Suboptimal distributed MPC based on a block-coordinate descent method with feasibility and stability guarantees

Ion Necoara

Abstract—In this paper we propose a distributed optimization algorithm to solving discrete linear model predictive control (MPC) problems for general networked systems. The new optimization algorithm is based on block coordinate descent updates in parallel with very simple iteration complexity and using only local information. We show that for smooth objective functions it has sublinear rate of convergence, while for strongly convex objective functions it converges linearly. An MPC controller based on this distributed optimization method is derived, for which classical controllers may not even provide stability, let alone performance, which is important when considering the economic benefits of reducing consumption of raw materials or recycling waste from industrial processes.

In modern control practice, networks of interconnected subsystems can be found in newer applications such as traffic control systems [4], dampening of seismic shocks on civilian structures [8], satellite flight formation [11], electricity generation from renewable energies [10], control for power system networks [21], etc. This paper focuses on distributed MPC for a network of dynamically coupled linear systems. Each subsystem in the network has only access to local information which is obtained by measuring its own state and by sharing knowledge with a number of neighboring systems. These aspects require the design of distributed synthesis and distributed MPC procedures. Many recent work has focused on distributed MPC for different type of interconnected systems. Distributed control, where the interactions between subsystems are only considered locally, has been approached in [2], [3], [9]. An hierarchical form of distributed MPC has been considered in [1], [17], where a coordinating MPC interacts with local, decentralized, controllers. In [19], [21], [15] the authors have approached the distributed MPC problem from the cooperative point of view, where distributed optimization algorithms are employed such that each controller optimizes a centralized controller objective.

In this paper we propose a distributed MPC strategy for a general networked system in which both state and input interactions between subsystems are modeled. Stability of the MPC scheme is guaranteed by using a linear feedback law, that allow us to construct local terminal cost functions in a completely distributed way. Compared with the existing approaches based on an end point constraint, we reduce the conservatism by combining the underlying structure of the system with distributed optimization. This leads to a larger region of attraction for the controller. For the MPC problem we consider only input constraints. We develop an efficient distributed optimization algorithm for solving the MPC problem, with a low iteration complexity and a guaranteed (sub)linear rate of convergence. This algorithm employs parallel block-coordinate updates for the optimization variables under the assumption of a coordinate-wise Lipschitz continuous gradient and it is similar to the optimization algorithm proposed in [19], but with a simpler iteration complexity. Because the optimization problem arising from MPC formulation is usually terminated before convergence, our distributed MPC controller is a form of suboptimal
control. However, using the theory of suboptimal control we can still guarantee feasibility and stability of the distributed MPC scheme presented in this paper.

The paper is organized as follows: In Section II we introduce the dynamics for a general network of subsystems, formulate the necessary conditions for ensuring the stability of the model predictive control problem via a terminal cost approach and provide the means for which this terminal cost can be distributively synthesized. In Section III we outline our parallel block-coordinate descent algorithm and prove the (sub)linear convergence rate for it. We also focus on stability of the distributed MPC scheme based on suboptimal control.

II. PROBLEM FORMULATION

In this paper we consider discrete-time networked systems which are comprised of $M$ interconnected subsystems, whose dynamics are defined by the following linear state space equations:

$$x^{i}_{t+1} = A^i x^i_t + B^i u^i_t + \sum_{j \in N^i \setminus i} A^{ij} x^j_t + B^{ij} u^j_t,$$  

(1)

where $x^i_t \in \mathbb{R}^{n_i}$ and $u^i_t \in \mathbb{R}^{m_i}$ represent the state and respectively the input of $i$th subsystem at time $t$, $A^i \in \mathbb{R}^{n_i \times n_i}$, $B^i \in \mathbb{R}^{n_i \times m_i}$, $A^{ij} \in \mathbb{R}^{n_i \times n_j}$ and $B^{ij} \in \mathbb{R}^{n_i \times m_j}$, and $N^i$ is the set of indices which contains the index $i$ and that of its neighboring subsystems. A particular case of the system (1), that is frequently found in literature [15], [17], [19], [21], has the following description:

$$x^{i}_{t+1} = A^i x^i_t + B^i u^i_t + \sum_{j \in N^i \setminus i} B^{ij} u^j_t,$$  

(2)

For the stability analysis, we also express the dynamics of the entire system as:

$$x_{t+1} = Ax_t + Bu_t,$$

where $x_t \in \mathbb{R}^n$, $u_t \in \mathbb{R}^m$ and $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, with $n = \sum_{i=1}^{M} n_i$ and $m = \sum_{i=1}^{M} m_i$.

For a system of type (1) or (2) we consider local input constraints of the form:

$$u^i_t \in U^i \quad \forall i = 1, \cdots, M, \forall t \geq 0,$$  

(3)

with $U^i \subseteq \mathbb{R}^{m_i}$, compact convex sets with the origin in their interior. We also consider local stage costs and a terminal cost of the form:

$$\ell^i(x^i_t, u^i_t) \quad \forall i = 1, \cdots, M, \quad \ell_t(x_t),$$

that can be e.g. of quadratic form:\footnote{In this paper, by $\|\cdot\|$ we denote the standard Euclidean norm, and by $\|x\|^2_P$ we denote the quadratic term $x^T P x$.} $\ell^i(x^i_t, u^i_t) = \frac{1}{2} \left( \|x^i_t\|^2_{Q^i} + \|u^i_t\|^2_{R^i} \right)$ and $\ell_t(x_t) = \frac{1}{2} \|x_t\|^2_P$, where the matrices $Q^i \in \mathbb{R}^{n_i \times n_i}$ are positive semidefinite, whilst matrices $R^i \in \mathbb{R}^{m_i \times m_i}$ and $P$ are positive definite. We can now formulate the MPC problem for system (1) over a prediction horizon of length $N$ and a given initial state $x$:

$$V^*(x) = \min_{x_t^i, u^i_t} \sum_{i=1}^{M} \sum_{t=0}^{N-1} \ell^i(x^i_t, u^i_t) + \ell_t(x_N)$$  

(4)

s.t. $x^i_{t+1} = A^i x^i_t + B^i u^i_t + \sum_{j \in N^i \setminus i} A^{ij} x^j_t + B^{ij} u^j_t,$

$$x^i_0 = x^i, \quad u^i_t \in U^i \quad \forall i = 1, \cdots, M, \quad t \geq 0.$$

A. Stability of MPC without end constraints

We assume that stability of this MPC scheme (4) is enforced by adapting the terminal cost $\ell_t$ and the horizon length $N$ appropriately such that sufficient stability criteria are fulfilled [7], [20]. Usually, stability of MPC with quadratic stage cost and without terminal constraint is enforced if the following criteria holds: there exists a stabilizing feedback law $\kappa(\cdot)$ and a terminal cost $\ell_t(\cdot)$ such that the following property is satisfied

$$\ell_t(Ax + B\kappa(x)) - \ell_t(x) + (\kappa(x)^T R \kappa(x) + x^T Q x) \leq 0 \quad \forall x \in \mathbb{R}^n,$$  

(5)

where the matrices $Q$ and $R$ have a block diagonal structure and are composed of the blocks $Q^i$ and $R^i$ on the diagonals, respectively.

Keeping in line with the distributed nature of our system, the control law $\kappa(\cdot)$ and the final stage cost $\ell_t(\cdot)$ need to be computed locally. In the sequel we develop a distributed synthesis procedure to construct them locally. For ensuring stability, we introduce a terminal cost for each subsystem:

$$\ell^i_t(x^i_N) = \|x^i_N\|^2_{P^i},$$

where the matrices $P^i \in \mathbb{R}^{n_i \times n_i}$ are positive definite and form the block-diagonal matrix $P = \text{diag}(P^i)$. For a locally computed $\kappa(\cdot)$, we employ distributed control laws:

$$u^i = K^i x^i,$$

with $K^i \in \mathbb{R}^{m_i \times n_i}$, i.e. $\kappa(\cdot)$ is taken linear with a block-diagonal structure $K = \text{diag}(K^i)$. We see that the terminal cost $\ell_t(\cdot)$ is a sum of local terminal costs of the form:

$$\ell_t(x_N) = \sum_{i=1}^{M} \ell^i_t(x^i_N).$$

Given $\ell_t(\cdot)$ of quadratic form, in terms of matrix inequality in the unknowns $P$ and $K$, (5) takes the well known form:

$$(A + BK)^T P (A + BK) - P + K^T R K + Q \preceq 0.$$

We need to solve this matrix inequality distributively. To this purpose, we first need to introduce vectors $x^{N^i} \in \mathbb{R}^{n^{N^i}}$ and $u^{N^i} \in \mathbb{R}^{m^{N^i}}$ for subsystem $i$, where $n^{N^i} = \sum_{j \in N^i} n_j$ and $m^{N^i} = \sum_{j \in N^i} m_j$. These vectors are comprised of the state and input vectors of subsystem $i$ and those of its neighbors:

$$x^{N^i} = [(x^j)^T, j \in N^i]^T, \quad u^{N^i} = [(u^j)^T, j \in N^i]^T.$$
Since our synthesis procedure needs to be distributed and taking into account that the terminal cost is a summation of local terminal costs, we impose the following distributed structure to ensure (5) (see also [6] for a similar approach):

\[
E_i ((x_i^i)^\top) - E_i ((x_i^i)) + (K^i x_i^i)^T R^i K^i x_i^i + (x_i^i)^T Q^i x_i^i \leq q_i^i (x_i^i) \quad \forall x_i^N_i \in \mathbb{R}^{n_i}, \quad i = 1, \ldots, M,
\]

such that:

\[
q(x) = \sum_{i=1}^{M} q_i^i (x_i^N_i) \leq 0.
\]

We assume that \( q_i^i (x_i^N_i) \) have also a quadratic form, with \( q_i^i (x_i^N_i) = \| x_i^N_i \|_{W_i}^2 \), where \( W_i \in \mathbb{R}^{n_i \times n_i} \). Being a sum of quadratic functions, \( q(x) \) can itself be expressed as a quadratic function, \( q(x) = \| x \|_W^2 \), where \( W \in \mathbb{R}^{n \times n} \) isformed from the appropriate block components of matrices \( W_i \). Note that we do not require that matrices \( W_i \) to be negative semi-definite. On the contrary, positive or indefinite matrices allow local terminal costs to increase so long as the global cost still decreases. This approach reduces the conservatism in deriving the matrices \( P_i \) and \( K_i \).

For obtaining \( P_i \) and \( K_i \), we introduce matrices \( E_n^i \in \mathbb{R}^{n \times n_i}, E_m^i \in \mathbb{R}^{m \times m}, J_n^i \in \mathbb{R}^{n_i \times n_i}, J_m^i \in \mathbb{R}^{m_i \times m} \) such that:

\[
x_i^i = E_n^i x, \quad u_i^i = E_m^i u, \quad x_i^N_i = J_n^i x, \quad u_i^N_i = J_m^i u.
\]

We can now define the matrices \( A_i^N_i = E_n^i A(J_n^i)^T \), \( B_i^N_i = E_n^i B(J_m^i)^T \) and \( K_i^N_i = J_m^i K(J_m^i)^T \), as to express the dynamics (1) for subsystem \( i \):

\[
x_{i+1}^i = (A_i^N_i + B_i^N_i K_i^N_i) x_i^N_i.
\]

Using these notations we can now recast inequality (6) as:

\[
(A_i^N_i + B_i^N_i K_i^N_i)^T P_i (A_i^N_i + B_i^N_i K_i^N_i) - J_m^i (E_m^i)^T P_i (Q^i + (K_i^i)^T R^i K_i^i) E_n^i (J_n^i)^T \preceq W_i.
\]

The task of finding suitable \( P_i, K_i \) and \( W_i \) matrices is now reduced to the following optimization problem:

\[
\begin{align*}
\min_{P_i, K_i, W_i} \quad & \tau \\
\text{s.t.:} \quad & \text{MI (8) } \forall i = 1, \ldots, M, \quad W \preceq \tau I.
\end{align*}
\]

It can be easily observed that if the optimal value \( \tau^* \leq 0 \), consequently \( W \leq 0 \) and (5) holds. This optimization problem, in its current nonconvex form, cannot be solved efficiently. However, it can be recast as a sparse SDP if we can reformulate (8) as an LMI. We need now to make the assumption that all the subsystems have the same dimension for the states, i.e. \( n_i = n_j \) for all \( i, j \). Subsequently, we introduce the following linearizations:

\[
P_i = (S^i)^{-1}, \quad K_i = Y^i G_i^{-1},
\]

and a series of matrices that will be of aid in formulating the LMIs:

\[
G_i^N = I_{|N_i|} \otimes G, \quad G_{i_bar} = [0 \quad I_{|N_i|-1} \otimes G],
\]

\[
Y_{i_bar}^j = K_i^j G_i, \quad j \in N_i, \quad Y_{N_i} = \text{diag}(Y^i, Y_{i_bar}^i) = K_i^N G_i^N,
\]

\[
S_{i_bar} = \text{diag}(S_i, \mu_i I_{(n_{x_i} - n_i)}),
\]

\[
T_{i_bar}^N = \begin{bmatrix} A_{i_bar}^N G_i^N + B_{i_bar}^N Y_{N_i}^I \end{bmatrix}, \quad T_i^I = \begin{bmatrix} (Q^i)^{\frac{1}{2}} G_i^N \end{bmatrix},
\]

where the 0 blocks are of appropriate dimensions.2

**Theorem 1:** If the following SDP:

\[
\begin{align*}
\min_{G, S_i, Y_i, Y_{i_bar}^i, \mu, \tau} \quad & \tau \\
\text{s.t.:} \quad & \begin{bmatrix} G_i^N + (G_{i_bar})^T - S_{i_bar} + \tilde{W}_{i_bar} & * & * \\
T_{i_bar}^N & S_{i_bar} & * & * \\
T_i^I & 0 & I \end{bmatrix} \succeq 0
\end{align*}
\]

\[
Y_{i_bar}^j = Y_j^i \quad \forall j \in N_i, \quad i = 1, \ldots, M, \quad \tilde{W} \preceq \tau I,
\]

has an optimal value \( \tau^* \leq 0 \), then (5) holds.

**Proof:** From (11) we observe that \( S_{i_bar} > 0 \), so that

\[
(S_{i_bar} - G_{i_bar})^T (S_{i_bar})^{-1} (S_{i_bar} - G_{i_bar}) \succeq 0,
\]

which in turn implies

\[
G_{i_bar} + (G_{i_bar})^T - S_{i_bar} \preceq (G_{i_bar})^T (S_{i_bar})^{-1} G_{i_bar}. \tag{12}
\]

If we apply the Schur complement to (11), we obtain:

\[
0 \preceq G_{i_bar} + (G_{i_bar})^T - S_{i_bar} \preceq (G_{i_bar})^T (S_{i_bar})^{-1} G_{i_bar},
\]

and by (12)

\[
(G_{i_bar})^{-T} \begin{bmatrix} (T_{i_bar}^N)^T (S_{i_bar})^{-1} T_{i_bar}^N + (T_i^I)^T T_i^I \end{bmatrix} (G_{i_bar})^{-1} - (S_{i_bar})^{-1} \preceq (G_{i_bar})^{-T} \tilde{W}_{i_bar} (G_{i_bar})^{-1},
\]

which is equivalent to (8) if we take \( W_i = (G_{i_bar})^{-T} \tilde{W}_{i_bar} (G_{i_bar})^{-1} \).

There exist in the literature many optimization algorithms for solving in parallel or distributively sparse SDP problems in the form (10).

### III. A DISTRIBUTED ALGORITHM BASED ON BLOCK COORDINATE DESCENT OPTIMIZATION

We now denote the input trajectory for subsystem \( i \) by:

\[
u_i^i = [(u_1^i)^T \cdots (u_{n_i}^i)^T]^T,
\]

and the overall input trajectory for the entire system as:

\[
u = [(u_1^1)^T \cdots (u_M^M)^T]^T.
\]

2By \( I_n \) we denote the identity matrix of size \( n \times n \), by \( \otimes \) we denote the standard Kronecker product and by \( |N_i| \) the cardinality of the set \( N_i \).

3By \( * \) we denote the transpose of the symmetric block of the matrix.
By eliminating the states from the dynamics (1), MPC problem (4) can be expressed as a large scale optimization problem of the form:

$$
\begin{align*}
    f^* &= \min_{u^1, \ldots, u^M} f(u^1, \ldots, u^M) \\
    \text{s.t. } u^i &\in U^i \quad \forall i = 1, \ldots, M,
\end{align*}
$$

where the function $f$ is smooth, convex and differentiable, whilst the convex sets $U^i$ are the cartesian product of the sets $U_i$ for $N$ times.

In the particular case when dynamics of the form (2) are considered in the MPC problem (4), the objective function $f$ in the convex optimization problem (13) has a sparse structure according to the interconnections between the subsystems. Indeed, by eliminating the states in (2), we get the following:

$$
x_{t+1}^i = A_i^i x_0^i + \sum_{t=0}^{N-1} (A_i^i)^{N-2-t} \left( B_i^i u_t^i + \sum_{j=N^i \setminus i} B_i^j u_t^j \right)
$$

which, by $u^i$ and $u_t^i$, can be expressed as:

$$
x^i = (A_i^i)^N x_0^i + A_N \text{diag}(B_i^i) u^i + A_N^\prime \sum_{j \in N^i \setminus i} \text{diag}(B_i^j) u^j.
$$

Thus, the state of the subsystem $i$ can, at any time be expressed as a function of initial state $x_0^i$ and input trajectories $u^i$ and $u_t^i$, with $j \in N^i \setminus i$. As a result, we can express the objective function of problem (13) as a sum of local functions with sparse structure:

$$
f(u^1, \ldots, u^M) = \sum_{i=1}^M f_i(u^i, j \in N^i).
$$

### A. Parallel Coordinate Descent Method

In this section we propose a block-coordinate descent based algorithm which permits optimization problem (13) to be solved distributively. We define the corresponding partition of the identity matrix:

$$
I = [E^1 \cdots E^M] \in \mathbb{R}^{Nm \times Nm},
$$

where $E^i \in \mathbb{R}^{Nm \times Nm}$, for all $i = 1, \ldots, M$. Thus the input trajectory $[(u^1)^T \cdots (u^M)^T]^T$ can be represented as:

$$
u = \sum_{i=1}^M E^i u^i.
$$

We define the partial gradient $\nabla_i f(u) \in \mathbb{R}^{Nm_i}$, of $f(u)$:

$$
\nabla_i f(u) = (E^i)^T \nabla f(u).
$$

We assume that the gradient of $f$ is coordinate-wise Lipschitz continuous with constants $L_i$ i.e:

$$
\|\nabla_i f(u + E^i h_i) - \nabla_i f(u)\| \leq L_i \|h_i\| \quad \forall h_i \in \mathbb{R}^{Nm_i},
$$

where $\|\| \|$ is the standard Euclidean norm.

Due to the assumption that $f$ is coordinate-wise Lipschitz continuous, it can be easily deduced that [13]:

$$
f(u + E^i h_i) \leq f(u) + \langle \nabla_i f(u), h_i \rangle + \frac{L_i}{2} \|h_i\|^2 \quad \forall h_i \in \mathbb{R}^{Nm_i}.
$$

### Remark 1:
Note that in the particular case of quadratic stage costs, the objective function in (13) is also quadratic, $f(u) = u^T H u + q^T u$, where $H$ is positive definite matrix due to the assumption that $R^i$ are positive definite. In this case, we find that $\nabla f(u) = H u + q$, thus the Lipschitz constants $L_i$ are $\|H^{ii}\| = \lambda_{\text{max}}(H^{ii})$, i.e the largest eigenvalue of the block matrix $H^{ii}$ and can be computed locally by each subsystem.

We now introduce the following norm for the extended space $\mathbb{R}^{Nm}$:

$$
\|u\|_1 = \sum_{i=1}^M L_i \|u^i\|^2,
$$

which will prove useful for estimating the rate of convergence for our algorithm. Additionally, if function $f$ is smooth, and strongly convex with regards to $\|\|_1$ with a parameter $\sigma$, then [14]:

$$
f(w) \geq f(v) + \langle \nabla f(v), w - v \rangle + \frac{\sigma}{2} \|w - v\|^2 \quad \forall v, w.
$$

Note that, if $f$ is strongly convex w.r.t the standard Euclidean norm $\|\|_2$ with a parameter $\sigma_0$, then $\sigma_0 \geq \sigma L_{\text{max}}$, where $L_{\text{max}} = \max L_i$. By taking $w = v + E^i h_i$ and $v = u$ in (17) we also get:

$$
f(u + E^i h_i) \geq f(u) + \langle \nabla_i f(u), h_i \rangle + \frac{\sigma L_i}{2} \|h_i\|^2,
$$

and combining with (15) we also deduce that $\sigma \leq 1$.

We now define the constrained coordinate update for our algorithm:

$$
v_i^*(u) = \arg \min_{v^i \in U^i} \left( \langle \nabla_i f(u), v^i - u^i \rangle + \frac{L_i}{2} \|v^i - u^i\|^2 \right),
$$

$$
u_i^*(u) = u + E^i (v^i(u) - u^i) \quad \forall i = 1, \ldots, M.
$$

The optimality conditions for the previous optimization problem are:

$$
\langle \nabla_i f(u) + L_i (v^i(u) - u^i), v^i - v^i(u) \rangle \geq 0, \quad \forall v^i \in U^i.
$$

Taking $v^i = u^i$ in the previous inequality and combining with (15) we obtain the following decrease in the objective function:

$$
f(u) - f(u^*_i(u)) \geq \frac{L_i}{2} \|v^i(u) - u^i\|^2.
$$

We now present our Parallel Coordinate Descent Method, that resembles the method in [19] but with a simpler iteration update and is a parallel version of the block-coordinate descent method from [13]:

### Algorithm PCDM($u_0$)

1) For $k \geq 0$, compute in parallel $u^i_k(u_k)$ for $i = 1, \ldots, M$.

2) Udpate:

$$
u_{k+1} = \sum_{i=1}^M \frac{1}{M} u^i_k(u_k)
$$

4483
Remark 2: Note that if the sets $U^i$ are simple (by simple we understand that the projection on this sets is easy) and the objective function has cheap coordinate derivatives, then computing $v^i(u)$ consists of projecting a vector on these sets and can be done numerically very efficient. For example, if these sets are simple box sets, i.e. $U^i = \{ v^i \in \mathbb{R}^{N_m^i} | v^i_{\min} \leq v^i \leq v^i_{\max} \}$, then the complexity of computing $v^i(u)$ from $\nabla_i f(u)$ is $O(N_m^i)$.

In turn, computing $\nabla_i f(u)$ has complexity $O(N_m mn_i)$ for quadratic functions, but when we have sparse hessians the complexity is much lower. In conclusion, Algorithm (PCDM) has usually a very low computing cost per iteration, compared to other existing methods, e.g. Jacobi type algorithm presented in [19], which usually require numerical complexity $O((Nm)^3)$ per iteration or even higher.

From (19) and convexity of $f$ we see immediately that the method (PCDM) decreases the objective function at each iteration:

$$f(u_{k+1}) \leq f(u_k) \forall k \geq 0. \quad (20)$$

The following theorem provides the convergence rate of the algorithm (PCDM) and employs the standard techniques for proving the rate of convergence of the projected gradient method [13], [14]:

**Theorem 2:** If function $f$ has a coordinate-wise Lipschitz continuous gradient with constants $L_i$ then algorithm PCDM has the following sublinear rate of convergence:

$$f(u_k) - f^* \leq \frac{M}{M+k} \left( \frac{1}{2} r_k^2 + f(u_0) - f^* \right).$$

Additionally, if $f$ is also strongly convex with regards to $\| \cdot \|_1$ with a constant $\sigma$, then the following linear rate of convergence is achieved for PCDM:

$$f(u_k) - f^* \leq \left( 1 - \frac{2\sigma}{M(1+\sigma)} \right)^k \left( \frac{1}{2} r_k^2 + f(u_0) - f^* \right).$$

**Proof:** We introduce the following term:

$$r_k^2 = \| u_k - u_* \|_1^2 = \sum_{i=1}^M L_i \langle u_k^i - u_*^i, u_k^i - u_*^i \rangle,$$

where $u_*$ is the optimal solution of (13) and $u_*^i = E^i u_*$. For the next iterate we obtain:

$$r_{k+1}^2 = \sum_{i=1}^M L_i \left( \frac{1}{M} v^i(u_k) + (1 - \frac{1}{M}) u_*^i - u_*^i \right)^2$$

$$= r_k^2 + \sum_{i=1}^M L_i \left( \frac{1}{M} v^i(u_k) - u_*^i \right)^2 +$$

$$\quad \left( \frac{2L_i}{M} \langle v^i(u_k) - u_*^i, v^i(u_k) - u_*^i \rangle \right)^2$$

$$\leq r_k^2 + \sum_{i=1}^M L_i \left( \frac{1}{M} \right)^2 \left( \| v^i(u_k) - u_*^i \| \right)^2 +$$

$$\quad \left( \frac{2L_i}{M} \langle v^i(u_k) - u_*^i, v^i(u_k) - u_*^i \rangle \right)^2$$

$$\leq r_k^2 + \sum_{i=1}^M \left( \frac{2L_i}{M} \right)^2 \left( \| v^i(u_k) - u_*^i \| \right)^2 +$$

$$\quad \frac{2}{M} \langle \nabla_i f(u_k), u_*^i - v^i(u_k) \rangle$$

$$\leq r_k^2 + \frac{2}{M} \langle \nabla_i f(u_k), u_*^i - v^i(u_k) \rangle$$

$$\leq r_k^2 + \frac{2}{M} \langle \nabla_i f(u_k), u_*^i - v^i(u_k) \rangle$$

By convexity of $f$ and (15) we get:

$$r_{k+1}^2 \leq r_k^2 - 2 \langle f(u_{k+1}) - f(u_k) \rangle +$$

$$\quad \frac{2}{M} \langle \nabla f(u_k), u_* - u_k \rangle. \quad (21)$$

Adding up these inequalities and using again convexity of $f$ we obtain:

$$\frac{1}{2} r_k^2 + f(u_k) - f^* \geq \frac{1}{2} r_{k+1}^2 + f(u_{k+1}) - f^* \geq \left[ \frac{1}{2} + \frac{1}{M} \sum_{j=0}^k (f(u_j) - f^*) \right] \geq f(u_k) - f^* + \frac{k}{M} \sum_{j=0}^k (f(u_j) - f^*).$$

Taking into account that our algorithm is a descent algorithm, i.e. $f(u_j) \geq f(u_{k+1})$ for all $j = 0, \ldots, k$, from the previous inequality we obtain the first part of the theorem.

For the strongly convex case, we take $w = u^*$ and $v = u_k$ in (17) and through (21) we get:

$$\frac{1}{2} r_k^2 + f(u_k) - f^* \leq \frac{1}{2} r_{k+1}^2 + f(u_{k+1}) - f^* - \frac{1}{M} (f(u_k) - f^* + \frac{\sigma r_k^2}{2}).$$

From the strong convexity of $f$ in (17) we also get:

$$f(u_k) - f^* + \frac{\sigma r_k^2}{2} \geq \sigma r_k^2.$$

We now define $\gamma = \frac{2\sigma}{1+\sigma} \in [0, 1]$ and using the previous inequality we obtain the following result:

$$f(u_k) - f^* + \frac{\sigma r_k^2}{2} \leq \gamma (f(u_k) - f^* + \frac{\sigma r_k^2}{2}) + (1 - \gamma) \sigma r_k^2.$$

Using this inequality in (22) we get:

$$\frac{1}{2} r_k^2 + f(u_k) - f^* \leq (1 - \frac{1}{M}) \left( \frac{1}{2} r_k^2 + f(u_k) - f^* \right).$$

Applying this inequality iteratively, we obtain the following convergence result for $k \geq 0$:

$$\frac{1}{2} r_k^2 + f(u_k) - f^* \leq (1 - \frac{1}{M})^k \left( \frac{1}{2} r_0^2 + f(u_0) - f^* \right),$$

and by replacing $\gamma = \frac{2\sigma}{1+\sigma}$ we obtain the second part of the theorem. ■

The following properties are immediate:

**Lemma 1:** The following properties follows immediately for our PCDM algorithm:

(i) Given any initial guess $u_0$, the iterates of the PCDM are feasible at each iteration, i.e. $(u_k^i \in U^i$ for all $k$).

(ii) The cost function $f$ is nonincreasing, i.e. $f(u_{k+1}) \leq f(u_k)$ according to (20).

(iii) The rate of convergence of PCDM algorithm is given in Theorem 2.
B. Stability of the distributed MPC scheme

The parallel coordinate descent Algorithm (PCDM) presented in this paper is used to solve distributively the MPC problem (4). We denote the approximate solution of the Algorithm (PCDM) after certain number of iterations with $u^{CD}_t$. In this case we can consider the cost function $V(x,u^{CD})$ as a Lyapunov function, using the standard theory for suboptimal control (see e.g. [7], [19], [20] for similar approaches). Note that we can consider that at each MPC step the optimization Algorithm (PCDM) is initialized (warm start) with the shifted sequence of controllers obtained at the previous step and concatenated with the feedback controller $\kappa(\cdot)$ computed in Section II such that (5) is satisfied and we denote this control sequence by $(u^{CD}_t)^\dagger$. Then, from the properties of Algorithm (PCDM) (see Lemma 1) combined with inequality (5) we get:

$$ V(x^+,u^{CD}) \leq V(x^+,\tilde{x}_N) \leq V(x,u^{CD}) - \ell(x,u_0^{CD}) $$

for all $x \in X_N$, where $X_N$ is a domain of attraction of the controlled system defined appropriately (see [7], [20] for more details). Thus, this MPC scheme ensure stability for the controlled system.

Furthermore, given the fact that $\nabla f(u) = (E^i)^T \nabla f(u)$, we can perform distributed computations in our MPC scheme since only local information is required by the Algorithm (PCDM). For example, if model (1) is used, the Hessian $H$ of $f$ has a sparse structure, according to the dynamics (1). In particular, for systems given by (2), then $\nabla f(u) = (E^i)^T (Hu + q^T u) = \sum_{j \in N_i} H^j u^j + q^j$, such that only the blocks $H^j$ of $H$ which concern $j \in N_i$ are used.

IV. Conclusions

In this paper we have proposed a distributed optimization algorithm to solve MPC problems for general linear systems comprised of interconnected subsystems. The new optimization algorithm is based on the block coordinate descent framework but with very simple iteration complexity and parallel computations using only local information. We have shown that for smooth objective functions it has sublinear rate of convergence, while for strongly convex objective functions it converges linearly. An MPC controller based on this distributed optimization method was derived, for which every subsystem in the network can compute feasible and stabilizing control inputs using distributed computations. An analysis for obtaining a stabilizing linear control law from a distributed viewpoint was made that provides local terminal costs which guarantees stability of the closed-loop interconnected system. Implementation results on a programmable logic controller for a 4 tank process are reported in [16].

Acknowledgements: The research leading to these results has received funding from: the European Union, Seventh Framework Programme (FP7/2007–2013), EMBOCON, under grant agreement no 248940; CNCS-UEFISCDI (project TE, no. 19/11.08.2010); ANCS (project PN II, no. 80EU/2010); Sectoral Operational Programme Human Resources Development 2007–2013 of the Romanian Ministry of Labor, Family and Social Protection through the Financial Agreements POSDRU/81/1.5/S/62557.

REFERENCES


