$\forall p \in \mathbb{Z}_{[1,lN]}$	$lN \cdot \mathcal{O}((m+n)N)$	(61g)
$\mathbf{x}_p = \tilde{\mathbf{x}}_p - \mathcal{B}_p^{-1}\mathbf{z}_p.$	$\forall p \in \mathbb{Z}_{[1,lN]}$	$lN \cdot \mathcal{O}((m+n)^2N).$	(61h)

The complexity of the solution to the linear system (61a) follows from Lemma 2(i). The complexity of the solution to (61b) and (61h) follows from Lemma 2(ii), where each of the matrices $\mathcal{J}_p'$ in (61b) have $\mathcal{O}((m+n)N)$ non-zero columns.

Remark 5 For the solution procedure in (61), it is important to note that since the coupling matrices $\mathcal{J}_i$ have no more than a single 1 on every row and column, matrix products involving left or right multiplication by $\mathcal{J}_i$ or $\mathcal{J}_i'$ do not require *any* floating point operations to calculate. The reader is referred to [10, App. C] for a more complete treatment of complexity analysis for matrix operations.

Remark 6 If the solution procedure (61) is employed, then the robust optimization problem is an obvious candidate for parallel implementation. However, it is generally not necessary to hand implement the suggested variable interleaving and block factorization procedure to realize the suggested block-bordered structure in (60) and $\mathcal{O}((m+n)^3N^3)$ solution time, as any reasonably efficient sparse factorization code can be expected to perform similar steps automatically; see [18]. Note that the “arrowhead” structure in (60) should be reversed (i.e. pointing down and to the right) in order for direct LDL' factorization to produce sparse factors.

Remark 7 Recalling the discussion of soft constraint handling in Section 2.4 and Remark 3, it is easy to show that the inclusion of soft constraints does not qualitatively alter the complexity results of Theorem 2, since the inclusion of such constraints amounts only to a modification of the matrix $\mathcal{A}$ (and thus of the dense matrix Δ) in (61c), and does *not* effect the complexity of any of the operations involving the banded matrices $\mathcal{B}_i$.

5 Results

Two sparse QP solvers were used to evaluate the proposed formulation. The first, OOQP [24], uses a primal-dual interior-point approach configured with the sparse factorization code MA27 from the HSL library [34] and the OOQP version of the multiple-corrector interior-point method of Gondzio [25].

The second sparse solver used was the QP interface to the PATH [17] solver. This code solves mixed complementarity problems using an active-set method, and hence can be applied to the stationary conditions of any quadratic program. Note that since we are dealing with convex QPs, each optimization problem and its associated complementarity system have equivalent solution sets.

All results reported in this section were generated on a single processor machine with a 3 GHz Pentium 4 processor and 2GB of RAM. We restrict our attention to sparse solvers as the amount of memory required for the problems considered is prohibitively large for dense factorization methods.

A set of test cases was generated to compare the performance of the two sparse solvers using the $(\mathbf{M},\mathbf{v})$ formulation in (26) and the decomposition-based method of Section 3. Each test case is defined by its number of states n and horizon length N. The remaining problem parameters were chosen using the following rules:

- There are twice as many states as inputs.
- The constraint sets W, $\mathcal{Z}$ and X_f represent randomly selected symmetric bounds on the states and inputs subjected to a random similarity transformation.
- The state space matrices A and B are randomly generated, with (A,B) controllable, and with A potentially unstable.
- The dimension l of the generating disturbance is chosen as half the number of states, with randomly generated E of full column rank.
- All test cases have feasible solutions. The initial state is selected such that at least some of the inequality constraints in (42b) are active at the optimal solution.

The average computational times required by each of the two solvers for the two problem formulations for a range of problem sizes are shown in Table 1. Each entry represents the average of ten test cases, unless otherwise noted.

It is clear from these results that, as expected, the decomposition-based formulation can be solved much more efficiently than the original $(\mathbf{M},\mathbf{v})$ formulation for robust optimal control problems of nontrivial size, and that the difference in solution times increases dramatically with increased problem dimension. Additionally, the decomposition formulation seems particularly well suited to the interior-point

Table 1 Average Solution Times (sec)

	$(\mathbf{M},\mathbf{v})$		Decomposition	
Problem Size	OOQP	PATH	OOQP	PATH
2 states, 4 stages	0.004	0.003	0.004	0.004
2 states, 8 stages	0.020	0.010	0.016	0.019
2 states, 12 stages	0.061	0.027	0.037	0.052
2 states, 16 stages	0.172	0.091	0.072	0.198
2 states, 20 stages	0.432	0.123	0.132	1.431
4 states, 4 stages	0.024	0.026	0.018	0.024
4 states, 8 stages	0.220	0.316	0.099	0.357
4 states, 12 stages	0.969	1.162	0.264	2.019
4 states, 16 stages	3.755	17.50	0.576	16.63
4 states, 20 stages	11.67	41.45	1.047	22.26
8 states, 4 stages	0.667	1.282	0.136	0.261
8 states, 8 stages	7.882	81.50	0.858	14.89
8 states, 12 stages	46.97	$257.9^{\dagger}$	2.81	$183.8^{\dagger}$
8 states, 16 stages	189.75	$2660^{\dagger}$	6.781	$288.9^{\dagger}$
8 states, 20 stages	620.3	x	13.30	x
12 states, 4 stages	6.292	75.608	0.512	5.044
12 states, 8 stages	132.1	$1160^{\dagger}$	4.671	$388.9^{\dagger}$
12 states, 12 stages	907.4	x	14.08	x
12 states, 16 stages	x	x	37.99	x
12 states, 20 stages	x	x	82.06	x

x – Solver failed all test cases

†– Based on limited data set due to failures

solver (OOQP), rather than the active set method (PATH). Nevertheless we expect the performance of active set methods to improve relative to interior-point methods when solving a sequence of similar QPs that would occur in predictive control, where a good estimate of the optimal active set is typically available at the start of computation. That is, interior-point methods are particularly effective in “cold start” situations, while the efficiency of active set methods is likely to improve given a “warm start”.

Figure 1 shows that the interior-point solution time increases cubicly with horizon length for randomly generated problems with 2, 4, 8 and 12 states. The performance closely matches the predicted behavior described in Section 3. For the particular problems shown, the number of iterations required for the OOQP algorithm to converge varied from 12 to 20 over the range of horizon lengths and state dimensions considered.

6 Conclusions and Future Work

We have derived a highly efficient computational method for calculation of affine state feedback policies for robust control of constrained systems with bounded disturbances. This is done by exploiting the structure of the underlying optimization problem and deriving an equivalent problem with considerable structure and sparsity, resulting in a problem formulation that is particularly suited to an interior-point solution method. As a result, robustly stabilizing receding horizon control laws based on optimal state-feedback policies have become practically realizable, even for systems of significant size or with long horizon lengths.

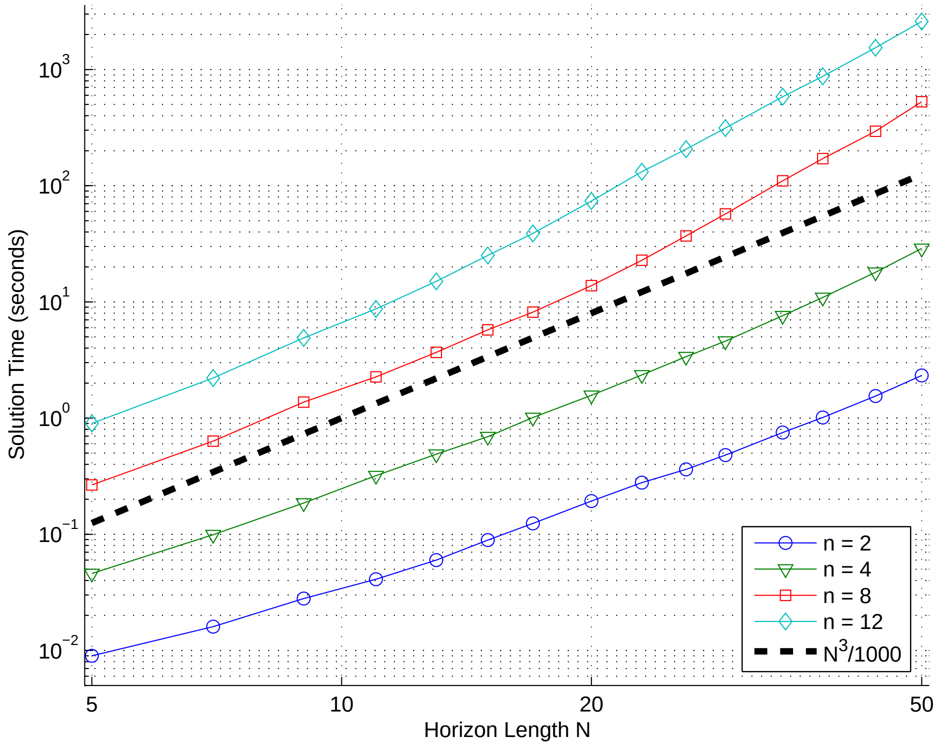

Fig. 1 Computation time vs. horizon length for systems of increasing state dimension, using the decomposition method and OOQP solver. Also shown is the constant line $N^3/1000$ for comparison.

In Section 4 we proved that, when applying an interior-point solution technique to our robust optimal control problem, each iteration of the method can be solved using a number of operations proportional to the cube of the control horizon length. We appeal to the Riccati based factorization technique in [46, 51] to support this claim. However, we stress that the results in Section 5, which demonstrate this cubic-time behavior numerically, are based on freely available optimization and linear algebra packages and *do not* rely on any special factorization methods.

A number of open research issues remain. It may be possible to further exploit the structure of our control problem by developing specialized factorization algorithms for the factorization of each interior-point step, e.g. through the parallel block factorization procedure alluded to in Remark 6. It may also be possible to achieve considerably better performance by placing further constraints on the structure of the disturbance feedback matrix **M**, though this appears difficult to do if the attractive invariance and stability properties of the present formulation are to be preserved.

Many of the system-theoretic results developed in [28] hold for a fairly broad class of disturbances and cost functions [26]. For example, when the disturbance is Gaussian the problem may be modified to require that the state and input constraints hold with a certain pre-specified probability, and the probabilistic constraints converted to second-order cone constraints [10, pp. 157–8]. Alternatively, the cost function for the finite horizon control problem may require the minimization of the finite-horizon ℓ_2 gain of a system [27, 36]. In all of these cases, there is a strong possibility that the underlying problem structure may be exploited to realize a substantial increase in computational efficiency.

References

1. Anderson, B.D.O., Moore, J.B.: Optimal control: linear quadratic methods. Prentice-Hall, Inc., Upper Saddle River, NJ, USA (1990)
2. Bemporad, A.: Reducing conservativeness in predictive control of constrained systems with disturbances. In: Proc. 37th IEEE Conference on Decision and Control, pp. 1384–1391. Tampa, FL, USA (1998)
3. Bemporad, A., Borrelli, F., Morari, M.: Min-max control of constrained uncertain discrete-time linear systems. IEEE Transactions on Automatic Control **48**(9), 1600–1606 (2003)
4. Bemporad, A., Morari, M.: Robust Model Predictive Control: A Survey *in* Robustness in Identification and Control, *Lecture Notes in Control and Information Sciences*, vol. 245, pp. 207–226. Ed. A. Garulli, A. Tesi, and A. Vicino. Springer-Verlag, Berlin. (1999)
5. Ben-Tal, A., Boyd, S., Nemirovski, A.: Extending scope of robust optimization: Comprehensive robust counterparts of uncertain problems. Mathematical Programming **107**(1–2), 63–89 (2006)
6. Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A.: Adjustable robust solutions of uncertain linear programs. Mathematical Programming **99**(2), 351–376 (2004)
7. Bertsekas, D.P., Rhodes, I.B.: Sufficiently informative functions and the minimax feedback control of uncertain dynamic systems. IEEE Transactions on Automatic Control **AC-18**(2), 117–124 (1973)
8. Biegler, L.: Efficient solution of dynamic optimization and NMPC problems. In: F. Allgöwer, A. Zheng (eds.) Nonlinear Model Predictive Control, *Progress in Systems and Control Theory*, vol. 26, pp. 219–243. Birkhäuser (2000)
9. Blanchini, F.: Set invariance in control. Automatica **35**(1), 1747–1767 (1999)
10. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)
11. Bunch, J.R., Kaufman, L., Parlett, B.N.: Decomposition of a symmetric matrix. Numerische Mathematik **27**, 95–110 (1976)
12. Camacho, E.F., Bordons, C.: Model Predictive Control, second edn. Springer-Verlag, London. (2004)
13. Chisci, L., Rossiter, J.A., Zappa, G.: Systems with persistent state disturbances: predictive control with restricted constraints. Automatica **37**(7), 1019–1028 (2001)
14. Dahleh, M.A., Diaz-Bobillo, I.J.: Control of Uncertain Systems. Prentice Hall, Englewood Cliffs, NJ, USA (1995)
15. Diehl, M., Björnberg, J.: Robust dynamic programming for min-max model predictive control of constrained uncertain systems. IEEE Transactions on Automatic Control **49**(12), 2253–2257 (2004)
16. Diehl, M., Bock, H.G., Schlöder, J.P.: A real-time iteration scheme for nonlinear optimization in optimal feedback control. SIAM Journal on Optimization **43**(5), 1714–1736 (2005)
17. Dirske, S.P., Ferris, M.C.: The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems. Optimization Methods and Software **5**, 123–156 (1995)
18. Duff, I., Erisman, A., Reid, J.: Direct Methods for Sparse Matrices. Oxford University Press, Oxford, England (1986)
19. Dullerud, G.E., Paganini, F.: A Course in Robust Control Theory: A Convex Approach. Springer-Verlag, New York (2000)
20. Dunn, J.C., Bertsekas, D.P.: Efficient dynamic programming implementations of Newton's method for unconstrained optimal control problems. Journal of Optimization Theory and Applications **63**(1), 23–38 (1989)
21. Fialho, I.J., Georgiou, T.T.: ℓ_1 state-feedback control with a prescribed rate of exponential convergence. IEEE Transactions on Automatic Control **42**(10), 1476–81 (1997)
22. Fletcher, R.: Practical methods of optimization, 2nd edn. Wiley-Interscience, New York, NY, USA (1987)

23. Garstka, S.J., Wets, R.J.B.: On decision rules in stochastic programming. Mathematical Programming **7**, 117–143 (1974)
24. Gertz, E.M., Wright, S.J.: Object-oriented software for quadratic programming. ACM Transactions on Mathematical Software **29**, 58–81 (2003)
25. Gondzio, J.: Multiple centrality corrections in a primal-dual method for linear programming. Computational Optimization and Applications **6**, 137–156 (1996)
26. Goulart, P.J.: Affine feedback policies for robust control with constraints. Ph.D. thesis, University of Cambridge (2006). URL `http:\\www-control.eng.cam.ac.uk`
27. Goulart, P.J., Kerrigan, E.: A convex formulation for receding horizon control of constrained discrete-time systems with guaranteed ℓ_2 gain. In: Proc. 45th IEEE Conference on Decision and Control and 2005 European Control Conference (2006)
28. Goulart, P.J., Kerrigan, E.C., Maciejowski, J.M.: Optimization over state feedback policies for robust control with constraints. Automatica **42**(4), 523–533 (2006)
29. Green, M., Limebeer, D.J.N.: Linear Robust Control. Prentice Hall (1995)
30. Guslitser, E.: Uncertainty-immunized solutions in linear programming. Master's thesis, Technion, Israeli Institute of Technology (2002)
31. van Hessem, D.H.: Stochastic inequality constrained closed-loop model predictive control. Ph.D. thesis, Technical University of Delft (2004)
32. van Hessem, D.H., Bosgra, O.H.: A conic reformulation of model predictive control including bounded and stochastic disturbances under state and input constraints. In: Proc. 41st IEEE Conference on Decision and Control, pp. 4643–4648 (2002)
33. Horn, R., Johnson, C.: Matrix Analysis. Cambridge University Press (1985)
34. HSL: HSL 2002: A collection of Fortran codes for large scale scientific computation. http://www.cse.clrc.ac.uk/nag/hsl (2002). URL `www.cse.clrc.ac.uk/nag/hsl`
35. Jacobson, D., Mayne, D.: Differential Dynamic Programming. Elsevier, New York, NY, USA (1970)
36. Kerrigan, E.C., Alamo, T.: A convex parameterization for solving constrained min-max problems with a quadratic cost. In: Proc. 2004 American Control Conference, pp. 2220–2221. Boston, MA, USA (2004)
37. Kerrigan, E.C., Maciejowski, J.M.: On robust optimization and the optimal control of constrained linear systems with bounded state disturbances. In: Proc. 2003 European Control Conference. Cambridge, UK (2003)
38. Lee, Y.I., Kouvaritakis, B.: Constrained receding horizon predictive control for systems with disturbances. International Journal of Control **72**(11), 1027–1032 (1999)
39. Löfberg, J.: Approximations of closed-loop MPC. In: Proc. 42nd IEEE Conference on Decision and Control, pp. 1438–1442. Maui, Hawaii, USA (2003)
40. Löfberg, J.: Minimax approaches to robust model predictive control. Ph.D. thesis, Linköping University (2003)
41. Maciejowski, J.M.: Predictive Control with Constraints. Prentice Hall, UK (2002)
42. Mayne, D.Q.: Control of constrained dynamic systems. European Journal of Control **7**, 87–99 (2001). Survey paper
43. Mayne, D.Q., Rawlings, J.B., Rao, C.V., Scokaert, P.O.M.: Constrained model predictive control: Stability and optimality. Automatica **36**(6), 789–814 (2000). Survey paper
44. Mayne, D.Q., Seron, M.M., Raković, S.V.: Robust model predictive control of constrained linear systems with bounded disturbances. Automatica **41**(2), 219–24 (2005)
45. Qin, S.J., Badgwell, T.A.: A survey of industrial model predictive control technology. Control Engineering Practice **11**, 733–764 (2003)
46. Rao, C.V., Wright, S.J., Rawlings, J.B.: Application of interior–point methods to model predictive control. Journal of Optimization Theory and Applications **99**, 723–757 (1998)
47. Scokaert, P.O.M., Mayne, D.Q.: Min-max feedback model predictive control for constrained linear systems. IEEE Transactions on Automatic Control **43**(8), 1136–1142 (1998)
48. Scokaert, P.O.M., Rawlings, J.B.: Feasibility issues in linear model predictive control. AIChE J. **45**(8), 1649–1659 (1999)
49. Shamma, J.S.: Optimization of the ℓ^∞-induced norm under full state feedback. IEEE Transactions on Automatic Control **41**(4), 533–44 (1996)
50. Stein, G.: Respect the unstable. IEEE Control Systems Magazine **34**(4), 12–25 (2003)
51. Steinbach, M.C.: Fast recursive SQP methods for large-scale optimal control problems. Ph.D. thesis, University of Heidelberg (1995)
52. Sznaier, M., Bu, J.: Mixed ℓ_1/H_∞ control of MIMO systems via convex optimization. IEEE Transactions on Automatic Control **43**(9), 1229–1241 (1998)

53. Witsenhausen, H.S.: A minimax control problem for sampled linear systems. IEEE Transactions on Automatic Control **AC-13**(1), 5–21 (1968)
54. Wright, S.J.: Interior point methods for optimal control of discrete-time systems. Journal of Optimization Theory and Applications **77**, 161–187 (1993)
55. Wright, S.J.: Primal-Dual Interior-Point Methods. SIAM Publications, Philadelphia (1997)
56. Youla, D.C., Jabr, H.A., Bongiorno, J.J.: Modern Weiner-Hopf design of optimal controllers: Part II. IEEE Transactions on Automatic Control **AC-21**, 319–338 (1976)
57. Zames, G.: Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. IEEE Transactions on Automatic Control **AC-26**(2), 301–320 (1981)
58. Zhou, K., Doyle, J., Glover, K.: Robust and Optimal Control. Prentice-Hall (1996)

A Matrix Definitions

Let the matrices $\mathbf{A} \in \mathbb{R}^{n(N+1)\times n}$ and $\mathbf{E} \in \mathbb{R}^{n(N+1)\times nN}$ be defined as

$$\mathbf{A} := \begin{bmatrix} I_n \\ A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \quad \mathbf{E} := \begin{bmatrix} 0 & 0 & \cdots & 0 \\ I_n & 0 & \cdots & 0 \\ A & I_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1} & A^{N-2} & \cdots & I_n \end{bmatrix}. \tag{62}$$

We also define the matrices $\mathbf{B} \in \mathbb{R}^{n(N+1)\times mN}$, $\mathbf{C} \in \mathbb{R}^{(sN+r)\times n(N+1)}$ and $\mathbf{D} \in \mathbb{R}^{(sN+r)\times mN}$ as

$$\mathbf{B} := \mathbf{E}(I_N \otimes B), \; \mathbf{C} := \begin{bmatrix} I_N \otimes C & 0 \\ 0 & Y \end{bmatrix}, \; \mathbf{D} := \begin{bmatrix} I_N \otimes D \\ 0 \end{bmatrix}. \tag{63}$$

and define $F := \mathbf{CB} + \mathbf{D}$, $G := \mathbf{CE}$, $T := -\mathbf{CA}$, $c := \left[\begin{smallmatrix} \mathbf{1}_N \otimes b \\ z \end{smallmatrix}\right]$.

B Rank of the Jacobian and Reduction to Riccati Form

B.1 Rank of the Robust Control Problem Jacobian (Proof of Lemma 1)

We demonstrate that the Jacobian matrix defined in (57) is always full rank. Recalling the discussion in Section 4.1, for *any* quadratic program the Jacobian matrix is full rank if the only solution to the system

$$\begin{bmatrix} \mathsf{Q} & \mathsf{A}' & \mathsf{C}' \\ \mathsf{A} & 0 & 0 \\ \mathsf{C} & 0 & -\Sigma \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\pi \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{64}$$

satisfies $\Delta\theta = 0$, $\Delta\pi = 0$, and $\Delta\lambda = 0$, where $\Sigma := \Lambda^{-1}Z \succ 0$, $\mathsf{Q} \succeq 0$ and the coefficient matrices A and C come from the equality and inequality constraints of the QP respectively (cf. (52)). From the first two rows of this system,

$$\Delta\theta'\mathsf{Q}\Delta\theta + (\Delta\theta'\mathsf{A}')\Delta\pi + \Delta\theta'\mathsf{C}'\Delta\lambda = \Delta\theta'\mathsf{Q}\Delta\theta + \Delta\theta'\mathsf{C}'\Delta\lambda = 0. \tag{65}$$

Incorporating the final block row, $\mathsf{C}\Delta\theta = \Sigma\Delta\lambda$, we have

$$\Delta\theta'\mathsf{Q}\Delta\theta + \Delta\lambda'\Sigma\Delta\lambda = 0. \tag{66}$$

Since $\mathsf{Q} \succeq 0$ for a convex QP and $\Sigma \succ 0$ for a strictly interior point, we conclude that $\Delta\lambda = 0$. We next make use of the following matrix condition, which is easily verified:

Fact 1 The matrix $\left[\begin{smallmatrix} X & Y \\ 0 & Z \end{smallmatrix}\right]$ is full column rank for any Y if both X and Z are full column rank.

Since $\Delta\lambda = 0$ always holds, sufficient conditions to guarantee $\Delta\theta = 0$ and $\Delta\pi = 0$ in (64) are that:

(i) A is full row rank.
(ii) $\left[\begin{smallmatrix}\mathsf{A}\\ \mathsf{C}\end{smallmatrix}\right]$ is full column rank.

For the quadratic program defined by the robust control problem (42)-(44), the equality and inequality constraints are defined as in (47). For this convex QP, it is straightforward to show that the above rank conditions on A and C are equivalent to requiring that:

(i) Each of the matrices $\mathsf{A}_0, \mathsf{A}_1, \dots, \mathsf{A}_{lN}$ is full row rank.
(ii) Each of the matrices $\begin{bmatrix}\mathsf{A}_0\\ \mathsf{C}_0\end{bmatrix}, \begin{bmatrix}\mathsf{A}_1\\ \mathsf{C}_1\end{bmatrix}, \dots, \begin{bmatrix}\mathsf{A}_{lN}\\ \mathsf{C}_{lN}\end{bmatrix}$ is full column rank.

The condition (ii) is derived by noting that, for the particular problem (42)-(44), the general rank condition on $\left[\begin{smallmatrix}\mathsf{A}\\ \mathsf{C}\end{smallmatrix}\right]$ is equivalent to requiring that the matrix

$$\begin{bmatrix}
\mathsf{C}_0 & \mathsf{J}_1 & \mathsf{J}_2 & \dots & \mathsf{J}_{lN} \\
\mathsf{A}_0 & & & & \\
 & \mathsf{C}_1 & & & \\
 & \mathsf{A}_1 & & & \\
 & & \mathsf{C}_2 & & \\
 & & \mathsf{A}_2 & & \\
 & & & \ddots & \\
 & & & & \mathsf{C}_{lN} \\
 & & & & \mathsf{A}_{lN}
\end{bmatrix}$$

is full column rank, which reduces to (ii) upon repeated application of Fact 1 above to eliminate the coupling terms J_p. If Assumption 1 holds, both of these rank conditions are easily verified by examination of the definitions in (48) and (50). The Jacobian matrix for the QP defined in (42)-(44) is thus full rank, and it remains full rank if its rows and columns are reordered as in (57).

B.2 Solution of $\mathcal{B}_p\mathbf{x}_p = \mathbf{b}_p$ via Riccati recursion (Proof of Lemma 2)

We demonstrate that the system of equations $\mathcal{B}_p\mathbf{x}_p = \mathbf{b}_p$ has a unique solution for every $\mathbf{b}_p$, where $\mathcal{B}_p$, $\mathbf{x}_p$ and $\mathbf{b}_p$ are defined as

$$\mathcal{B}_p := \left[\begin{array}{ccccccccccccc}
0 & 0 & \bar{D}' & B' & & & & & & & & & \\
0 & 0 & H' & 0 & & & & & & & & & \\
\bar{D} & H & -\Sigma^p_{k+1} & 0 & & & & & & & & & \\
B & 0 & 0 & 0 & -I & & & & & & & & \\
 & & & -I & 0 & 0 & 0 & \bar{C}' & A' & & & & \\
 & & & & 0 & 0 & 0 & \bar{D}' & B' & & & & \\
 & & & & 0 & 0 & 0 & H' & 0 & & & & \\
 & & & & \bar{C} & \bar{D} & H & -\Sigma^p_{k+2} & 0 & \ddots & & & \\
 & & & & A & B & 0 & 0 & & \ddots & & & \\
 & & & & & & & & & \ddots & 0 & 0 & \bar{Y}' \\
 & & & & & & & & & & 0 & 0 & H_f' \\
 & & & & & & & & & & \bar{Y} & H_f & -\Sigma^p_N
\end{array}\right], \tag{67a}$$

$$\mathbf{x}_p := (\Delta u^p_{k+1}, \Delta\delta c^p_{k+1}, \Delta\lambda^p_{k+1}, \Delta\pi^p_{k+1}, \Delta x^p_{k+2}, \Delta u^p_{k+2}, \Delta\delta c^p_{k+2}, \Delta\lambda^p_{k+2}, \dots, \Delta x^p_N, \Delta\delta c^p_N, \Delta\lambda^p_N),$$
$$\mathbf{b}_p := (\ r^{u^p_{k+1}}\ ,\ r^{\delta c^p_{k+1}}\ ,\ r^{\lambda^p_{k+1}}\ ,\ r^{\pi^p_{k+1}}\ ,\ r^{x^p_{k+2}}\ ,\ r^{u^p_{k+2}}\ ,\ r^{\delta c^p_{k+2}}\ ,\ r^{\lambda^p_{k+2}}\ , \dots,\ r^{x^p_N}\ ,\ r^{\delta c^p_N}\ ,\ r^{\lambda^p_N}\ )$$

and $k = \lfloor \frac{p-1}{l} \rfloor$, and that this solution is obtainable in $\mathcal{O}((m+n)^3 N)$ time. We first perform a single step of block elimination on the variables $\Delta\lambda^p_i$ and $\Delta\delta c^p_{k+1}$, so that the resulting linear system is solvable via specialized methods based on Riccati recursion techniques [46, 51] (see also related results in [20] for the unconstrained case).

It is straightforward to eliminate the terms $\Delta\lambda_i^p$ and $\Delta\delta c_i^p$ from each of the subproblems, yielding a linear system $\tilde{\mathcal{B}}_p\tilde{\mathbf{x}}_p = \tilde{\mathbf{b}}_p$. The coefficient matrix$\tilde{\mathcal{B}}_p$ is:

$$\tilde{\mathcal{B}}_p := \begin{bmatrix} R_{k+1}^p & B' & & & & & & & & \\ B & 0 & -I & & & & & & & \\ & -I & Q_{k+2}^p & M_{k+2}^p & A' & & & & & \\ & & (M_{k+2}^p)' & R_{k+2}^p & B' & & & & & \\ & & A & B & 0 & -I & & & & \\ & & & & -I & Q_{k+3}^p & M_{k+3}^p & A' & & \\ & & & & & (M_{k+3}^p)' & R_{k+3}^p & B' & \ddots & \\ & & & & & A & B & \ddots & \ddots & \\ & & & & & & & & \ddots & Q_N^p \end{bmatrix} \tag{68}$$

where, for stages $i \in \mathbb{Z}_{[k+1,N-1]}$:

$$\Phi_i^p := H'(\Sigma_i^p)^{-1}H \tag{69a}$$

$$\Theta_i^p := (\Sigma_i^p)^{-1} - (\Sigma_i^p)^{-1}H(\Phi_i^p)^{-1}H'(\Sigma_i^p)^{-1} \tag{69b}$$

$$Q_i^p := \bar{C}'\Theta_i^p\bar{C} \tag{69c}$$

$$R_i^p := \bar{D}'\Theta_i^p\bar{D} \tag{69d}$$

$$M_i^p := \bar{C}'\Theta_i^p\bar{D}, \tag{69e}$$

and for stage N:

$$\Phi_N^p := H_f'(\Sigma_N^p)^{-1}H_f \tag{69f}$$

$$\Theta_N^p := (\Sigma_N^p)^{-1} - (\Sigma_N^p)^{-1}H_f(\Phi_N^p)^{-1}H_f'(\Sigma_N^p)^{-1} \tag{69g}$$

$$Q_N^p := \bar{Y}'\Theta_N^p\bar{Y}. \tag{69h}$$

The vectors $\tilde{\mathbf{x}}_p$ and $\tilde{\mathbf{b}}_p$ are defined as:

$$\tilde{\mathbf{x}}_p := (\Delta u_{k+1}^p, \Delta\pi_{k+1}^p, \Delta x_{k+2}^p, \Delta u_{k+2}^p, \Delta\pi_{k+2}^p, \ldots, \Delta x_N^p) \tag{70}$$

$$\tilde{\mathbf{b}}_p := (\ \tilde{r}^{u_{k+1}^p}\ ,\ r^{\pi_{k+1}^p}\ ,\ \tilde{r}^{x_{k+2}^p}\ ,\ \tilde{r}^{u_{k+2}^p}\ ,\ r^{\pi_{k+2}^p}\ ,\ldots,\ \tilde{r}^{x_N^p}\ )\ , \tag{71}$$

where, for stages $i \in \mathbb{Z}_{[k+1,N-1]}$:

$$\tilde{r}^{x_i^p} := r^{x_i^p} + \bar{C}\left(\Theta_i^p r^{\lambda_i^p} - (\Sigma_i^p)^{-1}H(\Phi_i^p)^{-1}r^{\delta c_i^p}\right) \tag{72a}$$

$$\tilde{r}^{u_i^p} := r^{u_i^p} + \bar{D}\left(\Theta_i^p r^{\lambda_i^p} - (\Sigma_i^p)^{-1}H(\Phi_i^p)^{-1}r^{\delta c_i^p}\right), \tag{72b}$$

and, for stage N:

$$\tilde{r}^{x_N^p} := r^{x_N^p} + \bar{Y}\left(\Theta_N^p r^{\lambda_N^p} - (\Sigma_N^p)^{-1}H_f(\Phi_N^p)^{-1}r^{\delta c_N^p}\right). \tag{72c}$$

Remark 8 The matrix $\tilde{\mathcal{B}}_p$ is equivalent to the KKT matrix for the unconstrained control problem:

$$\min_{\substack{u_{k+1},\ldots,u_{N-1},\\ x_{k+1},\ldots,x_N}} \left(\frac{1}{2}x_N'Q_N^p x_N + \sum_{i=(k+1)}^{N-1}\frac{1}{2}(x_i'Q_i^p x_i + u_i'R_i^p u_i + 2x_i M_i^p u_i)\right) \tag{73}$$

subject to:

$$x_k = E_{(j)}, \tag{74a}$$

$$x_{i+1} = Ax_i + Bu_i, \quad \forall i \in \mathbb{Z}_{[k+1,N-1]}. \tag{74b}$$

Lemma 3 *Each of the matrices R_i^p, Q_i^p and Q_N^p are positive semi-definite. If Assumption 1 holds, then R_i^p is positive definite.*

Proof Recall that the matrix Θ_i^p is defined as

$$\Theta_i^p = (\Sigma_i^p)^{-1} - (\Sigma_i^p)^{-1} H \left(H'(\Sigma_i^p)^{-1} H\right)^{-1} H'(\Sigma_i^p)^{-1}, \tag{75}$$

and partition the diagonal and positive definite matrix Σ_i^p into $\Sigma_i^p = \left[\begin{smallmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{smallmatrix}\right]$. Recalling that $H := -\left[\begin{smallmatrix} I \\ I \end{smallmatrix}\right]$, Θ_i^p can be written as

$$\Theta_i^p = \begin{bmatrix} \Sigma_1^{-1} - \Sigma_1^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_1^{-1} & -\Sigma_1^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_2^{-1} \\ -\Sigma_2^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_1^{-1} & \Sigma_1^{-1} - \Sigma_2^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_2^{-1} \end{bmatrix} \tag{76}$$

$$= \begin{bmatrix} I \\ -I \end{bmatrix} (\Sigma_1 + \Sigma_2)^{-1} \, [I \;\; -I] \tag{77}$$

which is easily verified using standard matrix identities and the fact that the matrices Σ_1 and Σ_2 are diagonal.

Recalling that $\bar{D} := \left[\begin{smallmatrix} D \\ -D \end{smallmatrix}\right]$, it follows that R_i^p is positive semidefinite since it can be written as

$$R_i^p = \bar{D}' \begin{bmatrix} I \\ -I \end{bmatrix} (\Sigma_1 + \Sigma_2)^{-1} \, [I \;\; -I] \, \bar{D} \tag{78}$$

$$= 4D'(\Sigma_1 + \Sigma_2)^{-1} D \succeq 0. \tag{79}$$

If Assumption 1 holds, so that D is full column rank, then R_i^p is positive definite. A similar argument establishes the result for Q_i^p and Q_N^p.

We are now in a position to prove Lemma 2. Since R_i^p is positive definite and Q_i^p and Q_N^p are positive semidefinite, the linear system $\tilde{\mathcal{B}}_p \tilde{\mathbf{x}}_p = \tilde{\mathbf{b}}_p$ (and consequently the original system $\mathcal{B}_p \mathbf{x}_p = \mathbf{b}_p$) has a unique solution that can found in $\mathcal{O}((m+n)^3(N-k+1))$ operations using the Riccati recursion procedure described in [46, 51]. Once such a solution has been obtained, a solution for each additional right hand side requires $\mathcal{O}\left((m+n)^2(N-k+1)\right)$ operations [46, Sec. 3.4]. We note that in [46] the Riccati factorization procedure is shown to be numerically stable, and that similar arguments can be used to show that factorization of (68) is also stable. We omit details of this for brevity.