USING A MODIFIED PREDICTOR-CORRECTOR ALGORITHM FOR MODEL PREDICTIVE CONTROL

A. G. Wills* and W. P. Heath**

* Department of Electrical and Computer Engineering, University of Newcastle, Australia.
** Centre for Integrated Dynamics And Control (CIDAC), University of Newcastle, Australia.

Abstract: A modified predictor-corrector algorithm is presented. This algorithm obtains a pre-specified point on the primal-dual central-path. It is shown to be suitable for a recently proposed class of receding horizon control laws which include a recentred barrier in the cost function. The significance of these controllers is that hard constraints are replaced by penalty type soft constraints, which has the effect of backing-off the control action near the constraint boundary. The class of controllers is parameterised by a positive scalar with an associated unconstrained minimisation problem. The solution to this problem for a fixed parameter value is given by the corresponding point on the primal-dual central-path.

Keywords: Receding horizon control, recentred barrier function, interior-point methods, predictor-corrector algorithm, self-scaled cones.

1. INTRODUCTION

Model predictive control (MPC) requires the solution of an optimisation problem at each time interval. This determines a sequence of control moves that steer the system state to some desired set-point. An MPC strategy is often chosen for its constraint handling capabilities. Recently, interior-point methods have been proposed for solving the associated constrained optimisation problem (Rao et al., 1998; Wright, 1997; Hansson, 2000).

Wills and Heath (2002) have proposed a class of receding horizon control laws which are based on quite traditional interior-point methods. In particular a recentred barrier function is used to regulate points to lie inside the constraint set. The significance of these controllers is that hard constraints are replaced by penalty type soft constraints using a recentred barrier function. This has the effect of backing-off the control action near the constraint boundary. The extent to which this backing-off occurs is determined by a weighting parameter \( \eta \). For each value of \( \eta \) there is an associated convex unconstrained minimisation problem; the basic approach is to fix \( \eta \), to say \( \eta_p \), and solve the corresponding problem at each time step.

In the case where \( \eta_p \) is sufficiently large the optimisation problem may be solved using simple Newton iterations. However, when \( \eta_p \) is chosen to be small, this approach may have poor numerical properties. This phenomenon is common to barrier methods for small values of weighting parameter where the Hessian matrix becomes ill-conditioned (Wright, 1992).

In this paper we present a predictor-corrector algorithm that is suitable for any choice of parameter value. The algorithm terminates when the iterates become sufficiently close to a prespecified point on the primal-dual central-path. It is intended for (but not restricted to) the above mentioned class of receding horizon controllers.

The paper structure is as follows. In section 2 we provide some notation and definitions relevant to the above mentioned class of controllers. In section 3 we provide a brief overview of standard conic quadratic form. In order to take advantage of primal-dual interior-point methods, we
reformulate the ‘limiting case’ minimisation problem into conic form (Nesterov and Nemirovskii, 1994). We consider the plant to be represented by a linear time-invariant discrete-time state-space model with both linear and convex quadratic constraints. Furthermore, we make the usual assumption that the finite receding horizon cost function can be expressed as a convex quadratic function of future inputs for a given system state. In this case, we may represent the ‘limiting case’ optimisation problem in standard conic quadratic form. In section 4 we provide an algorithm which is primarily based on §7 of (Nesterov and Todd, 1998). In particular, we are interested in stopping at the point on the primal-dual central-path which corresponds to the parameter value \( \eta_p \). The resulting point may then be used to obtain a solution to the original minimisation problem. In section 5 we confirm that the solution obtained in section 4 indeed corresponds to the original problem posed by Wills and Heath (2002). In section 6 we provide a simple example simulation to help illustrate these ideas. Section 7 concludes the paper.

2. RECENTRED BARRIER MPC

When designing a receding horizon controller, it is customary to represent physical and imposed constraints as a closed convex subset of a finite dimensional real vector space (Mayne et al., 2000). We may regulate points to lie inside this feasible domain by including a barrier function with a fixed weighting parameter \( \eta_p > 0 \) (see e.g. Fiacco and McCormick, 1968). In this section we provide a summary of relevant definitions and notation for this approach.

For a finite dimensional real vector space \( Z = \mathbb{R}^n \), let \( G \) denote the constraint set defined as,

\[
G := \{ z \in Z : f_i(z) \leq 0 \text{ for } i = 1, \ldots, M \},
\]

where \( z \) typically represents a stacked vector of future input signals and each \( f_i(z) \) is a convex quadratic function, i.e. \( f_i(z) \) may be expressed as \( f_i(z) = z^T A_i z + b_i^T z + c_i \) with \( A_i \) positive semi-definite and symmetric for \( i = 1, \ldots, M \). Furthermore, let \( G^0 \) denote the interior of \( G \). It is assumed throughout this paper that \( G^0 \neq \emptyset \), and \( G \) is bounded. Let \( L(z) \) be the standard logarithmic barrier function,

\[
L(z) = \begin{cases} 
-\sum_{i=1}^{M} \ln(-f_i(z)) & \text{if } z \in G^0 \\
\infty & \text{otherwise.}
\end{cases}
\]

For a point \( z_d \in G^0 \), let \( L_{zd}(z) \) denote the recentred barrier function defined as,

\[
L_{zd}(z) = L(z) + b_{zd}^T z, \quad b_{zd} = \sum_{i=1}^{M} \frac{1}{f_i(z_d)} \nabla f_i(z_d).
\]

The class of receding horizon optimisation problems may be expressed as,

\[
(RH_{\eta}) : \min_{z \in Z} \{ \tilde{f}_0(z) + \eta L_{zd}(z) \},
\]

where \( \tilde{f}_0(z) = z^T \Delta_0 z + \tilde{b}_0^T z + \tilde{c}_0 \) represents the receding horizon cost function and \( \eta \in (0, \infty) \). The approach is to fix the value of \( \eta \), to say \( \eta = \eta_p > 0 \), and solve the corresponding unconstrained minimisation problem \((RH_{\eta_p})\). The associated receding horizon control law is then constructed in the standard manner by selecting the first control move. This process is repeated at each time interval.

By construction, the minimum of the recentred barrier occurs at \( z_d \in G^0 \). This is an essential characteristic for the class of receding horizon controllers which are constructed from \((RH_{\eta_p})\). It means precisely that if the closed-loop system is stable and \( z_d \) is the desired steady-state set-point, then the system will indeed converge to \( z_d \). This property is not guaranteed with a more general barrier (for example the logarithmic barrier) - even with integral action.

For a fixed weighting parameter \( \eta = \eta_p > 0 \) and a point \( z_d \in G^0 \), we find it convenient to express \((RH_{\eta_p})\) as an instance of the following class of optimisation problems,

\[
(\mathcal{RC}_\mu) : \min_{z \in Z} \{ f_0(z) + \mu L(z) \},
\]

where \( \mu \in (0, \infty) \) and \( f_0(z) \) is given by \( f_0(z) = z^T \tilde{A}_0 z + \tilde{b}_0^T z + \tilde{c}_0 \), with \( \tilde{b}_0 = b_0 + \eta_p b_{zd}, \tilde{A}_0 = \Delta_0 \) and \( \tilde{c}_0 = c_0 \). Clearly, \((\mathcal{RC}_\mu)\) and \((RH_{\eta_p})\) are equivalent when \( \mu = \eta_p \).

It is well known that in the limit as \( \mu \rightarrow 0 \), the solution to \((\mathcal{RC}_\mu)\) tends to the solution of the following problem (see e.g. Fiacco and McCormick, 1968),

\[
(C) : \min_{z \in G} f_0(z) \text{ s.t. } z \in G
\]

In the sequel we may refer to \((C)\) as the ‘limiting-case’ optimisation problem.

3. CONIC FORM

In order to take advantage of recent developments in interior-point machinery, it is first necessary to translate \((C)\) into standard conic quadratic form. Nesterov and Nemirovskii (1994) defined the primal conic form as,

\[
(\mathcal{P}) : \min_{c, x} \{ c^T x \} \text{ s.t. } Ax = b, \ x \in K
\]

where \( K \) is a pointed convex cone with non-empty interior. In particular, for the case of a single convex quadratic constraint, the cone is given by the \( n \)-dimensional second order cone defined as

\[
K_n^2 := \{ x \in \mathbb{R}^n : \| x \|_2^2 \leq x_{2:n}^T x_{2:n} \},
\]

where \( x_{2:n} \) refers to the \((n-1)\) vector whose \( i \)th element is \( x_{i+1} \) for \( i = 1, \ldots, n-1 \). Moreover, for the case of a single linear inequality constraint, the cone is given by the non-negative half-axis denoted \( \mathbb{R}_+ \). In what follows we will consider a combination of \( K_n^2 \) and \( \mathbb{R}_+ \) to construct \( K \).

The following, which is broadly based on §6.2 of
expressed as
and the solution set of the following problem,
how to convert (\text{Nesterov and Nemirovskii}, 1994), demonstrates
it is first necessary to construct an affine mapping
where \(A\) and \(B\) are augmented versions of \(A_i\) and \(b_i\) that cater for the extra variable \(t\).

In order to express (CT) in standard conic form, it is first necessary to construct an affine mapping
for each constraint; such a mapping will be denoted by \(B_i\). Without loss of generality, we assume that the first \(p\) constraints are convex quadratic and the remaining \(q\) constraints are linear. Using an appropriate decomposition, let \(A_i = D_i^T D_i\), where \(D_i\) is an \(r_i \times (n + 1)\) matrix and \(r_i\) is the rank of \(A_i\). Note that since \(A_i\) is non-negative definite and symmetric, such a decomposition always exists. For the first \(p\) constraints, we have the following relation,
\[ g_i(v) \leq 0 \iff B_i(v) \in K_{r_i+2}^2, \tag{10} \]
where the affine mapping \(B_i(v)\) is given by,
\[ B_i : V \rightarrow \mathbb{R}^{r_i+2}, B_i(v) = B_i v + d_i \tag{11} \]
with
\[ B_i := \begin{bmatrix} -b_i^T \\ 2D_i \\ b_i^T \end{bmatrix} \quad \text{and} \quad d_i := \begin{bmatrix} 1 - c_i \\ 0 \\ 1 + c_i \end{bmatrix} \tag{12} \]
This relationship may be demonstrated as follows. From the definition of \(K_{r_i+2}^2\) and \(B_i(v)\) we have that \(B_i(v) \in K_{r_i+2}^2\) if and only if,
\[ 4(v^T D_i^T D_i v) + (1 + b_i^T v + c_i)^2 \leq (1 - b_i^T v - c_i)^2. \tag{13} \]
Since
\[ (1 + b_i^T v + c_i)^2 - (1 - b_i^T v - c_i)^2 = 4b_i^T v + 4c_i, \tag{14} \]
then (13) becomes
\[ v^T A_i v + b_i^T v + c_i \leq 0. \tag{15} \]
For the \(q\) remaining linear constraints, we can define the corresponding \(B_i(v)\) as follows,
\[ B_i : V \rightarrow \mathbb{R}, B_i(v) = -b_i^T v - c_i \tag{16} \]
Clearly in this case, \(g_i(v) \leq 0 \iff B_i(v) \in \mathbb{R}_+\). Let \(r_i = -1\) for the case of linear constraints.

Let \(K_i\) denote the \(i\)th cone for \(i = 0, \ldots, M\). We may define the cone \(K\) and vector space \(X\) as
\[ K := \prod_{i=0}^{M} K_i \quad \text{and} \quad X := \prod_{i=0}^{M} \mathbb{R}^{r_i+2}. \tag{17} \]
Furthermore, with a slight abuse of notation we can define the mapping \(\mathcal{B}(v)\) as
\[ \mathcal{B} : V \rightarrow X, \mathcal{B}(v) = [B_0^T(v), \ldots, B_M^T(v)]^T. \tag{18} \]
Let \(r = \sum_{i=0}^{M}(r_i + 2)\). We may write \(\mathcal{B}(v)\) in a more convenient form as \(\mathcal{B}(v) = Bv + d\), where \(B\) is the \(r \times (n + 1)\) matrix formed by stacking the \(M + 1\) matrices \(B_i\) (where \(B_i = -b_i^T\) in the linear case), and \(d\) is the vector formed by stacking the \(M + 1\) vectors \(d_i\) (where \(d_i = -c_i\) in the linear case). Therefore, \(v \in G_t\) if and only if \(Bv + d \in K\).

Since \(B\) is an affine mapping, we may represent the image of \(B\) as an affine hyperplane in \(X\). We make the usual assumption that \(B\) has full row-rank. For the case where \(A_0\) is positive definite and symmetric (this is common to receding-horizon control), we construct the affine hyperplane as follows: form the Cholesky factorisation of \(A_0\), i.e. let \(A_0 = C_0^T C_0\), where \(C_0\) is an upper triangular matrix. Let \(C := [0 C_0]\) be the matrix constructed by augmenting a vector of zeros with \(C_0\). It follows from the definition of \(g_0(v)\) that \(A_0 = C_0^T C_0\) and \(b_0 = [1, 0, \ldots, 0]^T\). We may partition \(B\) as follows,
\[ B = \begin{bmatrix} U \\ \tilde{B} \end{bmatrix}, \quad U = \begin{bmatrix} \tilde{b}_0^T \\ 2C_0^T \\ \vdots \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} -b_0^T \\ B_1 \\ \vdots \\ B_M \end{bmatrix} \tag{19} \]
Note that \(U\) is a full-rank upper triangular matrix. Let \(A\) be the matrix given by \(A = [BU^{-1}\ I]\) and let \(b = Ad\). Then \(x\) lies in the image of \(B\) if and only if \(Ax = b\), which is exactly the form required in (P). It remains to find \(c\) such that \((c, x) = t\), in which case (CT) would be equivalent to (P). Form the QR factorisation of \(B\), i.e. let \(B = QR\) where \(Q\) is an \(r \times (n + 1)\) matrix with orthonormal columns and \(R\) is an \((n + 1) \times (n + 1)\) upper triangular matrix. Then \(c\) may be given by \(c := QR^{-1}c'\), where \(c' = [1, 0, \ldots, 0]^T\).

4. ALGORITHM

In this section we present a primal-dual algorithm which terminates when the iterates become sufficiently close to a pre-specified point on the central path of (P). In particular, we are interested in the point that corresponds to the parameter value \(\eta_p\). In this case, we may use the solution generated by the algorithm to obtain a solution to (RH) for the chosen parameter value \(\eta_p\). The algorithm is based on a primal-dual predictor-corrector method for self-scaled cones introduced by Nesterov and Todd (1998).

We define the dual optimisation problem in the standard manner as,
\[ (D) : \max_{y \in \mathbb{R}^n} \langle b, y \rangle \quad \text{s.t.} \quad A^T y + s = c, \quad s \in K^* \tag{20} \]
where \(Y = \mathbb{R}^{r-n+1}\) and \(K^*\) is the cone dual to \(K\), which in this paper is \(K\) itself. We may define the combined primal-dual minimisation problem as,
\[ (PD) : \min \{ \langle c, x \rangle - \langle b, y \rangle \} \quad \text{s.t.} \quad Ax = b \quad A^T y + s = c \quad x \in K, \quad s \in K^* \tag{21} \]
We may define a barrier for the cone $K$ as follows: let $x(i) \in \mathbb{R}^{r+2}$ denote the $i$'th ‘block vector’ of $x \in X$ for $i = 0, \ldots, M$. Then $F(x)$ is given by
\[
F(x) = -\sum_{i=0}^{p-1} \ln x^{T}(i)^{T}Q_{i}x(i) - \sum_{i=p}^{M} \ln x(i),
\]
where $Q_{i} := \text{diag}(1, -1, \ldots, -1)$. Let $F_{s}(s)$ denote the dual barrier defined as,
\[
F_{s}(s) = F(s) + p \ln(4) - \nu
\]
where $\nu = 2p + q$. For feasible $(s, x, y)$ we have that $(s, x) = (c, x) - (b, y)$. Then we may define a perturbed problem for $(PD)$ as,
\[
(PD_{p}): \min \left\{ \frac{1}{p} \langle s, x \rangle + F(x) + F_{s}(s) \right\}
\]
s.t $Ax = b, A^{T}y + s = c$
Let $(x(\rho), s(\rho), y(\rho))$ denote the solution to $(PD_{p})$. Then the collection of points $\{(x(\rho), s(\rho), y(\rho)) : \rho \in (0, \infty)\}$ defines the primal-dual central path for $(PD)$. Furthermore, for any $\rho > 0$, the following relation holds (Nesterov and Todd, 1998),
\[
s(\rho) = -\rho F'(x(\rho)).
\]
To measure the ‘closeness’ of a primal-dual pair $(x, s, y)$ to a point on the central-path, we use the functional proximity measure defined in (Nesterov and Todd, 1998), where the particular point on the central-path corresponds to the parameter value given by $\rho(x, s) := \frac{1}{\rho}(s, x)$.

For the case of linear and convex quadratic constraints, the functional proximity measure may be expressed as,
\[
\gamma_{F}(x, s) := F(x) + F(s) + \nu \ln(\rho(x, s)) + p \ln(4)
\]
We have by definition that $K$ is a self-scaled cone (Nesterov and Todd, 1994), and $F(x)$ is a self-scaled barrier for $K$. A remarkable property associated with self-scaled barriers is the existence of a unique scaling point $\omega \in \text{int} K$ (where $\text{int} K$ refers to the interior of $K$) such that $F^{\omega}(x) = s$ for $x \in \text{int} K$ and $s \in \text{int} K^{*}$. For the case of linear and convex quadratic constraints, the point $\omega$ may be easily computed, e.g. see (Andersen et al., 2000).

Solving the following set of linear equations is integral to many feasible-start interior-point algorithms,
\[
F^{\omega}(x)\Delta x + d_{s} = \zeta s + \xi F'(x)
\]
\[
Ad_{x} = 0
\]
\[
d_{s} + A^{T}d_{y} = 0
\]
where $\zeta$ and $\xi$ are variables that change according to the particular algorithm (and possibly at different stages in the algorithm).

Let $\mathcal{F}(\beta)$ denote the set of all strictly feasible primal-dual points $(x, s, y)$ such that $\gamma_{F}(x, s) \leq \beta$ (see figure 1), compute the following:

**Algorithm:** Choose $\epsilon_{1} > 0$, $\epsilon_{2} > 0$, $\Delta > 0$ and $\beta$ such that $0 < \beta < 1 - \ln 2$. Given a positive weighting parameter $\eta_{p}$ and a strictly feasible initial primal-dual pair $(x_{0}, s_{0}, y_{0})$ such that $\gamma_{F}(x_{0}, s_{0}) < \beta$, we have the following.

While $|p_{k}^{+} - \eta_{p}| \geq \epsilon_{1}$ and $(x_{k+1}, s_{k+1}, y_{k+1}) \in \mathcal{F}(\epsilon_{2})$ then iterate the following:

1. Let $\rho_{k} = \rho(x_{k}, s_{k})$ and $\epsilon_{2} = \rho_{k} - \eta_{p}$.
   - (a) If $\epsilon_{k} > \epsilon_{1}$ then obtain $(d_{x_{k}}, d_{s_{k}}, d_{y_{k}})$ by solving (27) with $\zeta = 1$ and $\xi = 0$.
   - (b) If $\epsilon_{k} < \epsilon_{1}$ then obtain $(d_{x_{k}}, d_{s_{k}}, d_{y_{k}})$ by solving (27) with $\zeta = -1$ and $\xi = 0$.
   - Let $\alpha_{k} = 1 + \frac{\epsilon_{2}}{\rho_{k}}$.
2. Form the ‘predictor’ point,
   \[
x_{k}^{+}(\alpha) = x_{k} - \alpha d_{x_{k}},
   \]
   \[
s_{k}^{+}(\alpha) = s_{k} - \alpha d_{s_{k}},
   \]
   \[
y_{k}^{+}(\alpha) = y_{k} - \alpha d_{y_{k}},
   \]
   If $(x_{k}^{+}(\alpha_{k}^{+}), s_{k}^{+}(\alpha_{k}^{+}), y_{k}^{+}(\alpha_{k}^{+}))$ is strictly feasible and $\gamma_{F}(x_{k}^{+}(\alpha_{k}), s_{k}^{+}(\alpha_{k}), y_{k}^{+}(\alpha_{k})) \leq \beta + \Delta$, then let $\alpha_{k} = \alpha_{k}^{+}$ and proceed to step (3). Otherwise, find $\alpha_{k} \in (0, \alpha_{k}^{+})$ such that $(x_{k}^{+}(\alpha_{k}), s_{k}^{+}(\alpha_{k}), y_{k}^{+}(\alpha_{k}))$ is strictly feasible and $\gamma_{F}(x_{k}^{+}(\alpha_{k}), s_{k}^{+}(\alpha_{k}), y_{k}^{+}(\alpha_{k})) = \beta + \Delta$.
3. Compute the new point $(x_{k+1}, s_{k+1}, y_{k+1})$ by using the Newton method defined in §5.2 of Nesterov and Todd (1998) starting from $(x_{k}^{+}(\alpha_{k}), s_{k}^{+}(\alpha_{k}), y_{k}^{+}(\alpha_{k}))$. Note that the central direction can be found by solving (27) with $\zeta = 1$ and $\xi = \rho_{k}$, where $\rho_{k} = \rho(x_{k}^{+}(\alpha_{k}), s_{k}^{+}(\alpha_{k})).$ If $|p_{k}^{+} - \eta_{p}| > \epsilon_{1}$ then terminate as soon as a point in $\mathcal{F}(\epsilon_{2})$ is found. Otherwise, terminate as soon as a point in $\mathcal{F}(\epsilon_{2})$ is found.

**end.**

For the case of $\epsilon_{k} > \epsilon_{1}$ (Zone 2 in figure 1), then the above algorithm takes predictor-corrector steps in the usual fashion until the iterates become sufficiently close to the desired point on the central-path.

![Fig. 1. This figure shows a conceptual view of the central path. Zone 1 and 2 represent the regions where $\rho_{k} - \eta_{p} < -\epsilon_{1}$ and $\rho_{k} - \eta_{p} > \epsilon_{1}$ respectively. The dashed line passing through the central path represents the level set where $(s, x) = \eta_{p}$. Furthermore, the dotted lines running along-side the central-path represent the neighbourhood such that $\gamma_{F}(x, s) \leq \beta.$](image-url)
The mechanism for ensuring that \( \rho(x_{k+1}, s_{k+1}) \) does not go beyond \( \eta_p \) comes from the identity (see §5.1 of Nesterov and Todd, 1998):

\[
\langle s_k^*: (\alpha), x_k^* (\alpha) \rangle = (1 - \alpha) \langle s_k, x_k \rangle. 
\]

(28)

Since we are trying to find \((x_{k+1}, s_{k+1}, y_{k+1})\) such that \(\rho(x_{k+1}, s_{k+1}) = \eta_p\), then we may set the right hand side of (28) to \(\eta_p\) and solve for \(\alpha\) (this is equivalent to \(\alpha_k^*\) from step (1a)).

In the event that an initial point has \(c_0 \leq -\epsilon_1\) (Zone 1 in figure 1), then the algorithm takes negative predictor steps and normal corrector steps until the iterates become sufficiently close to the desired point on the central-path. In this case the first equation of (27) becomes,

\[
\nabla^T (\omega_k) dx_k + ds_k = -s_k, 
\]

and since \(s = [\nabla^T (\omega)]^{-1} x\) then the following relation holds,

\[
\langle s_k, dx_k \rangle + \langle ds_k, x_k \rangle = -\langle s_k, x_k \rangle, 
\]

(30)

and therefore, \(\langle s_k^* (\alpha), x_k^* (\alpha) \rangle = (1 - \alpha) \langle s_k, x_k \rangle\) (since \(\langle ds_k, dx_k \rangle = 0\)). Moreover, for \(\alpha_k^*\) defined in step (1b) we have that \(\rho(x_{k+1}, s_{k+1}) = \eta_p\).

In both cases, we are trying to find a primal-dual pair such that \(\rho(x_{k+1}, s_{k+1}) - \eta_p \leq \epsilon_1\). Once this is achieved, the algorithm enters a final corrector stage to ensure the resulting pair is close to the central-path (as determined by \(\epsilon_2\)).

To initialise the algorithm we propose the following approach: obtain a strictly feasible primal point (this is usually a trivial task for receding horizon control since the constraints are commonly related to physical phenomenon). Use the initialisation method described in §9 of (Nesterov and Todd, 1994) with a fixed weighting parameter \(\tau > 0\), where \(\tau\) should be chosen large enough that convergence is rapid. At each iteration of the method, construct \(s\) via the following projection: let \(y = [AA^*]^{-1} A(c + \tau \nabla F(x))\) and then let \(s = c - A^* y\). We have from equation (25) that \(s \rightarrow -\tau F'(x)\) as \(x \rightarrow x(\tau)\), and therefore a strictly feasible primal-dual pair may be obtained in this manner.

Remark 4.1. In the case where constraints are of a static nature, i.e. do not change with time, then it suffices to compute, off-line, a point close to the analytic centre of \(G\) and use primal steps and the above projection to find a strictly feasible primal-dual pair. This is the approach taken for the simulations presented in section 6.

Remark 4.2. When implementing an interior-point algorithm for receding-horizon control, care should be taken to exploit any matrix sparseness and problem structure. For the algorithm presented above, it is necessary to reformulate the problem into comatic quadratic form at each time step. For the case of a linear plant model and convex quadratic objective function, it is often the case that only \(b_0\) changes between time intervals. In this case, we may update the QR factorisation of \(B\) via a single Givens rotation, and therefore \(c\) may be obtained ‘cheaply’ (as long as we store \(Q\) and \(R\)). Furthermore, \(\tilde{A}\) can also be obtained ‘cheaply’ by updating the first row of \(U^{-1}\) (which corresponds to \(b_0\)) via forward substitution.

5. CENTRAL PATH EQUIVALENCE

In this section we demonstrate that the solution to \((R_H)\), for some fixed parameter \(\eta = \eta_p > 0\), coincides with the point on the central path of \((P)\) (and therefore \((PD)\)) corresponding to the same parameter value \(\eta_p\).

Let \((CT\sigma)\) denote a perturbed problem for \((CT)\) given by,

\[
\text{(CT}_\sigma\text{)} : \min_{v \in V} \{ t + \sigma H(v) \}
\]

(31)

where \(H(v)\) is the standard logarithmic barrier for the constraint set \(G_t\). Let \(v(\sigma) : \sigma \in (0, \infty)\) defines the central path for \((CT)\). First, we show that the solutions to \((R_H)\) and \((CT)\) coincide when \(\sigma = \mu\) (independent of the last variable \(t\)). The optimality condition for \((R_H)\) with \(z \in G_0^0\) can be expressed as

\[
\nabla f_0(z) + \mu \sum_{i=1}^M \frac{1}{f_i(z)} \nabla f_i(z) = 0
\]

(32)

Similarly, the optimality condition for \((CT\sigma)\) with \(v \in G_0^0\) is

\[
\begin{bmatrix} 0 \\ 1 \end{bmatrix} + \sigma \sum_{i=0}^M \frac{1}{g_i(v)} \nabla g_i(v) = 0
\]

(33)

Using the definition of \(v\) and \(g_i(\cdot)\), we may express (33) as

\[
\begin{bmatrix} \sigma' \nabla f_0(z) + \sigma \sum_{i=1}^M \frac{1}{f_i(z)} \nabla f_i(z) \\ - \sigma' - 1 \end{bmatrix} = 0
\]

(34)

where \(\sigma' = \frac{\sigma}{1 - g_0(v)}\). From the last equation of (34) we have that \(t - f_0(z) = \sigma\), and therefore (33) is equivalent to (32) when \(\sigma = \mu\) (independent of the last variable \(t\)). Furthermore, in the case where \(\sigma = \eta_p\), the solutions to \((R_H)\) and \((CT_{\eta_p})\) coincide (independent of the last variable \(t\)).

It remains to verify that \(v(\eta_p)\) coincides with the point on the central path of \((P)\) corresponding to the same parameter value \(\eta_p\). Let \((P_\gamma)\) denote a perturbed problem for \((P)\) defined as,

\[
(P_\gamma) : \min_{x \in X} \{ c(x) + \gamma F(x) \} \text{ s.t. } Ax = b
\]

(35)

Let \(x(\gamma)\) denote the solution to \((P_\gamma)\); then the set of points \(\{x(\gamma) : \gamma \in (0, \infty)\}\) defines the primal central path for \((P)\). Indeed, from the definitions of \(v\), \(g_i(v)\), \(B(v)\), \(H(v)\) and \(F(x)\) we have that \(x(\eta_p) = B(v(\eta_p))\).
6. SIMULATION

In this section we provide a simple example that simultaneously illustrates the effect of different choices of weighting parameter $\eta_p$ and verifies the algorithm in section 4. We are using the recentred barrier function receding horizon controller as defined in (Wills and Heath, 2002), and the algorithm described in section 4 to solve the associated optimisation problem at each time interval. The plant model is given by,

$$x_{k+1} = \begin{bmatrix} -0.3 & -0.8 \\ 0.5 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \quad y_k = [0.5 \ 0] x_k$$

(36)

The input signal is constrained to lie within simple bounds given by $-1 \leq u_k \leq 0.4$. For a prediction horizon of $N = 10$, we applied a step-disturbance to the system and the results are illustrated in figures 2 and 3. Note that the results here are equivalent to those found in (Wills and Heath, 2002) where Newton steps are used.

![Comparison of different values of weighting parameter $\eta_p$.](image)

**Fig. 2.** Comparison of output signals for different values of $\eta_p$.

![Comparison of different values of weighting parameter $\eta_p$.](image)

**Fig. 3.** Comparison of input signals for different values of $\eta_p$. For $\eta_p$ small, the input signal may travel very close to the constraint boundary. But as $\eta_p$ increases, the control system penalises points near the constraint boundary more heavily, hence the cautious control trajectory observed for large values of $\eta_p$.

7. CONCLUSION

We have presented a modified primal-dual predictor-corrector algorithm for the case of convex quadratic cost with linear and convex quadratic constraints. The objective of this algorithm is to find a point sufficiently close to a particular point on the primal-dual central path. The point of interest corresponds to the fixed weighting parameter $\eta_p > 0$ which characterises the receding horizon controller instance. In the case where only linear constraints are present, then a similarly modified predictor-corrector method for mixed linear complementarity problems may be more appropriate. Furthermore, the above algorithm can be extended naturally to include the cone of positive semi-definite matrices.

**Acknowledgements.** The Authors would like to acknowledge the helpful comments made by Katrina Lau and Minyue Fu.

References


