Distributed cooperative nonlinear economic MPC

Jaehwa Lee, and David Angeli

1 Introduction

Model predictive control (MPC) is a design technique that solves (typically on-line) optimization problems to determine a suitable feedback action. It is widely used because of the capability to deal with constraints and MIMO systems [3].

To treat practical industrial processes with MPC, use of so called economic cost functionals are proposed and considered for optimization in [1]. In this scheme, which is called economic MPC to differentiate from standard MPC, the economic cost function is not necessarily minimum at the desired steady state.

The Lyapunov-based analysis first achieved in [2] and later extended in [1], deals with a centralized control scheme. However, implementation of economically-based control algorithms is also of great interest in large-scale systems for which presence of a centralized optimization solver is not ideal.

In [3] control methods for distributed systems are classified into 4 categories according to two criteria, the extent of sharing information and the type of cost function. These go under the name of centralized, decentralized, noncooperative and cooperative control. In this paper we focus on the cooperative distributed MPC schemes. Cooperative MPC for linear systems with convex objective functions and nonlinear systems with nonconvex objective functions are dealt in [4] and [5] respectively.

Cooperative suboptimal MPC for linear distributed systems with convex economic cost functions is suggested in [6]. Similarly to the standard cooperative dis-

*We omit proofs for lemmas and theorems here due to the limited pages
†Dept. of Electrical and Electronic Engineering, Imperial College London, U.K., j.lee09@imperial.ac.uk
‡Dept. of Electrical and Electronic Engineering, Imperial College London, U.K. and Dip. Sistemi e Informatica, Univ. of Florence, Italy, d.angeli@imperial.ac.uk
ttributed MPC in [4], the proof for feasibility over each iteration highly relies on convexity of objective functions and constraints. In this note we suggest a control method based on multiple distributed iterations of control actions by means of interconnected communicating subsystems for nonlinear systems and nonconvex cost functions.

2 Economic MPC for distributed nonlinear model

2.1 Distributed nonlinear model

We assume a discrete-time nonlinear system which is governed by the following equation:

\[ x^{+} = f(x, u) \]  \hspace{1cm} (1)

where \( x \in X \subset \mathbb{R}^n \) and the control input \( u \in U \subset \mathbb{R}^m \) is partitioned as \( u = [u_1, \ldots, u_M] \) for some positive integer \( M \). For every \( i \in I_1:M \), \( u_i \in U_i \subset \mathbb{R}^{m_i} \), and \( m_1 + m_2 + \ldots + m_M = m \). The states of the systems are broadcasted to all controllers, so it is possible to design state-feedback controllers.

2.2 Economic model predictive control

For a nonnegative stage cost function \( \ell(\cdot) \) the plant-wide performance index to optimize is as follows:

\[ \sum_{k \geq 0} \ell(x(k), u(k)). \]  \hspace{1cm} (2)

The pointwise-in-time constraints with compact set \( W \) of suitable dimensions for plant operation are given as \((x(k), u(k)) \in W, \forall k \geq 0\).

The best steady-state solution is therefore the pair \((x_s, u_s)\) which satisfies

\[ (x_s, u_s) = \arg\min_{(x,u) \in \mathbb{R}^n \times \mathbb{R}^m} \ell(x, u). \]  \hspace{1cm} (3)

Throughout this paper we assume for the sake of simplicity \((x_s, u_s)\) to be unique, although this assumption can normally be removed with some extra-care. Now we define the objective function to be optimized over a sufficiently long but finite horizon \( N \), with the same stage cost used in (2) and (3):

\[ V(x, u) = \sum_{k=0}^{N-1} \ell(x(k), u(k)) \]  \hspace{1cm} (4)

in which \( u = [u(0), \ldots, u(N-1)] \), \( x^{+} = f(x, u) \) and \( x(0) = x \). Then the so called Centralized Economic Model Predictive Control is the feedback law that is obtained by solving on-line the following optimization problem:

\[ \min_{\nu} V(x, \nu) \]

subject to \((z(k), v(k)) \in W, \forall k \in \{0, 1, \ldots, N-1\}\)

\[ z^{+} = f(z, v) \]

\[ z(0) = x, z(N) = x_s, \]  \hspace{1cm} (5)
and by $\mathbf{v}^*$ we denote the optimal solution. As a control policy, which gives asymptotic performance at least as good as that of the best equilibrium point, we apply the first element of $\mathbf{v}^*$ to the plant (1). That is:

\[ u(t) = v^*(0) \]  

The state $x(t)$ is called feasible if the centralized control problem (5) admits a feasible solution.

**Definition 1 (Feasible set for nonlinear systems).** The feasible set $\mathcal{Z}_N$ as the set of $(x,u)$ pairs is defined as follows:

\[ \mathcal{Z}_N = \left\{ (x,u) \in \mathbb{R}^n \times \mathbb{R}^{Nm} \mid x(0) = x, x(N) = x_s, 
\begin{align*}
    x(k + 1) &= f(x(k), u(k)), 
    (x(k), u(k)) \in \mathcal{W}, 
    \forall k \in \{0, N-1\} \right\}. 
\right. \]  

The set of admissible states $\mathcal{X}_N$ is then defined as $\mathcal{X}_N := \{x\mid \exists u \text{ s.t. } (x,u) \in \mathcal{Z}_N \}$. Forward invariance of $\mathcal{X}_N$ is a well-known property under (6), so $x \in \mathcal{X}_N$ implies $f(x,v^*(0)) \in \mathcal{X}_N$.

### 2.3 Cooperative control

We propose a new way to implement a decentralized iteration for the computation of the control action. Our goal is designing a cooperative controller, so that subsystems seek to optimize the common objective function while satisfying state and input constraints without a central coordinator. By letting $\mathbf{u}^p$ denote the $p$-th iteration for the computation of the optimal control, the following partition is defined:

\[ \mathbf{u}^p = [\mathbf{u}^p_1, \mathbf{u}^p_2, \ldots, \mathbf{u}^p_M]. \]  

For the update of individual components of $\mathbf{u}^p$, it is necessary to define $\mathbf{u}^*_i$:

\[ \mathbf{u}^*_i(x(t), (\mathbf{u}_1, \ldots, \mathbf{u}_{i-1}, \mathbf{u}_{i+1}, \ldots, \mathbf{u}_M)) 
= \arg \min_{\mathbf{v}_i} V(z, \mathbf{v}) 
\text{subject to } z^+ = f(z, \mathbf{v}) 
\begin{align*}
    (z(k), v(k)) &\in \mathcal{W}, \quad \forall k \in \{0, 1, \ldots, N-1\} \\
    z(0) &= x(t), \quad z(N) = x_s \\
    \mathbf{v}_j &= \mathbf{u}_j, \quad j \in \{1, \ldots, i-1, i+1, \ldots, M\}. 
\end{align*} \]  

Through this paper we limit our discussion to the case $M = 2$, which means a system with two subsystems. Extensions to higher number of subsystems are possible in the form of gossiping-type iterations. In other words, the fact that only 2 subsystems update their current virtual control value at any given iteration is crucial to the results proposed in the present note.
Definition 2 (Individually feasible). The pair \((x, u)\) is called individually feasible if there exist control \(v_j\) \(j = \{1, 2\}\) such that
\[
(x, (v_1, u_2)) \in \mathcal{Z}_N \quad \text{and} \quad (x, (u_1, v_2)) \in \mathcal{Z}_N.
\]

The iteration to compute the optimal strategy can then be defined as follows:
\[
\begin{bmatrix}
u_{p+1}^1 \\
u_{p+1}^2
\end{bmatrix} := h(u_p) =
\begin{bmatrix}
u_1^*(u_2^p) \\
u_2^*(u_1^p)
\end{bmatrix}.
\] (10)

For the sake of simplicity we dropped dependence of the optimal solution on the current state. The following lemma shows that feasibility of computed control actions is preserved for all \(p\)-th iterates.

Lemma 3. For an individually feasible input sequence and state pair \((x, u^0)\), the next input and state pair \((x, u^1)\) proposed in (10) is also individually feasible. Then, by induction, \((x, u^p)\) is also individually feasible for all \(p \geq 0\).

Note that if we stop iterating at the \(p\)-th step, the control policy applied is either \((u_{p+1}^1, u_2^p)\) or \((u_1^p, u_{p+1}^2)\), which are feasible from the definition (9) while \((u_1^p, u_2^p)\) is not necessarily feasible. We inject the first element of one of the sequences to each subsystem as the control policy at the current time.

2.4 Assumptions

For subsystems \(i \in \{1, 2\}\), the following Lipschitz continuity and weak controllability assumptions are necessary to implement the distributed controller.

Assumption 1 (Lipschitz Continuity). Both \(f(x, u)\) and \(\ell(x, u)\) are locally Lipschitz, so there always exist constants \(L_f\) and \(L_{\ell}\) which satisfy
\[
|f(x, u) - f(x_0, u_0)| \leq L_f |(x, u) - (x_0, u_0)|
\]
\[
|\ell(x, u) - \ell(x_0, u_0)| \leq L_{\ell} |(x, u) - (x_0, u_0)|.
\]
for all \((x, u) \in \mathcal{W}\).

Assumption 2 (Weak Controllability). There exists a \(K_{\infty}\)-function \(\gamma(\cdot)\) such that for every \(x \in \mathcal{X}_N\), there exists \(u\) such that \((x, u) \in \mathcal{Z}_N\) and
\[
\sum_{k=0}^{N-1} |u_k - u_s| \leq \gamma(|x - x_s|).
\]

This condition is weaker than controllability assumption, but bounds the total cost of steering \(x\) to the steady state \(x_s\) [2]. A Lyapunov stability analysis is suggested in [2] under the strong duality assumption with rotated cost function for
MPC with an economic stage cost function $\ell(\cdot)$. We state the same assumption and rotated cost function here to proceed our discussion.

Assumption 3 (Strong duality of steady-state problem). There exists a multiplier $\lambda_s$ so that $(x_s, u_s)$ uniquely solves
\[
\min_{x,u} \ell(x,u) + [x - f(x,u)]'\lambda_s \quad \text{s.t.} \quad (x,u) \in \mathbb{W}.
\]
Furthermore, there exists a $K_\infty$-function $\beta$ such that
\[
L(x,u) := \ell(x,u) + [x - f(x,u)]'\lambda_s - \ell(x_s,u_s)
\]
satisfies $L(x,u) \geq \beta(|x - x_s|)$ for all $(x,u) \in \mathbb{W}$.

2.5 Suboptimal MPC

We propose a controller for cooperative control in terms of suboptimal MPC as specified in [3]. With a current feasible state $x \in X_N$ and a feasible initial input trajectory $\tilde{u} \in U$, each subsystem $i$ performs iterations to generate a new control sequence $u_i$ which minimizes the objective function. The iterations are synchronized between subsystems, and the first component $u_i(0)$ is applied to the plant for the state at the next time step according to the system evolution equation $x^{k+1} = f(x_k, u_k)$. Given a feasible input sequence $u$, we set $\tilde{u}^+ := \{u(1), u(2), \ldots, u(N-1), u_s\}$ as a warm start, and $h(\tilde{u})$ as the $p$-th iterate of (10) from $\tilde{u}$. In particular, $h^p(\tilde{u})$ denotes the $i$-th component $i = \{1, 2\}$ of $h^p(\cdot)$. More details are discussed in later Sections.

2.6 Stability and average performance

In the standard MPC scheme, defining a cost-to-go and using it as a candidate Lyapunov function is a well-known method to establish closed-loop stability of equilibria [3]. In our notation it is stated as follows:

\[
\hat{V}(x) := \min_{u} V(x, u)
\]
subject to $x(k+1) = f(x(k), u(k))$, $k \in I_{\leq 0}$
\[
(x(k), u(k)) \in \mathbb{W}, \quad k \in I_{0:N-1}
\]
\[
x(0) = x, \quad x(N) = x_s
\]

Here, the assumption $\ell(x_s, u_s) \leq \ell(x,u)$, which implies $0 = \hat{V}(x_s) \leq \hat{V}(x)$, holds for all feasible states. Along solutions of the closed-loop systems the following inequality holds for all $k \geq 0$:

\[
\hat{V}(x(k+1)) - \hat{V}((x(k)) \leq \ell(x_s, u_s) - \ell(x(k), u(k)) \quad (11)
\]

This condition implies the monotonic decrease of cost-to-go function evaluated along solutions of closed-loop systems. The monotonicity is important because the cost-to-go function is used as a Lyapunov candidate to prove asymptotic stability under some mild conditions.
In contrast, the monotonic decrease (11) does not hold in economic MPC, so Lyapunov-like analysis is generally unavailable to prove asymptotic stability with the cost-to-go function.

One of the strength of economic MPC is that even if the stability is not guaranteed, asymptotic performance is preserved. The performance aspect of economic MPC is analyzed in [1]. Its main result is that for a feasible initial state $x \in X_N$, a closed-loop system has an average performance no worse than that of the constant feasible best steady state, viz, $(x_s, u_s)$. We present next that similar performance bounds hold in the case of cooperative decentralized MPC.

**Lemma 4.** Let $(x, u) \in Z_N$, where $u = [u'_1, u'_2]'$. We implement the following control strategy:

$$
\begin{bmatrix}
  x^+
  \\
  u_1^+
  \\
  u_2^+
\end{bmatrix} =
\begin{bmatrix}
  f(x, u_1(0), u_2(0)) \\
  h_1^{p+1}(x, u_1^+, u_2^+) \\
  h_2^p(x, u_1^+, u_2^+) 
\end{bmatrix}
$$

where $p$ is any positive integer. Then,

1. $(x(t), u(t)) \in Z_N$ for all $t \geq 0$;
2. $(x(t), u(t)) \in W$, for all $t \geq 0$;
3. $\lim \sup_{T \to +\infty} \frac{\sum_{t=0}^{T-1} f(x(t), u(t))}{T} \leq \ell(x_s, u_s)$.

We point out the key technical step for proving the stability and performance results is to show that the (centralized) optimizer yields a solution at least as good as the ‘warm start’ which is obtained by shifting the feasible solution at the previous sample time [6]. In this context this allows a decentralized solution computed by means of an arbitrary number of iterations.

### 3 Results

In this section we analyze the stability and convergence properties of the proposed algorithm. Two kinds of iterations are studied, time iterations, leading to a standard asymptotic stability analysis, and players iterations, which occur within a single sampling interval. Players’ decision in linear distributed systems with convex objective functions converges to the centralized control along iterations [2], [3], [6]. Hence it ensures the Pareto optimality of the entire system.

#### 3.1 Convergence of the players’ iteration

To analyze the convergence of players iterations, we use the sum of two objective functions $W(x, u_1, u_2) := V(x, u_1, u_2^*(u_1)) + V(x, u_1^*(u_2), u_2)$, where $u_i(\cdot)$ is the solution of (9), for a Lyapunov-like analysis. The following lemma shows that $W(\cdot)$ is monotonically decreasing for the iteration (10).
Lemma 5. For the iteration given in (10), $W(x, u_1^p, u_2^p)$ monotonically decreases as $p$ increases.

Remark 1. If we iterate $u_1$ once more after $p$-th iteration, $V(x, h_1^{p+1}(u), h_2^p(u)) \leq V(x, u_1, u_2)$ holds for all $p \geq 0$.

Lemma 6. The sequence $W(x, u_1^p, u_2^p)$ admits a limit as $p \to \infty$. Furthermore, if the value function $W(x, u^+)$ is continuous with respect to $u$, the sequence $(u_1^p, u_2^p)$ approaches the set

$$K_x := \{(u_1, u_2) \mid W(x, u_1^+, u_2^+) = W(x, u_1, u_2)\}$$

as $p \to +\infty$. That is, $d((u_1^p, u_2^p), K_x) \to 0$ as $p \to +\infty$.

Lemma 7. For all $(v_1, v_2) \in K_x$, $(u_1^*(v_2), v_2)$ is a local minimizer with respect to both controls of subsystems $u_1$ and $u_2$. That is

$$V(x, u_1^*(v_2), v_2) = \min_{u_1} V(x, u_1, v_2) \quad (13)$$
$$V(x, u_1^*(v_2), v_2) = \min_{u_2} V(x, u_1^*(v_2), u_2). \quad (14)$$

Remark 2. Lemma 7 implies that if $(u_1^*(v_2^p), u_2^p)$ converges to a point as $p \to \infty$. This is a Nash equilibrium. Since Nash equilibrium cannot guarantee the Pareto efficiency, we can only state it as a suboptimal MPC controller. Normally we cannot generate the same performance of a centralized optimizer.

3.2 Closed-loop stability of the system

We now establish the stability of the closed-loop system (12) under the assumptions previously discussed.

Theorem 8 (Closed-loop Stability). Under the given assumptions 1, 2 and 3, the equilibrium point $x_*$, which satisfies the steady-state condition (3), is asymptotically stable on the set of feasible states for the closed-loop system (12).

4 Conclusion

The control design scheme suggested in this paper is the suboptimal economic MPC controller for nonlinear distributed systems. Individual feasibility property, which is suggested in the previous section, enables the iteration so the controller for the whole system can be computed in a separated manner. The set of control sequence that iterates converge is a local minimizer of the objective function under the locally lipschitz assumption. The convergent set of input sequence over iteration is an equilibrium point which is expressed similary to Nash equilibrium in a non-cooperative game.

The closed-loop stability can be proved with the warm start which is shifted from the control sequence and then concatenated by the steady state input $u_*$. 
Bibliography


