A Fast Well-conditioned Interior Point Method for Predictive Control

Amir Shahzad, Eric C. Kerrigan and George A. Constantinides

***Abstract*— Interior point methods (IPMs) have proven to be an efficient way of solving quadratic programming problems in predictive control. A linear system of equations needs to be solved in each iteration of an IPM. The ill-conditioning of this linear system in the later iterations of the IPM prevents the use of an iterative method in solving the linear system due to a very slow rate of convergence; in some cases the solution never reaches the desired accuracy. In this paper we propose the use of a well-conditioned, approximate linear system, which increases the rate of convergence of the iterative method. The computational advantage is obtained by the use of an inexact Newton method along with the use of novel preconditioners. Numerical results indicate that the computational complexity of our proposed method scales quadratically with the number of states and linearly with the horizon length.**

I. INTRODUCTION

This paper addresses the computational issues involved in solving a finite horizon optimal control problem for a discrete-time linear system with quadratic cost and linear inequality constraints on the states and control inputs. In predictive control, at each sampling instant with the given state information, a suitably-defined quadratic program (QP) is solved to obtain a sequence of inputs and only the first input is applied to the plant. This process is repeated at every sample instant using the current state estimate.

There are essentially two popular ways to solve the finite horizon optimal control problem, namely *explicit* and *implicit*. In the explicit approach states are eliminated from the objective function and constraints [1], which results in a small, but dense Hessian in the QP. The computational complexity of this approach in each iteration of an IPM is $\mathcal{O}((l+m)m^2N^3)$, where l is the number of constrained variables, m is the number of inputs, and N is the horizon length. In the implicit approach states are considered as unknowns and state equations are treated as equality constraints, which results in a large, but sparse Hessian. In the implicit approach, it is shown in [2] that the computational complexity in each iteration of an IPM can be reduced to $\mathcal{O}((n+m)^3 + l(m+n)^2)N)$, where n is the number of states, using a Riccati recursion scheme.

In recent years, attempts have been made to use predictive control in fast processes with a short sampling time. To reduce the computational load in solving the finite horizon optimal control problem, new techniques have emerged [3]–[5]. In [3] a large number of QPs are solved off-line for all possible initial states of the plant, then an explicit function is formed using the solutions of the QPs. This approach is generally applicable to small-scale problems. In a second approach, as described in [4], a QP problem is solved online. To reduce the number of computations, warm-starting and early termination of the QP problem is proposed. In warm-starting, the initialization of the QP problem is done using the predictions made in the previous step. This reduces the computational cost only if the new QP is similar to the previous one. The early termination significantly reduces the computations, but on the other hand it may lead to state equation violations. In [5], an iterative scheme based on fast gradient methods, is used to obtain the solution of an optimization problem arising in predictive control with input constraints only. This scheme allows one to compute an a priori upper bound on the number of iterations for a given accuracy.

The linear system of equations to be solved at each iteration of an IPM becomes ill-conditioned as the solution is approached. The linear system is either solved by direct or iterative methods. Iterative methods are often preferred, because their hardware implementation is efficient and a trade-off between speed and accuracy can be achieved [6]. To improve the ill-conditioning of the linear system, we propose an approximation of the ill-conditioned linear system where the resulting approximate linear system is well-conditioned. An upper bound on the introduced error, which gets reduced as IPM iterations are increased, is determined analytically. The minimal residual method (MINRES) is used here to solve the approximate well-conditioned system. It is shown that the approximate well-conditioned system can be solved much faster with fewer iterations than with the original ill-conditioned system. The rate of convergence of the approximate system is further enhanced with the development of two new preconditioners. The preconditioned system used here has a lower computational cost compared to the one used in [7]. Results obtained from numerical simulations indicate that the computational complexity of our proposed method scales quadratically with the number of states and linearly with the horizon length.

Notation: For a matrix, $A > 0$ (≥ 0) means that A is positive (semi-positive) definite and for a vector, $x < 0\,(\leq 0)$ means that each element of x is negative (non-positive). The infinity norm of a matrix $A \in \mathbb{R}^{m\times n}$ is defined as $\|A\|_\infty := \max_{1\leq i\leq m} \sum_{j=1}^{n} |a_{ij}|$. $A(1:r, 1:r)$ denotes the submatrix of A containing the first r rows and r columns. For a set X, card(X) denotes the number of elements in X.

A. Shahzad and G. Constantinides are with the Department of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, U.K. a.shahzad07@imperial.ac.uk, g.constantinides@imperial.ac.uk

E. Kerrigan is with the Department of Aeronautics and Department of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, U.K. e.kerrigan@imperial.ac.uk

The symbol $\otimes$ represents the Kronecker product.

II. PROBLEM DESCRIPTION

Consider a discrete-time linear time-invariant dynamic system of the form

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

where $x_i \in \mathbb{R}^n$ is the state vector and $u_i \in \mathbb{R}^m$ is the input vector at the i-th time instant. Let $x = x_0 \in \mathbb{R}^n$ be the measurement or estimate of the state at the current time instant. The objective is to find, over a finite horizon of length N, a sequence of optimal control inputs $u_0, \ldots, u_{N-1}$ subject to equality constraints (1) and the inequality constraints

$$\begin{aligned} J_i x_i + E_i u_i &\leq d_i, \quad i = 0, \ldots, N-1, && \text{(2a)} \\ J_N x_N &\leq d_N, && \text{(2b)} \end{aligned}$$

while minimizing the quadratic objective

$$x_N{}^T P x_N + \sum_{i=0}^{N-1} \left(x_i{}^T Q x_i + u_i^T R u_i + 2 x_i^T M u_i \right) \tag{3}$$

with $R > 0$, $Q - MR^{-1}M^T \geq 0$ and $P \geq 0$, $J_i, J_N \in \mathbb{R}^{l \times n}, E_i \in \mathbb{R}^{l \times m}, Q, P \in \mathbb{R}^{n \times n}, R \in \mathbb{R}^{m \times m}, M \in \mathbb{R}^{n \times m}$.

To solve the above problem the vector of decision variables is defined as

$$\theta := \begin{bmatrix} x_0^T & u_0^T & x_1^T & u_1^T & \cdots & u_{N-1}^T & x_N^T \end{bmatrix}^T. \tag{4}$$

The above problem can be converted to a QP problem, which can be written in the form

$$\min_{\theta} \frac{1}{2} \theta^T H \theta \quad \text{subject to} \quad F\theta = f(x), \quad G\theta \leq d, \tag{5}$$

where $\theta \in \mathbb{R}^{n_d}, H \in \mathbb{R}^{n_d \times n_d}, F \in \mathbb{R}^{n_e \times n_d}, G \in \mathbb{R}^{n_i \times n_d}$ with $n_d := (n+m)N+n$, $n_e := n(N+1)$ and $n_i := l(N+1)$ is the number of inequality constraints. The matrices H, F, G and the vectors f, d are defined in the Appendix. To simplify notation in subsequent sections the (x) is omitted from vector $f(x)$.

To solve the QP problem two approaches are used, namely active set methods [8] and interior point methods (IPMs) [2], [9]. We focus on IPMs because they have polynomial computational complexity while active set methods have exponential complexity in the worst case [9].

III. INTERIOR POINT METHODS

In this section we review the ideas in interior point methods [9]. The Karush-Kuhn-Tucker (KKT) conditions of (5) are

$$\begin{aligned} H\theta + F^T\nu + G^T\phi &= 0, && \text{(6a)} \\ F\theta - f &= 0, && \text{(6b)} \\ G\theta - d + s &= 0, && \text{(6c)} \\ \Phi S \mathbf{1}_{n_i} &= 0, \quad \phi, s \geq 0, && \text{(6d)} \end{aligned}$$

where $\nu \in \mathbb{R}^{n_e}$ and $\phi \in \mathbb{R}^{n_i}$ are called dual variables, $s \in \mathbb{R}^{n_i}$ is a vector of slack variables, $\mathbf{1}_{n_i} \in \mathbb{R}^{n_i}$ is a vector of ones and Φ, S are diagonal matrices defined by $\Phi := \text{diag}(\phi), \quad S := \text{diag}(s)$. This also indicates that we have n_e equality constraints and n_i inequality constraints.

In an IPM the optimal solution is obtained by solving the nonlinear optimality conditions (6). The classical algorithm to solve such equations is Newton's method. This is an iterative method in which at each iteration k, the solution of a linear system of the following form is required to find the search direction $\Delta\mathbf{x}^k$:

$$\underbrace{\begin{bmatrix} H & F^T & G^T \\ F & 0 & 0 \\ G & 0 & -W^k \end{bmatrix}}_{\mathcal{A}^k} \underbrace{\begin{bmatrix} \Delta\theta^k \\ \Delta\nu^k \\ \Delta\phi^k \end{bmatrix}}_{\Delta\mathbf{x}^k} = \underbrace{\begin{bmatrix} r_H^k \\ r_F^k \\ r_L^k \end{bmatrix}}_{b^k}, \tag{7}$$

where $\mathcal{A}^k$ and b^k are defined in Appendix I.

An IPM in which an initial guess of θ satisfies the equality and inequality constraints defined in (5) and $\phi, s > 0$ is called a *feasible* IPM. An IPM for which an initial guess of θ satisfies only $\phi, s > 0$ is called an *infeasible* IPM. An IPM that uses an *inexact* method to solve the linear system of equations (7) is called an inexact interior point method. An infeasible and inexact IPM is described in Algorithm 1 for a QP problem. This is an extension of the infeasible-path-following IPM (Algorithm IPF) of [9, p. 110], which was designed for a linear programming problem. In path following methods, we follow a parametrized path to the solution of (6), where Newton's method is used to target a succession of points along the path.

Algorithm 1 Inexact Interior Point Method (IIPM)

Input:
- H, F, G, f, d
- Initial guess $\theta^0, \nu^0, \phi^0 > 0, s^0 > 0, \zeta \geq 1, \sigma \in (0,1)$
- Tolerance $\epsilon > 0$

Output: Optimal θ.

Algorithm:

1: Set $k = 0$ and compute $\mu^0 := \frac{(\phi^0)^T s^0}{n_i}$, $e_{tol}^0 := \|b^0\|_\infty$.
2: **while** $\mu^k > \epsilon$ and $e_{tol}^k > \epsilon$ **do**
3: Solve (7) or (13) for $\Delta\mathbf{x}^k$ or its approximation with an iterative method with relative residual tolerance

$$\eta^k := \max\left\{ \min\left\{ \frac{1}{(k+1)^\zeta}, \|b^k\|_\infty \right\}, \epsilon \right\}. \tag{8}$$

4: Choose α^k as the largest value in $[0,1]$ such that conditions (6.5) and (6.6) in [9] are satisfied.
5: $(\mathbf{x}^{k+1}, s^{k+1}) := (\mathbf{x}^k, s^k) + \alpha^k(\Delta\mathbf{x}^k, \Delta s^k)$.
6: Compute $\mu^{k+1} := \frac{(\phi^{k+1})^T s^{k+1}}{n_i}$, $e_{tol}^{k+1} := \|b^{k+1}\|_\infty$.
7: Increment k by 1.
8: **end while**

In the IIPM the linear system (7) is solved with less accuracy in the initial iterations, as depicted by the decreasing function η^k. Therefore, direct methods, which solve the linear system exactly, are not applicable here. We focus on an *inexact* IPM, because the total number of floating-point operations needed to find the solution to (5) can be

reduced significantly by using an inexact method to solve (7), compared to using an exact method [10]. Note that the final output of an inexact IPM is within the same tolerance of the solution to (5) as the output of an exact IPM. The only part where approximate solutions are obtained are in the early iterations.

IV. PRECONDITIONED ITERATIVE METHODS

The system matrix $\mathcal{A}^k$ appearing in (7) is symmetric, but indefinite, hence the minimum residual (MINRES) iterative method [11] can be used to compute the solution for $\Delta\mathbf{x}^k$. The solution of (7), where most of the computations are done, is a key part of Algorithm 1. It is well-known that the condition number of the matrix $\mathcal{A}^k$ increases as k increases and iterative methods do not perform well with ill-conditioned systems. Therefore, in the predictive control literature, the main focus in solving such linear systems is on direct methods. In this paper we propose an approximation of (7) such that the resulting approximate linear system becomes well-conditioned. It is shown in the next section that this modified linear system can be solved with fewer MINRES iterations. The rate of convergence of the modified system is further enhanced with the development of two new preconditioners.

A. Approximation of the linear system (7)

In this section, we discuss the mechanisms of ill-conditioning, and present an approximation of (7) with a better-conditioned matrix. Note that if $s_i^k \to 0$ as $k \to \infty$, then (6c) indicates that the i-th inequality constraint is active. Let $\mathcal{N} := \{1, 2, \ldots, n_i\}$ and define a δ-active set $\mathcal{N}_A^k(\delta)$, depending upon the parameter $\delta > 0$, as

$$\mathcal{N}_A^k(\delta) := \{i \in \mathcal{N} \mid 0 < w_i^k < \delta\} \tag{9}$$

where $w_i^k := \frac{s_i^k}{\phi_i^k}$ and a δ-inactive set as

$$\mathcal{N}_I^k(\delta) := \mathcal{N} \setminus \mathcal{N}_A^k(\delta). \tag{10}$$

We propose that a sufficiently large value of δ is chosen such that all inequality constraints become δ-active for $k = 0$. Suppose $n_a^k := \text{card}(\mathcal{N}_A^k(\delta))$. In practice, for larger values of k and small δ, there is a small number of active constraints, i.e. $n_a^k \ll n_i$. As the iteration number k of Algorithm 1 increases, the values corresponding to the $\delta-$inactive constraints of the diagonal matrix W^k become very large. Note that if $s_i^k \nrightarrow 0$ as $k \to \infty$ then $\phi_i^k \to 0$, because $\mu^k = \frac{(\phi^k)^T s^k}{n_i} \to 0$ as $k \to \infty$. Therefore $w_i^k \to \infty$ as $k \to \infty$ for the constraints that are inactive at the solution. As a result, the condition number of $\mathcal{A}^k$ becomes very large.

Let us permute the matrix $\mathcal{A}^k$ according to the $\delta-$active and $\delta-$inactive constraints as

$$\left[\begin{array}{cccc} H & F^T & (G_1^k)^T & (G_2^k)^T \\ F^T & 0 & 0 & 0 \\ G_1^k & 0 & -W_1^k & 0 \\ \hline G_2^k & 0 & 0 & -W_2^k \end{array}\right] \left[\begin{array}{c} \Delta\theta^k \\ \Delta\nu^k \\ \Delta\phi_1^k \\ \hline \Delta\phi_2^k \end{array}\right] = \left[\begin{array}{c} r_H^k \\ r_F^k \\ r_{L_1}^k \\ \hline r_{L_2}^k \end{array}\right], \tag{11}$$

where $\left[\begin{array}{c} G_1^k \\ G_2^k \end{array}\right] = U^k G$, $\left[\begin{array}{c} \Delta\phi_1^k \\ \Delta\phi_2^k \end{array}\right] = U^k \Delta\phi^k$, $\left[\begin{array}{c} r_{L_1}^k \\ r_{L_2}^k \end{array}\right] = U^k r_L^k$, $U^k \in \mathbb{R}^{n_i \times n_i}$ is a permutation matrix, $W_1^k \in \mathbb{R}^{n_a^k \times n_a^k}$, $W_2^k \in \mathbb{R}^{(n_i - n_a^k)\times(n_i - n_a^k)}$, $\|W_1^k\|_\infty < \delta$ and $\|(W_2^k)^{-1}\|_\infty \leq \delta^{-1}$. For simplicity, we write (11) as

$$\left[\begin{array}{cc} A_1^k & (A_2^k)^T \\ A_2^k & -W_2^k \end{array}\right] \left[\begin{array}{c} \Delta\mathbf{z}_1^k \\ \Delta\mathbf{z}_2^k \end{array}\right] = \left[\begin{array}{c} b_1^k \\ b_2^k \end{array}\right], \tag{12}$$

where A_1^k, A_2^k represent the corresponding block matrices, and $\Delta\mathbf{z}_1^k, \Delta\mathbf{z}_2^k, b_1^k, b_2^k$ are the corresponding vectors in (11).

We propose that rather than solving (7) or (12), which becomes ill-conditioned at the later iterations of the IPM, we can find an approximate solution of (12) by solving

$$A_1^k \Delta\hat{\mathbf{z}}_1 = b_1^k, \tag{13a}$$

$$\Delta\hat{\mathbf{z}}_2 = (W_2^k)^{-1}(A_2^k \Delta\hat{\mathbf{z}}_1 - b_2^k). \tag{13b}$$

We claim that (13a) is better conditioned compared to (12). This increases the rate of convergence of the iterative method and also enables the use of low precision floating point arithmetic. To justify our claim we first recall Gershgorin's theorem and the interlacing property of symmetric matrices.

Theorem 1: [12, §8.1.2] (Gershgorin) Suppose $\mathbf{A} \in \mathbb{R}^{p\times p}$ is a symmetric matrix, then

$$\lambda(\mathbf{A}) \subseteq \bigcup_{i=1}^{p} [a_{ii} - r_i,\ a_{ii} + r_i]$$

where a_{ii} is the ii-th element of $\mathbf{A}$ and $r_i = \sum_{j=1, j\neq i}^{p} |a_{ij}|$.

Theorem 2: [12, §8.1.2] (Interlacing Property) Suppose $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a symmetric matrix with $\lambda_p(\mathbf{A}) \leq \lambda_{p-1}(\mathbf{A}) \cdots \leq \lambda_2(\mathbf{A}) \leq \lambda_1(\mathbf{A})$ and $\mathbf{A}_r := \mathbf{A}(1:r, 1:r)$, then

$$\begin{aligned} \lambda_{r+1}(\mathbf{A}_{r+1}) \leq \lambda_r(\mathbf{A}_r) \leq \lambda_r(\mathbf{A}_{r+1}) \leq \cdots \\ \leq \lambda_2(\mathbf{A}_{r+1}) \leq \lambda_1(\mathbf{A}_r) \leq \lambda_1(\mathbf{A}_{r+1}) \end{aligned}$$

for $r = 1, 2, \ldots, p-1$.

Proposition 1: Let $\rho(A_1^k)$ be the spectral radius of A_1^k, then

$$\rho(A_1^k) \leq \max\{h_{ii} + r_1, \delta + \|G\|_\infty\}$$

where $r_1 := \|H\|_\infty + \|F\|_\infty + \|G\|_\infty$ and h_{ii} is the ii-th element of H.

Proof: Follows from Gershgorin's theorem. ■

Proposition 2: Let $\lambda_i(A_1^k)$ be an eigenvalue of A_1^k, then

$$\lambda_{i+1}(\mathcal{A}_{n_c^k+1}^k) \leq \lambda_i(A_1^k) \leq \lambda_i(\mathcal{A}_{n_c^k+1}^k).$$

where $n_c^k := n_d + n_e + n_a^k$.

Proof: Follows from the interlacing property. ■

It is observed from numerical experiments that ill-conditioning of $\mathcal{A}^k$ for large k arises mainly due to the maximum eigenvalue of $\mathcal{A}^k$. Proposition 1 shows that the maximum eigenvalue of A_1^k is bounded and its upper limit depends upon the infinity norm of H, F, G and δ. However, it does not depend on the number of IPM iterations k. Proposition 2 indicates that the minimum eigenvalue of A_1^k is bounded below by the minimum eigenvalue of $\mathcal{A}^k$. This does not give any bound on the absolute minimum eigenvalue of

A_1^k, but shows that the spectrum of eigenvalues of A_1^k lies within the spectrum of eigenvalues of $\mathcal{A}^k$. Tighter clustering of eigenvalues results in fewer iterations of the MINRES method, because the number of iterations depends upon the distribution of eigenvalues [13, pp. 119-120]. Propositions 1 and 2 together justify why (13a) is better conditioned compared to (12), and numerical results in Section VI confirm this.

The approximation of (12) by (13) reduces the computational cost. However, it introduces an error in the solution of (12). The upper bound in the solution error is estimated in the following result:

Proposition 3: Let $e^k := \Delta\hat{\mathbf{z}}_1^k - \Delta\mathbf{z}_1^k$ be the error in the solution of the first part of (12), then

$$\|e^k\|_\infty \leq \delta^{-1}\left(c_2^2\|b_1^k\|_\infty + c_2\|b_2^k\|_\infty\right) + O(\delta^{-2}),$$

where $c_2 := \frac{\kappa_\infty(A_1^k)\|G\|_\infty}{c_3}, c_3 := \left\| \begin{array}{cc} H & F^T \\ F & 0 \end{array} \right\|_\infty$, and $\kappa_\infty(A_1^k)$ is the condition number of A_1^k.

Proof: See Appendix II. ∎

Note that $\|b^k\|_\infty < \epsilon$ at the termination of Algorithm 1. So by choosing appropriate values of ϵ, δ, and k, it is possible to make e^k arbitrarily small. This indicates that at the final stages of the IPM the error e^k would converge to zero.

To reduce the number of unknowns in (13), we prefer to solve

$$\left(U_1^k V(U_1^k)^T + W_1^k\right)\Delta\hat{\phi}_1^k = U_1^k\bar{G}\bar{H}^{-1}r_{\bar{H}}^k - r_{L_1}^k, \quad (14)$$

where $V := \bar{G}\bar{H}^{-1}(\bar{G})^T, \bar{H} := \left[\begin{array}{cc} H & F^T \\ F & 0 \end{array} \right], U^k := \left[\begin{array}{c} U_1^k \\ U_2^k \end{array} \right], \bar{G} := [G\ 0]$ and $U_1^k \in \mathbb{R}^{n_a^k \times n_i}$. The remaining part of the solution of (13) is calculated as

$$\left[\begin{array}{c} \Delta\hat{\theta}^k \\ \Delta\hat{\nu}^k \end{array} \right] = \left[\begin{array}{cc} H & F^T \\ F^T & 0 \end{array} \right]^{-1} \left[\begin{array}{c} r_H^k - (G_1^k)^T\Delta\hat{\phi}_1^k \\ r_F^k \end{array} \right]. \quad (15)$$

From (13b),

$$\Delta\hat{\phi}_2^k = (W_2^k)^{-1}(G_2^k\Delta\hat{\theta}_k - r_{L_2}^k). \quad (16)$$

B. Preconditioners

A preconditioner is a matrix that approximates the coefficient matrix of a linear system and its factorization is inexpensive [13, p. 119]. In the next two subsections, we propose two preconditioners to solve the linear system (14) and discuss their properties.

1) A Diagonal Preconditioner: Consider the following diagonal preconditioner

$$\mathcal{P}_1^k := W_1^k. \quad (17)$$

Lemma 1: The rank of $V \in \mathbb{R}^{n_i \times n_i}$ is less than or equal to $\min\{n_d, n_i\}$.

Proof: Let

$$\bar{H}^{-1} := \left[\begin{array}{cc} H_1 & H_2^T \\ H_2 & H_3 \end{array} \right],$$

where $H_1 \in \mathbb{R}^{n_d \times n_d}, H_2 \in \mathbb{R}^{n_e \times n_d}$ and $H_3 \in \mathbb{R}^{n_e \times n_e}$. Since

$$V = \left[\begin{array}{cc} G & 0 \end{array} \right] \left[\begin{array}{cc} H_1 & H_2^T \\ H_2 & H_3 \end{array} \right] \left[\begin{array}{c} G^T \\ 0 \end{array} \right] = GH_1G^T$$

and $\text{rank}(GH_1G^T) \leq \min(\text{rank}(G), \text{rank}(H_1))$; it follows that $\text{rank}(V) \leq \min\{n_d, n_i\}$. ∎

Theorem 3: [7] Let $\widetilde{\mathbf{A}} = \mathbf{A} + \Delta$ where $\widetilde{\mathbf{A}}$ is a symmetric matrix of size $p \times p$ with $\mathbf{A} > 0$ and $\text{rank}(\Delta) = r < p$. The MINRES method with preconditioner $\mathbf{A}$, when solving a linear system with coefficient matrix $\widetilde{\mathbf{A}}$, will terminate in at most $r + 1$ iterations.

Proposition 4: A preconditioned MINRES method with preconditioner $\mathcal{P}_1^k$ will terminate in at most $\min\{n_a^k, n_d\}$ iterations when solving the linear system (14).

Proof: We know that

$$\text{rank}(U_1^k V(U_1^k)^T) \leq \min\{\text{rank}(U_1^k), \text{rank}(V)\}.$$

Since $\text{rank}(U_1^k) = n_a^k$ and from Lemma 1, $\text{rank}(V) = \min\{n_a^k, n_d\}$, therefore

$$\text{rank}(U_1^k V(U_1^k)^T) \leq \min\{n_a^k, n_d\}. \quad (18)$$

From Theorem 3, it can be concluded that a preconditioned MINRES method with preconditioner $\mathcal{P}_1^k$ will terminate in at most $\min\{n_a^k, n_d\}$ iterations when solving the linear system (14). ∎

In most predictive control problems, we have $n_i > n_d$. In this scenario the preconditioner $\mathcal{P}_1^k$ is quite effective, because it reduces the maximum number of iterations to n_d, so we recommend the use of preconditioner $\mathcal{P}_1^k$ if $n_a^k > n_d$.

2) A Block Diagonal Preconditioner: If $n_a^k \leq n_d$, we propose a block diagonal preconditioner of the form

$$\mathcal{P}_2^k := U_1^k\bar{V}(U_1^k)^T + W_1^k, \quad (19)$$

where $\bar{V}$ is a block diagonal approximation of V, which is calculated by the following semidefinite program

$$\min_{\bar{V}} \|\bar{V} - V\|_\infty \quad \text{subject to} \quad \bar{V} \geq 0. \quad (20)$$

Our focus on the block diagonal preconditioner is mainly due to two reasons. Firstly, the matrices H and F have a special structure that can be exploited to convert $\bar{H}$ into a banded matrix with bandwidth $(2n+m)$. If we pick all block matrices of size $(2n + m)$ from the permuted $\bar{H}$ to form a block diagonal approximation of permuted $\bar{H}$, then after pre-multiplying it by $\bar{G}$ and post-multiplying with $(\bar{G})^T$ it still remains banded. This could be one way of forming $\bar{V}$. However, we prefer solving the semidefinite program (20), since this guarantees that $\mathcal{P}_2^k > 0$, because $W_1^k > 0$. Positive-definiteness is a requirement for a preconditioner in the MINRES method [13, pp. 119-120]. Secondly, all subsystems associated to each block diagonal of $\mathcal{P}_2^k$ would be independent and can be implemented in parallel if a parallel computer architecture is available.

C. Implementation Scheme

The linear system (14) is solved with a preconditioned MINRES (PMINRES) method. The preconditioner $\mathcal{P}_1^k$ is selected if $n_a^k > n_d$, otherwise preconditioner $\mathcal{P}_2^k$ is chosen. There could be two implementation schemes for solving (14). One way could be to precompute V off-line and then $U_1^k V (U_1^k)^T$ can be formed by just picking the rows and columns of V according to U_1^k. Secondly, the LU-factorization of the permuted $\bar{H}$ can be computed off-line and $\bar{H}^{-1}v$ for any vector v can be computed by using backward and forward substitution. In the second approach we only need to store the sparse LU-factorization matrices, while in the first approach we need to store the whole dense matrix V. We prefer the second approach in this paper, since it exploits the sparsity and requires less memory.

V. Computational Complexity Analysis

The computational complexity of each method is measured in flops. The cost of an IPM per iteration of the Riccati recursion method [2] is given in Table I. The cost of matrix-vector multiplication in the MINRES method is $(2n+m)^2 N + (n+m)n_a^k$. The total cost of solving (14) with the MINRES method is given in Table I, where N^k_{MINRES} denotes the number of MINRES iterations required at the k^{th} iteration of an IPM. It is observed in numerical simulations that N^k_{MINRES} is roughly constant. Note that only high-order terms are mentioned in Table I. However, all terms, including the lower order ones, are taken into account in the numerical results presented in Section VI.

TABLE I
Computational cost of an IPM per iteration.

Method	Flops
Riccati Recursion [2]	$(3n^3 + 6n^2m + 3nm^2 + \frac{1}{3}m^3)N + l(n^2 + 2nm + m^2)(N+1)$
MINRES	$(4(2n+m)^2 N + 4(n+m)n_a^k)N^k_{\text{MINRES}}$

VI. Numerical Results

We present a case study to evaluate the performance of our proposed method. Consider a system of $q := \frac{n}{2}$ equal masses connected by springs and to walls at the ends. The mass of each block is 1 kg and the spring constant of each spring is taken as 1 N/m. There is no damping. There are m actuators connected to the first m masses and each can exert a maximum force of ± 0.5. The displacements of the masses are restricted to ± 4. This continuous-time state-space system is transformed into a discrete-time system using a sample time of 0.5 sec. The objective is to regulate the displacements with the given constraints on displacements and control inputs. The regulator tuning matrices are taken as $R = I, M = 0$, and $Q = [I_p \ 0]^T [I_p \ 0]$, where $p = n/2$ and the states have been ordered such that the first p states describe the displacements. The matrix P satisfies the associated discrete-time algebraic Riccati equation. A number of simulations is carried out with initial conditions $x = 3.5[1 \ 1 \ 0 \ \cdots \ 0]^T, \theta^0 = \mathbf{1}_{n_d}, \nu^0 = \mathbf{1}_{n_e}, \phi^0 = s^0 = \mathbf{1}_{n_i}, \epsilon = 10^{-3}, \delta = 1.5, \zeta = 6$. The values of δ, ϕ^0 and s^0 are selected such that all inequality constraints become δ-active at $k = 0$.

Fig. 1(a) indicates that the rate of convergence of the MINRES method when solving the original linear system (7) is very slow, but much faster for the modified system (14). This rate of convergence is further enhanced using a preconditioned MINRES (PMINRES) method. It is also observed in some cases that the unpreconditioned MINRES fails to converge when solving (7) and the solution never reaches the desired accuracy, due to the high condition number of matrix $\mathcal{A}^k$. This shows that iterative methods without a preconditioner is not a good option for IIPMs. Fig. 1(b) indicates that the condition number of the original system increases while the condition number of the modified system remains almost the same as the iteration k of Algorithm 1 increases. Fig. 1(c) indicates that the number of δ-active inequality constraints decreases as the iteration k of Algorithm 1 increases. Fig. 1(d) indicates that the normalized error in the solution of the modified system (13a) gets reduced as the IPM iteration number increases.

To see the growth of computational cost with the number of states n, simulations are carried out with fixed inputs m and horizon length N. Fig. 2(a) shows that the cost of PMINRES method is less than the Riccati recursion method. The plots of $n^2/2000$ and $n^3/6500$ are also plotted for comparison. Note that the PMINRES method roughly scales with $\mathcal{O}(n^2)$. Secondly, keeping n and m fixed, simulations are carried out for varying N and results are plotted in Fig. 2(b), which indicates that PMINRES roughly scales with $\mathcal{O}(N)$.

VII. Conclusions

The ill-conditioning of the linear system in the final iterations of an interior point method is well-known, and iterative methods are more sensitive to numerical errors for ill-conditioned linear systems. Through the modification of the ill-conditioned system to a well-conditioned one, an efficient iterative scheme for solving a QP problem was developed in this work. We provide an upper bound on the error of the approximation used, which decreases as the IPM iteration increases. It is shown that the modified, well-conditioned linear system can be solved using the preconditioned minimal residual (PMINRES) method with fewer iterations, compared to the original ill-conditioned system. Results obtained from numerical simulations indicate that the computational complexity of our proposed method scales quadratically with the number of states and linearly with the horizon length. We expect that our scheme is applicable generally to other applications of interior point methods.

Acknowledgment

This work was funded by the EPSRC under grant number EP/G031576/1.

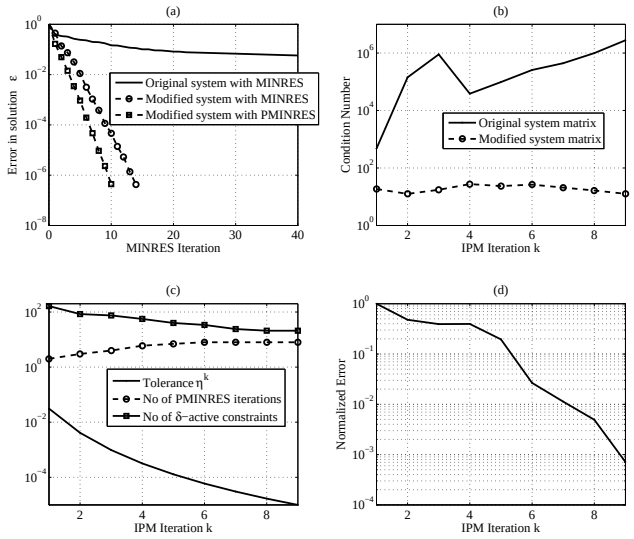

Fig. 1. The simulation results for (a), (b), (c) and (d) are computed with $n = 10, m = 2$ and $N = 20$ (a) Comparison of rate of convergence of solving the original linear system (7) and modified linear system (13) with MINRES and PMINRES for $k = 3$. (b) Comparison of condition numbers of original matrix $\mathcal{A}^k$ and modified matrix A_1^k. (c) Decay of relative residual tolerance η^k, decay of number of δ-active inequality constraints, and number of PMINRES iteration required for tolerance η^k with main loop iteration k. (d) Decay of normalized error $\frac{\|e^k\|_\infty}{\|e^0\|_\infty}$ in the solution of the modified system (13a).

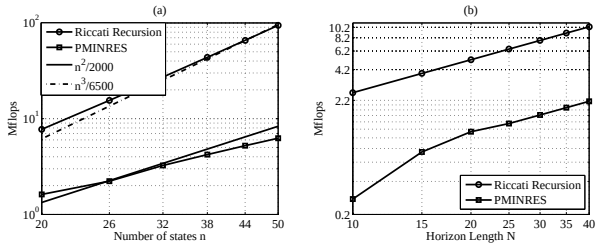

Fig. 2. Mflops represents the total number of flops required to solve (5) by Algorithm 1 and Riccati recursion method. (a) Growth of number of flops with number of states n, where $m = 1$ and $N = 30$. (b) Growth of number of flops with horizon length N, where $n = 20$ and $m = 1$.

APPENDIX I
DEFINITION OF MATRICES AND VECTORS

$$H := \begin{bmatrix} I_N \otimes \bar{Q} & 0 \\ 0 & P \end{bmatrix} \text{ where } \bar{Q} := \begin{bmatrix} Q & M \\ M^T & R \end{bmatrix}$$

$$F := \begin{bmatrix} -I_n & 0 \\ 0 & I_N \otimes [0 \quad -I_n] \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ I_N \otimes [A \quad B] & 0 \end{bmatrix}$$

$$\text{If } J_i = J, E_i = E \ \forall\, i \ \Rightarrow\ G := \begin{bmatrix} I_N \otimes [J \quad E] & 0 \\ 0 & J_N \end{bmatrix}$$

$$\text{If } d_i = \bar{d} \ \forall\, i \ \Rightarrow\ d := I_N \otimes \bar{d}, \quad f(x) := \begin{bmatrix} -x^T & 0 \end{bmatrix}^T$$

$$\mathbf{x}^k := \begin{bmatrix} (\theta^k)^T & (\nu^k)^T & (\phi^k)^T \end{bmatrix}^T$$

$$\begin{aligned} W^k &:= (\Phi^k)^{-1} S^k \\ r_B^k &:= r_H^k - G^T (S^k)^{-1} \left(\Phi^k r_G^k + r_S^k\right) \\ \Delta\phi^k &:= (S^k)^{-1} \left(\Phi^k r_G^k + r_S^k + \Phi^k G \Delta\theta^k\right) \\ \Delta s^k &:= (\Phi^k)^{-1} \left(r_S^k - S^k \Delta\phi^k\right) \\ r_H^k &:= -\left(H\theta^k + F^T \nu^k + G^T \phi^k\right), \\ r_F^k &:= -\left(F\theta^k - f\right) \\ r_L^k &:= r_G^k - (\Phi^k)^{-1} (r_S^k + \sigma\mu^k \mathbf{1}_{n_i}) \\ r_G^k &:= G\theta^k - d + s^k, \ r_S^k := -\Phi^k S^k \mathbf{1}_{n_i} \end{aligned}$$

APPENDIX II
PROOF OF PROPOSITION 3

From (12),

$$\Delta\mathbf{z}_1^k = \left(A_1^k + A_2^k (W_2^k)^{-1} (A_2^k)^T\right)^{-1} (b_1^k + A_2^k (W_2^k)^{-1} b_2^k).$$

For small ϵ and $\mathbf{A}, \mathbf{X} \in \mathbb{R}^{p \times p}$, the Sherman-Morrison-Woodbury formula [12, §2.1.3] can be written in the form

$$(\mathbf{A} + \epsilon\mathbf{X})^{-1} = \mathbf{A}^{-1} - \epsilon\mathbf{A}^{-1}\mathbf{X}\mathbf{A}^{-1} + O(\epsilon^2). \tag{21}$$

Let $\bar{W}_2^k := W_2^k/\delta$, then from (21):

$$\begin{aligned} \Delta\mathbf{z}_1^k = (A_1^k)^{-1} b_1^k - \delta^{-1} (A_1^k)^{-1} A_2^k (\bar{W}_2^k)^{-1} \\ \left((A_2^k)^T (A_1^k)^{-1} b_1^k - b_2^k\right) + O(\delta^{-2}). \end{aligned} \tag{22}$$

The infinity norm of the error $e^k := \Delta\hat{\mathbf{z}}_1^k - \Delta\mathbf{z}_1^k$ can then be calculated as

$$\|e^k\|_\infty \leq \delta^{-1} \left(c_2^2 \|b_1^k\|_\infty + c_2 \|b_2^k\|_\infty\right) + O(\delta^{-2}),$$

where c_2 and c_3 are defined as in the statement of Proposition 3.

REFERENCES

[1] J. M. Maciejowski, *Predictive Control with Constraints*. Prentice Hall, 2002.

[2] C. V. Rao, S. J. Wright, and J. B. Rawlings, "Application of interior point methods to model predictive control," *Journal of Optimization Theory and Applications*, vol. 99, no. 3, pp. 723–757, 1998.

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

[4] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 267–278, 2010.

[5] S. Richter, C. N. Jones, and M. Morari, "Real-time input-constrained MPC using fast gradient methods," in *Proc. IEEE Conference on Decision and Control*, Shanghai, China, December 2009.

[6] G. A. Constantinides, "Tutorial paper: Parallel architectures for model predictive control," in *Proc. European Control Conference*, Budapest, Hungary, August 2009, pp. 138–143.

[7] A. Shahzad, E. C. Kerrigan, and G. A. Constantinides, "Preconditioners for inexact interior point methods for predictive control," in *Proc. American Control Conference*, Baltimore, Maryland, USA, June 2010.

[8] R. R. Fletcher, *Practical Methods of Optimization*. Wiley, 2nd edition 1987.

[9] S. J. Wright, *Primal-Dual Interior Point Methods*. SIAM Publications, 1997.

[10] R. S. Dembo and T. Steihaug, "Truncated Newton algorithms for large-scale unconstrained optimization," *Mathematical Programming*, vol. 26, pp. 190–212, 1983.

[11] C. C. Paige and M. A. Saunders, "Solution of sparse indefinite systems of linear equations," *SIAM J. Numerical Analysis*, vol. 12, no. 4, pp. 617–629, 1975.

[12] G. A. Golub and C. F. V. Loan, *Matrix Computations*. The John Hopkins University Press, 1996.

[13] A. Greenbaum, *Iterative Methods for Solving Linear Systems*. SIAM, 1997.