Robust Model Predictive Control with Imperfect Information

Arthur Richards
Department of Aerospace Engineering
University of Bristol
Bristol, BS8 1TR, UK
Email: arthur.richards@bristol.ac.uk

Jonathan How
Aerospace Control Laboratory
Massachusetts Institute of Technology
Cambridge MA 02139
Email: jhow@mit.edu

Abstract— This paper presents two extensions to robust Model Predictive Control (MPC) involving imperfect information. Previous work developed a form of MPC guaranteeing feasibility and constraint satisfaction given an unknown but bounded disturbance and perfect state information. In the first extension, this controller is modified to account for an unknown but bounded state estimation error. As an example, a simple estimator is proposed and analyzed to provide the necessary error bounds. Furthermore, it is shown that delayed state information can be handled using the same method. These analyses depend on knowledge of bounds on the measurement and disturbance uncertainties. The second extension provides a method of estimating these bounds using available data, providing an adaptive form of the controller for cases where the error levels are poorly known *a priori*.

I. Introduction

This paper presents two extensions to robust Model Predictive Control (MPC). The first combines robust full-state feedback MPC [1] with an estimator, as shown in Fig. 1(a), to provide output feedback control, accounting for inaccuracy and time delay in state information. The second extension, outlined in Fig. 1(b), is applicable when the level of uncertainty is not well known. It is a form of adaptive control, using data history to estimate the uncertainty set and modifying the MPC controller to suit.

The vast majority of research on MPC assumes full state information. Recently, attention has turned to formal analysis of the output feedback case. We adopt the approach of Bertsekas and Rhodes [2] of transforming the problem to an equivalent form with perfect information. A simple finite-memory estimator is shown to provide state information with suitably-bounded errors. Further, we provide a formal analysis of the method proposed in [8] for handling time delay, showing that it can be considered in the same framework as estimation error. Closest to this work is that of Lee and Kouvaritakis [5], who derive invariant sets for the estimation error and then employ their own robust MPC method. Other approaches involve variations on a separation principle for nonlinear MPC [3], [4], including the application of receding horizon estimation [6].

Analytical consideration of sensing noise and delay requires knowledge of bounds on the process and measurement uncertainty. While judgement and experience can often fulfill this need, exact levels of uncertainty are rarely
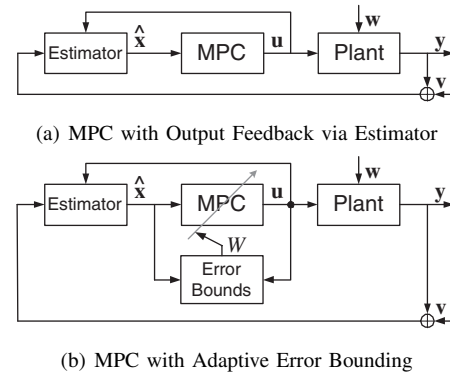

(a) MPC with Output Feedback via Estimator

(b) MPC with Adaptive Error Bounding

Fig.1: System Block Diagrams. In Fig. 1(a), a constant error set $\mathcal{W}$ is embedded in MPC to account for uncertainties $\mathbf{w}$ and $\mathbf{v}$. In Fig. 1(b), the set $\mathcal{W}$ is estimated from online measurements.

known in practice. The second development in this paper provides an alternative approach, deriving error bounds online from measurements of the system performance, resulting in an adaptive scheme. Since the adaptation is based on estimating the level of noise, the method is most closely related to adaptive tuning for Kalman filtering [7], [9], using the innovation as measurement data to update estimated levels of process and sensing noise. The adaptive MPC scheme proposed in this paper is outlined in Fig. 1(b). The prediction error can be directly measured and used to update estimates of the bounding sets that determine the constraint tightening. Adaptive filtering methods usually seek the variance of the signal [7], but the MPC robustness depends on bounds on the prediction error. Therefore, a novel algorithm is developed to predict bounds on future signals from past measurements. Since the bounds are estimated from imperfect information, there is a probability that they might be lower than the true bound and that the problem could become infeasible. The bounding algorithm is set up to allow the designer to choose the probability of this occurrence as a parameter in the adaptation scheme.

Section II reviews robust MPC with perfect information. Section III shows how problems with imperfect information, including estimation error in Section III-A and time delay in Section III-E, can be accommodated by the controller from Section II. Section IV develops the adaptive method

for estimating error bounds online.

II. REVIEW OF ROBUST MPC

This section gives a brief review of the robust MPC formulation from Ref. [1]. This controller guarantees robust feasibility and constraint satisfaction given full state information. The new results in this paper arise from the ability to convert other problems to this form and apply the same result. Assume the system is linear-time invariant and modeled by the discrete-time state-space equation

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{w}(k) \tag{1}$$

where the disturbance $\mathbf{w}$ is unknown but bounded, obeying

$$\forall k \; \mathbf{w}(k) \in \mathcal{W} \tag{2}$$

The objective is to satisfy output constraints

$$\forall k \; \mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) \tag{3a}$$
$$\mathbf{y}(k) \in \mathcal{Y} \tag{3b}$$

while minimizing the cost function

$$J = \sum_{k=0}^{\infty} \ell\left(\mathbf{u}(k), \mathbf{x}(k)\right) \tag{4}$$

where $\ell(\cdot)$ is some stage cost function. The matrices $\mathbf{C}$ and $\mathbf{D}$, the set $\mathcal{Y}$ and the function $\ell(\cdot)$ are all chosen by the designer as part of the problem statement. Now define $\mathsf{P}(\mathbf{x}(k), \mathcal{Y}, \mathcal{W})$, the MPC optimization starting from state $\mathbf{x}(k)$ with constraint set $\mathcal{Y}$ and disturbance set $\mathcal{W}$

$$J^*(\mathbf{x}(k), \mathcal{Y}, \mathcal{W}) = \min_{\mathbf{u}, \mathbf{x}, \mathbf{y}} \sum_{j=0}^{N} \ell\left(\mathbf{u}(k+j|k), \mathbf{x}(k+j|k)\right)$$

$$\text{subject to } \forall j \in \{0 \ldots N\}$$
$$\mathbf{x}(k+j+1|k) = \mathbf{A}\mathbf{x}(k+j|k) + \mathbf{B}\mathbf{u}(k+j|k) \tag{5a}$$
$$\mathbf{y}(k+j|k) = \mathbf{C}\mathbf{x}(k+j|k) + \mathbf{D}\mathbf{u}(k+j|k) \tag{5b}$$
$$\mathbf{x}(k|k) = \mathbf{x}(k) \tag{5c}$$
$$\mathbf{x}(k+N+1|k) \in \mathcal{X}_F \tag{5d}$$
$$\mathbf{y}(k+j|k) \in \mathcal{Y}(j) \tag{5e}$$

where the constraint sets for the plan $\mathcal{Y}(j)$ are tightened for robustness using the following recursions

$$\forall j \in \{0 \ldots N-1\}$$
$$\mathcal{Y}(0) = \mathcal{Y}$$
$$\mathcal{Y}(j+1) = \mathcal{Y}(j) \sim (\mathbf{C} + \mathbf{D}\mathbf{K}(j)) \mathbf{L}(j)\mathcal{W}$$
$$\mathbf{L}(0) = \mathbf{I}$$
$$\mathbf{L}(j+1) = (\mathbf{A} + \mathbf{B}\mathbf{K}(j)) \mathbf{L}(j)$$

for some candidate control $\mathbf{u}(j) = \mathbf{K}(j)\mathbf{x}(j)$, chosen by the designer, that stabilizes the system. The operator "$\sim$" denotes the Pontryagin difference [10] defined as

$$\mathcal{A} \sim \mathcal{B} = \{\mathbf{a} \mid \mathbf{a} + \mathbf{b} \in \mathcal{A} \; \forall \mathbf{b} \in \mathcal{B}\} \tag{6}$$

with the resulting useful property

$$\mathbf{c} \in \mathcal{A} \sim \mathcal{B} \Rightarrow \mathbf{c} + \mathbf{b} \in \mathcal{A} \; \forall \mathbf{b} \in \mathcal{B} \tag{7}$$

The terminal constraint set is given by

$$\mathcal{X}_F = \mathcal{R} \sim \mathbf{L}(N)\mathcal{W} \tag{8}$$

where $\mathcal{R}$ is a robust control invariant admissible set *i.e.* there exists a control law $\kappa(\mathbf{x})$ satisfying the following

$$\forall \mathbf{x} \in \mathcal{R}, \; \mathbf{A}\mathbf{x} + \mathbf{B}\kappa(\mathbf{x}) + \mathbf{L}(N)\mathbf{w} \in \mathcal{R}, \; \forall \mathbf{w} \in \mathcal{W} \tag{9a}$$
$$\mathbf{C}\mathbf{x} + \mathbf{D}\kappa(\mathbf{x}) \in \mathcal{Y}(N) \tag{9b}$$

This type of terminal constraint is a common feature of MPC formulations, included to guarantee stability.

Algorithm 1 (Robustly Feasible MPC)
1) Solve problem $\mathsf{P}(\mathbf{x}(k), \mathcal{Y}, \mathcal{W})$
2) Apply control $\mathbf{u}(k) = \mathbf{u}^*(k|k)$ from the optimal sequence
3) Increment k. Go to Step 1

Theorem 1 (Robust Feasibility) If $\mathsf{P}(\mathbf{x}(0), \mathcal{Y}, \mathcal{W})$ has a feasible solution then the system (1) controlled by Algorithm 1 and subjected to disturbances obeying (2) robustly satisfies the constraints (3) and all subsequent optimizations $\mathsf{P}(\mathbf{x}(k), \mathcal{Y}, \mathcal{W})$ are feasible.

This theorem is quoted from [1]. The proof is omitted here for brevity.

III. IMPERFECT STATE INFORMATION

A. MPC with Estimation Error

This section shows the MPC formulation of Section II is modified to operate with an inaccurate state estimate. The method transforms the problem to consider the dynamics of the estimate [2], which is perfectly known, by definition. Let the state estimate $\hat{\mathbf{x}}$ be the sum of the true state $\mathbf{x}$ and an error $\mathbf{e}$

$$\hat{\mathbf{x}}(k) = \mathbf{x}(k) + \mathbf{e}(k) \tag{10a}$$
$$\hat{\mathbf{x}}(k+1) = \mathbf{x}(k+1) + \mathbf{e}(k+1) \tag{10b}$$

where the error is unknown but bounded

$$\mathbf{e}(k) \in \mathcal{E} \; \forall k \tag{11}$$

Section III-B describes a simple estimator that is suitable for this purpose. Substituting (10a) and (10b) into (1) gives the dynamics of the estimate

$$\hat{\mathbf{x}}(k+1) = \mathbf{A}\hat{\mathbf{x}}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{w}(k) + \mathbf{e}(k+1) - \mathbf{A}\mathbf{e}(k) \tag{12}$$

This is the same form as the dynamics of the true state (1), but driven by an "effective process noise"

$$\hat{\mathbf{w}}(k) = \mathbf{w}(k) + \mathbf{e}(k+1) - \mathbf{A}\mathbf{e}(k) \tag{13}$$

Define the set $\hat{\mathcal{W}}$ such that

$$\hat{\mathbf{w}}(k) \in \hat{\mathcal{W}} \; \forall k \tag{14}$$

A suitable set can be found using the disturbance (2) and estimate (11) error sets in a vector (Minkowski) summation

$$\hat{\mathcal{W}} = \mathcal{W} \oplus \mathcal{E} \oplus (-\mathbf{A})\mathcal{E} \tag{15}$$

This is a suitable bound for (14) and will suffice for robust feasibility, but it ignores possible correlations between the estimation error and disturbance signals. This will be examined further in Section III-C.

To account for the discrepancy between the true state and the estimate, it is necessary to modify the constraint sets, tightening them slightly using a Pontryagin difference (6)

$$\hat{\mathcal{Y}} = \mathcal{Y} \sim (-\mathbf{C})\mathcal{E} \qquad (16)$$

Then the output feedback control algorithm involves changing only the uncertainty and constraint sets in the optimization and including the estimation step.

Algorithm 2 (Robustly Feasible Output Feedback MPC)

1) Take measurement and form estimate $\hat{\mathbf{x}}(k)$
2) Solve problem $\mathsf{P}(\hat{\mathbf{x}}(k), \hat{\mathcal{Y}}, \hat{\mathcal{W}})$
3) Apply control $\mathbf{u}(k) = \mathbf{u}^*(k|k)$ from the optimal sequence
4) Increment k. Go to Step 1

Theorem 2 (Robust Feasibility with State Error) If $\mathsf{P}(\hat{\mathbf{x}}(0), \hat{\mathcal{Y}}, \hat{\mathcal{W}})$ has a feasible solution then the system (1), with disturbances and estimate errors obeying (13) and (14), and controlled using Algorithm 2, is robustly-feasible and satisfies the constraints (3).

Proof The result of Theorem 1 applies for the system (12), with the estimate $\hat{\mathbf{x}}$ in place of the true state $\mathbf{x}$. Perfect information is known for this system and it is acted upon by an effective process noise (13) bounded by (14). Therefore Theorem 1 guarantees robust feasibility and satisfaction of the following constraints

$$\hat{\mathbf{y}}(k) = \mathbf{C}\hat{\mathbf{x}}(k) + \mathbf{D}\mathbf{u}(k) \in \hat{\mathcal{Y}} \qquad (17)$$

It remains to show that $\hat{\mathbf{y}}(k) \in \hat{\mathcal{Y}}$ implies $\mathbf{y}(k) \in \mathcal{Y}$. The true output is given by

$$\mathbf{y}(k) = \hat{\mathbf{y}}(k) + (-\mathbf{C})\mathbf{e}(k) \qquad (18)$$

So using the error bound (11), the constraints (16) and the property (7) of the Pontryagin difference gives

$$\hat{\mathbf{y}}(k) \in \hat{\mathcal{Y}} \Rightarrow \mathbf{y}(k) \in \mathcal{Y} \; \forall \mathbf{e}(k) \in \mathcal{E} \qquad (19)$$

satisfying the true constraints (3). ∎

B. Finite Memory Estimator

This section describes the implementation of a finite memory estimator that is suitable for employment within the MPC scheme described in Section III-A. Let the measurement be

$$\mathbf{z}(k) = \mathbf{F}\mathbf{x}(k) + \mathbf{v}(k) \qquad (20)$$

where $\mathbf{v}(k)$ is a random noise vector belonging to a known bounded set $\mathcal{V}$ for all k. Define the following notation for stacked vectors

$$\mathbf{r}(j;k) = \left(\mathbf{r}(j)^T \; \mathbf{r}(j+1)^T \; \cdots \; \mathbf{r}(k)^T \right)^T$$

for a general discrete time vector signal $\mathbf{r}(\cdot)$. Then the combined equations for N_m past measurements give

$$\begin{aligned}
\mathbf{z}(k - N_m + 1; k) &= \mathbf{R}\mathbf{x}(k - N_m + 1) \qquad (21) \\
&+ \mathbf{S}\mathbf{u}(k - N_m + 1; k - 1) \\
&+ \mathbf{T}\mathbf{w}(k - N_m + 1; k - 1) \\
&+ \mathbf{v}(k - N_m + 1; k)
\end{aligned}$$

where

$$\mathbf{S} = \begin{bmatrix} \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{FB} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{FAB} & \mathbf{FB} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \mathbf{0} \\ \mathbf{FA}^{(N_m-2)}\mathbf{B} & \mathbf{FA}^{(N_m-3)}\mathbf{B} & \cdots & \mathbf{FB} \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} \mathbf{F} \\ \mathbf{FA} \\ \vdots \\ \mathbf{FA}^{(N_m-1)} \end{bmatrix}, \mathbf{T} = \begin{bmatrix} \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{F} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{FA} & \mathbf{F} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \mathbf{0} \\ \mathbf{FA}^{(N_m-2)} & \mathbf{FA}^{(N_m-3)} & \cdots & \mathbf{F} \end{bmatrix}$$

If the system $(\mathbf{A}, \mathbf{F})$ is observable, then the matrix $\mathbf{R}$ has full rank and its pseudo-inverse, denoted by $\mathbf{R}^\dagger$ and satisfying $\mathbf{R}^\dagger \mathbf{R} = \mathbf{I}$, can be found. This inverse can be used to obtain an estimate of $\mathbf{x}(k - N_m + 1)$, which can then be propagated to get an estimate of $\mathbf{x}(k)$

$$\begin{aligned}
\hat{\mathbf{x}}(k) &= \mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger \mathbf{z}(k - N_m + 1; k) \qquad (22) \\
&+ \left(\mathbf{P} - \mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger \mathbf{S} \right) \mathbf{u}(k - N_m + 1; k - 1)
\end{aligned}$$

where

$$\mathbf{P} = \left[\mathbf{A}^{(N_m-2)}\mathbf{B} \cdots \mathbf{AB} \; \mathbf{B} \right]$$

Then the error in the estimate is given by

$$\begin{aligned}
\mathbf{e}(k) &= \hat{\mathbf{x}}(k) - \mathbf{x}(k) \\
&= \mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger \mathbf{v}(k - N_m + 1; k) \qquad (23) \\
&+ \left(\mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger \mathbf{T} - \mathbf{Q} \right) \mathbf{w}(k - N_m + 1; k - 1)
\end{aligned}$$

where

$$\mathbf{Q} = \left[\mathbf{A}^{(N_m-2)} \cdots \mathbf{A} \; \mathbf{I} \right]$$

If the measurement noise $\mathbf{v}$ and process noise $\mathbf{w}$ lie in bounded sets $\mathcal{V}$ and $\mathcal{W}$, respectively, vector summation can be used to find the set of all possible estimation errors

$$\begin{aligned}
\mathcal{E} &= \mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger (\mathcal{V} \times \mathcal{V} \times \cdots \times \mathcal{V}) \qquad (24) \\
&\oplus \left(\mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger \mathbf{T} - \mathbf{Q} \right) (\mathcal{W} \times \mathcal{W} \times \cdots \times \mathcal{W})
\end{aligned}$$

such that $\mathbf{e}(k) \in \mathcal{E}$ for all k.

C. Combining Finite Memory Estimator with MPC

The expression (24) for the estimation error can be used in (15) to find the set of all possible prediction errors. However, the expression (15) is only a tight bound if the quantities $\mathbf{e}(k)$, $\mathbf{e}(k + 1)$ and $\mathbf{w}(k)$ are independently distributed within their given sets. This is not true in general, as there is some overlap in the measurements used to determine any two successive estimates. This section identifies a better bound on the prediction error $\hat{\mathbf{w}}(k)$ than (15).

The expression for the estimation error (23) can be substituted into the prediction error (13) to give an expression for $\hat{\mathbf{w}}(k)$ in terms of independent quantities

$$\hat{\mathbf{w}}(k) = \tilde{\mathbf{P}}\mathbf{v}(k - N_m + 1; k + 1) + \tilde{\mathbf{Q}}\mathbf{w}(k - N_m; k)$$

where

$$\tilde{\mathbf{P}} = \begin{bmatrix} -\mathbf{A}^{N_m}\mathbf{R}^\dagger \mid \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \mid \mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger \end{bmatrix}$$

$$\tilde{\mathbf{Q}} = \begin{bmatrix} -\mathbf{A}\left(\mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger\mathbf{T} - \mathbf{Q}\right) \mid \mathbf{0} \end{bmatrix} + \\ \begin{bmatrix} \mathbf{0} \mid \left(\mathbf{A}^{(N_m-1)}\mathbf{R}^\dagger\mathbf{T} - \mathbf{Q}\right) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \cdots \mathbf{0} \mid \mathbf{I} \end{bmatrix}$$

Then an improved set bound for the prediction error (14), based on independent quantities, is given by

$$\hat{\mathcal{W}} = \tilde{\mathbf{P}}(\mathcal{V} \times \mathcal{V} \times \cdots \times \mathcal{V}) \oplus \tilde{\mathbf{Q}}(\mathcal{W} \times \mathcal{W} \times \cdots \times \mathcal{W}) \quad (25)$$

D. Example

Algorithm 2 and the estimator from Section III-B were used in simulation to control a simple double integrator plant with control and state constraints. Fig. 2(a) shows the estimation error $\mathbf{e}(k)$ at each step plotted in the state space. Observe that the errors $\mathbf{e}(k)$ lie within the set of all possible estimation errors $\mathcal{E}$ obtained from (24), shown shaded. Fig. 2(b) shows the state prediction error $\hat{\mathbf{w}}(k)$, again plotted as points in state space. The points all lie within the inner set, obtained using (25), the expression for the prediction error set that incorporates the correlation between estimation errors and process noise. The larger set, shown for comparison, is the set found using (15), assuming that estimation errors and process noise are all independent. This is a very large superset of the actual error set, showing that the assumption of independence leads to very conservative control, due to excessive constraint tightening, and that it is very important to analyze the estimator for prediction error.

E. Time Delay

This section extends the formulation to account for a time delay in the loop. Delay can arise from several effects, including measurement, computation and communication latencies. We assume that the length of the delay is known. Adopting the approach of [8], the state measurement is propagated forward by the length of the delay to obtain the initial condition for each MPC optimization. This may be regarded as an estimate of the state at the time the plan begins. The analysis proceeds by converting the delayed-state case to the uncertain-state case of Section III-A.

Consider an LTI system, in continuous time, with state $\mathbf{x}_c(t)$ and control $\mathbf{u}_c(t)$ at time t and the following dynamics

$$\dot{\mathbf{x}}_c(t) = \mathbf{A}_c\mathbf{x}_c(t) + \mathbf{B}_c\mathbf{u}_c(t) + \mathbf{E}_c\mathbf{w}_c(t) \quad (26)$$

The control is updated at discrete times with step length T using a zero-order hold

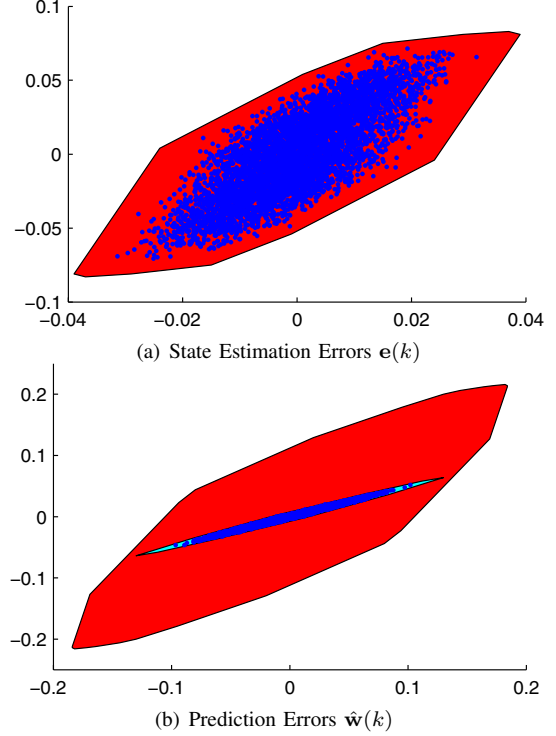$$\mathbf{u}_c(t) = \mathbf{u}(k) \quad kT \leq t < (k+1)T \quad (27)$$

(a) State Estimation Errors $\mathbf{e}(k)$

(b) Prediction Errors $\hat{\mathbf{w}}(k)$
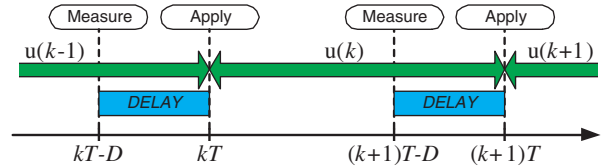
Fig.2: Error Sets and Simulation Results.

Fig.3: Timing Diagram for Problem with Delay

hence the dynamics of discrete-time state can be written in the form of (1) as

$$\overbrace{\mathbf{x}_c((k+1)T)}^{\mathbf{x}(k+1)} = \overbrace{e^{\mathbf{A}_cT}}^{\mathbf{A}}\overbrace{\mathbf{x}_c(kT)}^{\mathbf{x}(k)} + \overbrace{\int_0^T e^{\mathbf{A}_c(T-\tau)}d\tau\mathbf{B}_c}^{\mathbf{B}}\mathbf{u}(k)$$
$$+ \underbrace{\int_0^T e^{\mathbf{A}_c(T-\tau)}\mathbf{E}_c\mathbf{w}_c(kT+\tau)d\tau}_{\mathbf{w}(k)} \quad (28)$$

The discrete-time state estimates are based on measurements delayed by time D (see Fig. 3), where the delay is assumed to be less than the time step $D < T$. The delay can include measurement, computation and actuation latencies. At time $t = kT - D$, a measurement is taken and used to form an estimate of the state, corrupted by an error $\mathbf{n}(k)$. That estimate is then propagated forward using the model to form a prediction of the state at time kT given by

$$\hat{\mathbf{x}}_c(kT|kT - D) = e^{\mathbf{A}_cD}\mathbf{x}_c(kT - D) + e^{\mathbf{A}_cD}\mathbf{n}(k) \quad (29)$$
$$+ \int_0^D e^{\mathbf{A}_c(D-\tau)}d\tau\mathbf{B}_c\mathbf{u}(k-1)$$

A similar expression, but including the disturbance in place of the estimation error, gives the true state at time kT in terms of the true state at $kT - D$

$$
\begin{aligned}
\mathbf{x}_c(kT) = {} & e^{\mathbf{A}_c D} \mathbf{x}_c(kT - D) \\
& + \int_0^D e^{\mathbf{A}_c(D-\tau)} d\tau \mathbf{B}_c \mathbf{u}(k-1) \\
& + \int_0^D e^{\mathbf{A}_c(D-\tau)} \mathbf{E}_c \mathbf{w}_c(kT - D + \tau) d\tau
\end{aligned}
\tag{30}
$$

Subtracting (30) from (29) gives an equation for the effective estimation error at time kT

$$
\overbrace{\hat{\mathbf{x}}_c(kT|kT-D)}^{\hat{\mathbf{x}}(k)} - \overbrace{\mathbf{x}_c(kT)}^{\mathbf{x}(k)} = \tag{31}
$$
$$
\underbrace{e^{\mathbf{A}_c D} \mathbf{n}(k) - \int_0^D e^{\mathbf{A}_c(D-\tau)} \mathbf{E}_c \mathbf{w}_c(kT - D + \tau) d\tau}_{\mathbf{e}(k)}
$$

Comparing (31) and (28) with (10a) and (1), respectively, it is clear that the uncertainty introduced by the delay can be considered in the same framework as estimation error and disturbance. If bounding sets are known for disturbance $\mathbf{w}_c(t)$ and estimation error $\mathbf{n}(k)$, then a bound $\hat{\mathcal{W}}$ satisfying (14) can be obtained and the formulation of Section III-A is applicable. The only change to Algorithm 2 is the prediction in the estimate.

Algorithm 3 (Robust MPC with Time Delay)
1) (Time $t = kT - D$): Take measurement and form estimate $\hat{\mathbf{x}}(kT - D)$
2) Predict state D ahead $\hat{\mathbf{x}}(kT|kT - D)$ using (29)
3) Solve MPC optimization starting from predicted state $P(\hat{\mathbf{x}}(kT|kT - D), \hat{\mathcal{Y}}, \hat{\mathcal{W}})$
4) (Time $t = kT$): Update control input using $\mathbf{u}(k) = \mathbf{u}^*(k|k)$ as in (27)
5) Increment k. Go to Step 1

IV. ADAPTIVE PREDICTION ERROR BOUNDS

This section describes a method for deriving estimated bounds on the prediction error from online measurements. The method develops online an estimated set $\tilde{\mathcal{W}}(k)$ such that, if the MPC constraints are tightened to accommodate disturbances in $\tilde{\mathcal{W}}(k)$, then the problem will remain feasible with some probability λ' for a specified number of steps.

A. Bound Estimation for a Random Signal

This section briefly presents a method for estimating the maximum value of a random signal over a future window based on the observed maximum over a past window. The reader is referred to Ref. [11] for a thorough derivation.

Assume the stationary discrete-time scalar random process $X(k)$ is white and Gaussian with zero mean and Σ standard deviation, where Σ is a random variable, uniformly distributed in $[\underline{\sigma}, \overline{\sigma}]$. Define

$$
\begin{aligned}
Z &= \max_{k \in \{1..N\}} X(k) \\
Y &= \max_{k \in \{(N+1)..(N+M)\}} X(k)
\end{aligned}
$$

as the past (observed) and future (to be estimated) values of $X(k)$, respectively. Then the conditional probability density of Y given Z is

$$
f_{Y|Z}(y, z) =
$$
$$
\frac{M \int_{\underline{\sigma}}^{\overline{\sigma}} [F_{X|\Sigma}(y,\sigma)]^{M-1} f_{X|\Sigma}(y,\sigma) [F_{X|\Sigma}(z,\sigma)]^{N-1} f_{X|\Sigma}(z,\sigma) d\sigma}{\int_{\underline{\sigma}}^{\overline{\sigma}} [F_{X|\Sigma}(z,\sigma)]^{N-1} f_{X|\Sigma}(z,\sigma) d\sigma}
$$

where $f_{X|\Sigma}(x, \sigma)$ and $F_{X|\Sigma}(x, \sigma)$ are the conditional (given variance Σ) probability density and cumulative probability functions for $X(k)$, respectively. This can be numerically integrated and inverted to give a bound estimation function

$$
G_{NM}(z, \lambda) = y \Leftrightarrow \int_{-\infty}^y f_{Y|Z}(u, z) du = \lambda
$$

Hence if the measured maximum over a measured N samples is z, then with probability λ, a further M samples will not exceed $G_{NM}(z, \lambda)$.

B. Bounding Set Estimation for Vector Signals

This section employs the scalar bound estimate function $G_{NM}(z, \lambda)$ to identify bounds on multivariable signals. Assume a vector random process $\mathbf{x}(k)$ is white and Gaussian. Then for any vector $\mathbf{e}$ of suitable size, $X(k) = \mathbf{e}^T \mathbf{x}(k)$ is a scalar, white Gaussian random signal and the analysis in Section IV-A can be used to derive an estimated bound on $\mathbf{e}^T \mathbf{x}(k)$. Consider a set of L such vectors $\{\mathbf{e}_i\}$ such that the polytope defined by

$$
\mathbf{Ex} \le \mathbf{q}, \quad \mathbf{E} = [\mathbf{e}_1 \cdots \mathbf{e}_L]^T \tag{33}
$$

is bounded for all $\mathbf{q}$. Then set the threshhold values using

$$
q_i = G_{NM}\left(\max_{k \in \{1...N\}} \mathbf{e}_i^T \mathbf{x}(k), \lambda \right) \tag{34}
$$

where

$$
\lambda = 1 - \frac{(1 - \lambda')}{L} \tag{35}
$$

Then the probability of the future maximum $\mathbf{y}$ lying inside the polytope (33) is bounded by

$$
P(\mathbf{Ey} \le \mathbf{q}) \ge \lambda'
$$

The designer chooses $\lambda' \in (0, 1)$ to set the level of risk, with values closer to 1 giving smaller probability of exceeding the estimated bound but slower adaptation.

C. Adaptive MPC

This section describes the combination of the bound estimation from Section IV-B and the robustly-feasible MPC from Section II to form an adaptive controller, using the arrangement shown in Fig. 1(b). The prediction error can be written as

$$
\hat{\mathbf{w}}(k) = \hat{\mathbf{x}}(k+1) - \mathbf{x}(k+1|k) \tag{36}
$$

Both quantities on the right hand side are known at time $k + 1$, so measurements of $\hat{\mathbf{w}}$ can be taken and used

to form an estimated error set $\tilde{\mathcal{W}}(k)$ using the method in Section IV-B, such that with probability λ', all future prediction errors $\hat{\mathbf{w}}(k+j)$ over some window $j = \{1 \dots M\}$ lie within set $\tilde{\mathcal{W}}(k)$, and the optimization constraints are tightened accordingly. Feasibility is guaranteed provided the prediction error $\hat{\mathbf{w}}(k)$ remains in the set $\tilde{\mathcal{W}}(k)$, but exceeding the bound on any given step does not necessarily cause infeasibility. Therefore, the probability of infeasibility is less than $1 - \lambda'$.

A further complication when employing the bound estimation within adaptive MPC is that the error bound must always be shrinking, *i.e.* it is required that $\tilde{\mathcal{W}}(k + 1) \subseteq \tilde{\mathcal{W}}(k)$. The guarantee of robust feasibility relies on the feasibility of a particular candidate solution and this may not hold if the optimization constraints are tightened. To avoid this possibility, restrict the adaptation algorithm so that the set estimate is non-enlarging, using the estimate equation (34) only if the bound is decreasing

$$q_i(k + 1) = \min \left\{ G_{NM} \left(\max_{k \in \{0 \dots k\}} \mathbf{e}_i^T \hat{\mathbf{w}}(j), \lambda \right), q_i(k) \right\} \quad (37)$$

where $q_i(k)$ is the estimated bound in direction i at time k. In practice, this update should be performed relatively infrequently, *e.g.* once every 1000 steps. This reduces computation and also addresses an underlying assumption of sample independence (see Ref. [11]).

Algorithm 4 (Adaptive MPC)

1) (Initialization): Choose weighting matrix $\mathbf{E}$ and initial limits $\mathbf{q}(0)$
2) Take measurement and form current estimate $\hat{\mathbf{x}}(k)$ (including propagation for delay if necessary as in Section III-E)
3) Measure and store prediction error $\hat{\mathbf{w}}(k)$ using (36)
4) **If** k is an error-set update step, **then** update error bounds using (37), **else** $\mathbf{q}(k) = \mathbf{q}(k - 1)$.
5) Solve MPC optimization starting from estimate $\mathcal{P}(\hat{\mathbf{x}}(k), \hat{\mathcal{Y}}, \tilde{\mathcal{W}}(k))$
6) Apply first control input $\mathbf{u}(k) = \mathbf{u}^*(k|k)$
7) Increment k. Go to Step 2.

D. Examples

Fig. 4 shows the results of a simulation using adaptive MPC from Algorithm 4 for the double integrator example, previously seen in Section III-D, in which the aim is to keep the position within limits of ± 1 with minimum control energy. Error set updates were performed every 1000 steps. The top plot shows the position time history, with the dashed lines marking the constraints on the second plan step $\mathcal{Y}(1)$. The constraints can be seen to be relaxed at each update, most significantly at the first update, and the controller makes use of the relaxation, allowing greater state deviations and reducing the control energy use *i.e.* the slope of the bottom plot. The middle plot shows the time history of the prediction error and the estimated bounds. The bounds can be seen to be tightening but always overbounding the error.
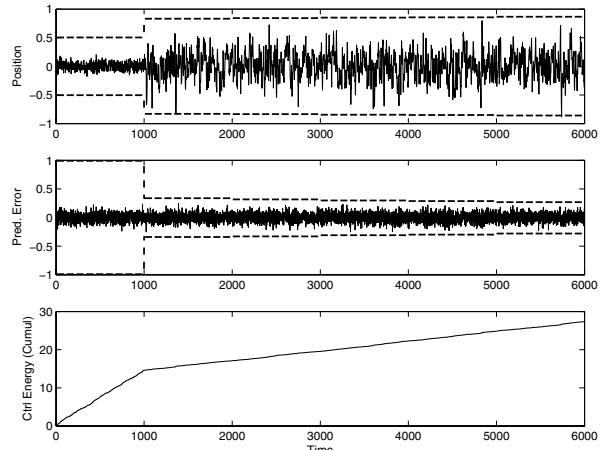

Fig.4: Time Histories from Adaptive MPC

In this case, the first 1000 samples give a good estimate of the error set.

V. Conclusions

Output feedback Model Predictive Control has been developed, guaranteeing robust constraint satisfaction and feasibility despite the presence of unknown but bounded measurement and process uncertainty. The same controller can accommodate time delay in the loop by including prediction in the estimator. Furthermore, for cases where the uncertainty bounds are not well known *a priori*, an adaptive form has been developed using online measurements to estimate the set of possible prediction errors.

Acknowledgements

References

[1] A. G. Richards, J. How, "A Decentralized Algorithm for Robust Constrained Model Predictive Control," *ACC*, Boston MA, 2004.
[2] D. P. Bertsekas and I. B. Rhodes, "On the Minmax Reachability of Target Sets and Target Tubes," *Automatica*, Vol. 7, 1971, pp. 233-247.
[3] R. Findeisen, L. Imsland, F. Allgower and B. A. Foss, "Output Feedback Stabilization of Constrained Systems with Nonlinear Predictive Control," *IJRNC* Vol. 13, 2003, pp. 211-227.
[4] L. Magni, G. De Nicolao and R. Scattolini, "Output Feedback Receding-Horizon Control of Discrete-Time Nonlinear Systems," IFAC NOLCOS 98.
[5] Y. I. Lee and B. Kouvaritakis, "Receding Horizon Output Feedback Control for Linear Systems with Input Saturation" 39th IEEE *CDC*, Sydney, 2000, p 656.
[6] H. Michalska and D. Q. Mayne, "Moving Horizon Observers and Observer-Based Control," IEEE *TAC*, Vol. 40 No. 6, June 1995, p. 995.
[7] R. K. Mehra, "On the Identification of Variances and Adaptive Kalman Filtering," IEEE *TAC*, Vol. 15 No. 2, April 1970, p. 175.
[8] R. Franz, M. Milam, and J. Hauser, "Applied Receding Horizon Control of the Caltech Ducted Fan," *ACC*, Anchorage, 2002.
[9] F. D. Busse, "Precise Formation-State Estimation in Low Earth Orbit Using Carrier Differential GPS," PhD Thesis, MIT, March 2003.
[10] I Kolmanovsky and E. G. Gilbert, "Maximal Output Admissible Sets for Discrete-Time Systems with Disturbance Inputs," *ACC*, Seattle, 1995, p.1995.
[11] A. G. Richards, "Robust Constraint Model Predictive Control," PhD Thesis, MIT, November 2004.