Optimality condition decomposition approach to distributed model predictive control
Joseph-Julien Yamé, Farah Gabsi, Tejaswinee Darure, Tushar Jain, Frédéric Hamelin, Nathalie Sauer

To cite this version:
Joseph-Julien Yamé, Farah Gabsi, Tejaswinee Darure, Tushar Jain, Frédéric Hamelin, et al.. Optimality condition decomposition approach to distributed model predictive control. American Control Conference, ACC 2019, Jul 2019, Philadelphia, PA, United States. hal-02288131

HAL Id: hal-02288131
https://hal.archives-ouvertes.fr/hal-02288131
Submitted on 13 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Abstract—This paper presents a new methodology for distributed model predictive control of large-scale systems. The methodology involves two distinct stages, i.e., the decomposition of large-scale systems into subsystems and the design of subsystem controllers. Two procedures are used: in the first stage, the structure of the Karush-Kuhn-Tucker matrix resulting from the necessary optimality conditions is exploited to yield a decomposition of the large-scale system into several subsystems. In the second stage, a particular technique, the so-called optimality condition decomposition makes it possible to synthesize distributed coordinated subcontrollers thus achieving an optimal distributed control of the large-scale system. The convergence of the proposed approach is stated.

I. INTRODUCTION

Complexity in controlling large scale systems like power plants, network flow and multizone buildings motivates the development of different architectures for distributed model predictive control (DMPC) [17], [14], [11]. This complexity may arise due to issues like the decomposition of large scale systems into subsystems and the coordination between controllers applied to each subsystem. Several architectures and decomposition methods are proposed in the literature to address the above issues. Siljak [1] summarized various large-scale system decomposition methods and introduced overlapping and nonoverlapping decomposition models based on sharing variables between the subsystems. Methods using bipartite graph theory, nested decomposition and the inclusion principle are widely practiced to partition a large-scale system into subsystems [2] [3]. Other interesting approaches like relative gain array matrix [5] and Grammians [6] are based on input-output mappings depending on the interaction between subsystems. In some cases, due to structural properties of the large scale system, it is straightforward to derive subsystem models by system identification, instead of partitioning the centralized large-scale model. The design of subsystem controllers and their coordination are closely related to the degree of interaction between the subsystems [8]. Clearly, if the interaction between subsystems is weak then each subsystem may be controlled independently without any coordination between subsystem controllers and this yields a decentralized structure [3]. On the other hand, if the interaction between subsystems is strong then coordination between subsystem controllers may improve the overall performance of the large-scale system. These coordination strategies may differ depending on the several ways the information is exchanged between controllers and this leads to noncooperative and cooperative strategies [7]. In noncooperative distributed model predictive structures, a subsystem controller optimizes the model predictive control problem locally using information of other subsystems and it reaches a Nash equilibrium. In cooperative DMPC architecture, the subsystem controller performs its optimization by considering the global objectives and it achieves pareto optimal solution [8]. The coordination can be done in a hierarchical structure [9], where a master level optimization problem uses the shared variables or constraints. The optimal solutions are sent as coordination parameters to the lower level consisting of subsystem controllers. Distributed model predictive problem may also be viewed as partitioning a centralized model predictive problem and such view is motivated by the decomposition methods of large-scale convex optimization problems as e.g. Dantzig Wolfe decomposition, Benders decomposition [10] or primal and dual decomposition techniques [11] [12]. In this paper, we address the DMPC problem for large-scale plants using the above latter view and we assume that the centralized model predictive control problem for the targeted large-scale plant is available through the knowledge of the cost functional and the overall system dynamics as well as the physical limitations on actuators expressed respectively through equality constraints and inequality constraints on the system variables. We propose a novel approach to distributed model predictive control of large-scale plants that addresses system decomposition and the design of distributed coordinated subsystem controllers in two distinct steps. In the first step related to system decomposition into subsystems, we formulate the Karush-Kuhn-Tucker (KKT) system of the centralized model predictive control problem for the targeted large-scale plant is available through the knowledge of the cost functional and the overall system dynamics as well as the physical limitations on actuators expressed respectively through equality constraints and inequality constraints on the system variables. We propose a novel approach to distributed model predictive control of large-scale plants that addresses system decomposition and the design of distributed coordinated subsystem controllers in two distinct steps. In the first step related to system decomposition into subsystems, we formulate the Karush-Kuhn-Tucker (KKT) system of the centralized model predictive control problem (CMPC). This KKT system is carefully analyzed and transformed in an efficient manner to exhibit subsystems of a large-scale system. The resulting KKT subsystems provide the basis of the decomposed subproblems. In the second step, the design of distributed coordinated subsystem controllers is performed based on the so-called optimality condition decomposition (OCD) method [13]. The coordination between subcontrollers is inherited from the OCD and thus the scheme does not need any hierarchical coordination layer.

The paper is organized as follows. The statement of the DMPC problem and the optimization model are given in section 2. In section 3 we present the system decomposition
method and in section 4, based on the obtained subsystems we derive the synthesis of coordinated subsystem controllers through the optimality condition decomposition. A simple example is provided in section 5 to illustrate the steps from system decomposition to the synthesis of decision variables.

II. PROBLEM SETTING

Consider a discrete-time state space representation for a large-scale system given by

\[ z(k+1) = Az(k) + Bu(k) + Gd(k) \]  

(1)

where \( z(k) \in \mathbb{R}^n_z \), \( u(k) \in \mathbb{R}^n_u \) and \( d(k) \in \mathbb{R}^n_d \) are the state, control input and disturbance vectors of the system, \( A, B, G \) are matrices with appropriate dimension and where \( k \) denotes the discrete time index. The classical formulation of the optimization problem for centralized model predictive control of the above large scale system reads as

\[
\begin{align*}
\min_{U_k} & \quad J(U_k, z(k)) \\
\text{s.t.} & \quad z(k+j+1|k) = Az(k+j|k) + Bu(k+j|k) + Gd(k) \\
& \quad z_{\min} \leq z(k+j|k) \leq z_{\max} \\
& \quad u_{\min} \leq u(k+j) \leq u_{\max} \\
& \quad z(k|k) = z(k)
\end{align*}
\]  

(2)

with index \( j = 0, 1, \ldots, N-1 \) and where \( J \) is the overall cost function, \( U_k = [u(k|k), \ldots, u(k+N-1|k)] \) the predicted control input sequence, \( \{d(k|k), \ldots, d(k+N-1|k)\} \) the forecast of the disturbances, \( z(k) \) the current measured state and \( N \) the prediction horizon. It is well-known that this centralized model predictive control achieves the best attainable solution [7], but its computational burden and the dependability issues due to its centralized structure are major drawbacks for its practical implementation. These drawbacks have motivated different schemes for distributed control architectures where the plant-wide system is decomposed into several interconnected subsystems with each subsystem controlled locally by a controller. These local controllers are interconnected by a digital communication network and coordinate their control signals through information exchange over the network. The design of distributed control structures proceeds generally through the decomposition of a large-scale plant into subsystems followed by the design and coordination between subsystem controllers. We propose an approach to tackle the above steps in designing distributed model predictive control (DMPC) schemes. This DMPC takes as a reference the centralized model predictive control and therefore our starting point will be the centralized formulation of the MPC problem. To make the notation less cluttered and the fundamentals of our results easier to understand, rewrite the centralized problem (2) into the equivalent general mathematical form:

\[
\begin{align*}
\mathcal{P}_c : \quad & \min_{x} \ f(x) \\
\text{s.t.} & \quad h(x) = 0 \\
& \quad x_{\min} \leq x \leq x_{\max}
\end{align*}
\]  

(3)

where \( x \in \mathbb{R}^{N(n_z+n_u)} \) is the vector of the optimization variables, \( f \) is the objective function and \( h = [h_1, \ldots, h_{n_h}]^T \) is the vector of \( n_h \) equality constraints representing the dynamics of the system. Specifically, we define the optimization variable by vector \( x \):

\[
\begin{align*}
x = [u^T(0|0), z^T(1|0), u^T(1|0), z^T(2|0), \ldots, \\
\ldots, u^T(N-1|0), z^T(N|0)]^T
\end{align*}
\]  

(4)

and we denote by \( n \) the dimension of \( x \), i.e., \( n = N(n_z+n_u) \). The bounds on the control variables and states translates as

\[
\begin{align*}
x_{\min} = 1_N \otimes \begin{bmatrix} (u_{\min})^T & (z_{\min})^T \end{bmatrix}^T \\
x_{\max} = 1_N \otimes \begin{bmatrix} (u_{\max})^T & (z_{\max})^T \end{bmatrix}^T
\end{align*}
\]  

(5)

where \( 1_N \) is the vector of dimension \( N \) whose components are all equal to 1, and the symbol \( \otimes \) represents the Kronecker product. In terms of the decision variable \( x \), the system dynamics in (2) is now described by the equality constraint

\[
h(x) = \mathcal{A} x - b = 0
\]  

(6)

where the \( (Nn_z \times n) \) matrix \( \mathcal{A} \) and the \( Nn_z \)-dimensional vector \( b \) are given by

\[
\mathcal{A} = \begin{pmatrix}
B & -I_{n_z} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & A & B & -I_{n_z} & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & A & B & -I_{n_z} & 0 \\
0 & 0 & \cdots & 0 & 0 & A & B & -I_{n_z} & 0
\end{pmatrix}
\]  

(7)

\[
b = \begin{pmatrix}
Az(0) + Gd(0|0) \\
Gd(1|0) \\
\vdots \\
Gd(N-1|0)
\end{pmatrix}
\]  

(8)

Throughout, we assume that the forecast of disturbances is available and consequently vector \( b \) is a known vector. Furthermore, the following technical hypothesis is assumed for problem \( \mathcal{P}_c \):

Assumption 1: The cost function \( f \) is convex and twice differentiable

Now, we aim at deriving the distributed model predictive control from the above centralized problem \( \mathcal{P}_c \). We proceed in two steps: i)- decomposition of vector \( x \) into \( p \)-subvectors \( \tilde{x}_i, (i = 1, 2, \ldots, p) \) that highlight \( p \) interconnected subsystems ii)- design of coordinated distributed \( p \) sub-controllers.

III. SYSTEM DECOMPOSITION

A. The Karush-Kuhn-Tucker matrix of equality constrained optimization problems

The Lagrange function for problem \( \mathcal{P}_c \) (3) is given by

\[
\mathcal{L}(x) = f(x) + \lambda^T h(x)
\]  

(9)

where \( \lambda = [\lambda_1, \ldots, \lambda_{n_h}]^T \) is the vector of Lagrange multipliers with \( n_h = Nn_z \). As the bounds on the variables will not affect the decomposition, we will ignore the associated inequalities. The search for a KKT (Karush-Kuhn-Tucker)
point to problem $\mathcal{P}_c$ comes down to solving the system of $n + nh$ equations in $n + nh$ unknowns $(x, \lambda)$

$$\nabla L(x, \lambda) = \begin{pmatrix} \nabla_x L(x, \lambda) \\ \nabla_\lambda L(x, \lambda) \end{pmatrix} = \begin{pmatrix} \nabla f(x) + J_h^T(x) \lambda \\ h(x) \end{pmatrix} = 0$$ (10)

where $J_h(x)$ is the Jacobian of $h(x)$ and $\nabla_x L(x, \lambda)$ and $\nabla_\lambda L(x, \lambda)$ are the gradients of $L'$ with respect to $x$ and $\lambda$. The nonlinear equation (10) in $(x, \lambda)$ can be solved iteratively using Newton’s method ([4],[15],[19]) from a starting guess $(x^0, \lambda^0)$. The linearized system to be solved at iteration $k$ for the Newton step $s^k = x^{k+1} - x^k$, $\delta^k = \lambda^{k+1} - \lambda^k$ is

$$\mathcal{K}_c(x^k, \lambda^k) \begin{pmatrix} s^k \\ \delta^k \end{pmatrix} = -\begin{pmatrix} \nabla f(x^k) + J_h^T(x^k) \lambda^k \\ h(x^k) \end{pmatrix}$$ (11)

where matrix

$$\mathcal{K}_c(x, \lambda) = \begin{pmatrix} \nabla_x^2 L(x, \lambda) & J_h^T(x) \\ J_h(x) & 0 \end{pmatrix}$$ (12)

is defined as the KKT matrix of the centralized constrained optimization problem (3) with $\nabla_x^2 L$ being the Hessian matrix of $L'$ with respect to variable $x$. Note that the KKT matrix is symmetric but not necessarily positive definite. A close look at the KKT matrix of the centralized constrained optimization problem reveals important information about the problem and system structures. Indeed, the left-upper block $\nabla_x^2 L$ reveals the separability of the cost function with respect to the decision variable $x$. If $\nabla_x^2 L$ shows up as a block-diagonal matrix, then the cost function has a separability property in the sense that it can be expressed as the sum of functions of sub-vectors of the decision vector $x$. As for block $J_h(x)$, it is really nothing else than the sensitivity matrix of the system dynamics with respect to vector $x$. Therefore, the off-diagonal coefficients in this matrix represent the degree of coupling between the original components of vector $x$. It turns out that this matrix reveals some structural information about the system dynamics and in case this matrix is sparse, this makes it convenient to identify groups of variables in weak or strong interaction.

B. Transformation of the centralized KKT matrix

Starting from the centralized KKT matrix $\mathcal{K}_c$ in (12), we note that it is always possible to transform it, through some rows/columns permutation, into the following equivalent form

$$\mathcal{K}_c \rightarrow \mathcal{K}_d \Delta = \begin{pmatrix} KKT_1 & * & \cdots & * \\ * & KKT_2 & \cdots & * \\ \vdots & \vdots & \ddots & \vdots \\ * & * & \cdots & KKT_p \end{pmatrix}$$ (13)

where the symbol $\sim$ stands for equivalent and the matrix $\mathcal{K}_d$ is by definition the matrix displayed in (13). By the very transformation of matrix $\mathcal{K}_c$ into the form (13), it appears $p$ block-matrices on the diagonal of the transformed matrix $\mathcal{K}_d$ and these block-matrices have the structure of KKT matrices. These block-matrices $KKT_i$ ($i = 1, \ldots, p$) may be viewed as those of some $p$ subsystems $(S_1, S_2, \ldots, S_p)$ composing the overall system so that the original decision vector $x$ is fully equivalent to vector $\tilde{x} = [\tilde{x}_1, \ldots, \tilde{x}_p]^T$ where $\tilde{x}_j$ is a subvector which is the decision vector of subsystem $S_j$.

The $(\ast)$-blocks in (13) denote sparse block-matrices which bear the interaction between the subsystems. If these sparse block-matrices are identically zero matrices then matrix $\mathcal{K}_c$ reduces to a block-diagonal matrix which may be regarded as the KKT matrix of a decentralized structure. Looking closely at (13), the equivalent centralized form may be viewed as that of a distributed structure in which $p$ subsystems are clearly exhibited with interactions between these subsystems achieved through the $(\ast)$-blocks. It is thanks to this viewpoint that we denoted this matrix in (13) as $\mathcal{K}_d$ where the subscript $d$ refers to a distributed structure. It is worth noticing that transforming the centralized KKT matrix into an equivalent distributed KKT matrix as in (13) achieves the goal of decomposing problem (3) on the dynamics/constraints level as well as on the control level. In the literature, various methods describe the transformation of a symmetric matrix into a block-diagonal matrix form.

For example, Dulmage-Mendelsohn [16] technique or the reachability matrix technique [1] are efficient methods in bipartite graph theory. But due to the large size of the KKT matrix resulting from the centralized MPC problem (2), the complexity of bipartite graph methods increases significantly. Moreover, for computational purposes, these methods set to one the non-zero matrix coefficients and transform the KKT matrix into an equivalent binary matrix. This may undermine the coupling information between the variables in the system. In the nested $\epsilon$ decomposition method [2], matrix coefficients that are less than $\epsilon$ are replaced by zeros and the resulting matrix is reordered into a block-diagonal form. This procedure is usually carried out iteratively by augmenting $\epsilon$ at each iteration till the block-diagonal form is achieved. Clearly, this method compromises the coupling information between variables due to the elimination of some matrix coefficients. To avoid such loss of information, we suggest reducing the bandwidth of the centralized KKT matrix by permuting its rows and columns such as to move all the nonzero elements of $\mathcal{K}_c$ in a band as close as possible to the diagonal. This permutation can be achieved using the sparse reverse Cuthill-McKee (CM) algorithm [18] and the decomposition of the system into subsystems proceeds through the following algorithm:

**Algorithm 1 KKT matrix based system decomposition**

**Require:** Input data : $h = [h_1, \ldots, h_n]^T; x = [x_1, \ldots, x_n]^T; f(x)$

**Ensure:** Result $\tilde{x}_i, (i = 1, \ldots, p)$

1. Derive the KKT matrix $\mathcal{K}_c$ for problem $\mathcal{P}_c$
2. Transform $\mathcal{K}_c$ into $\mathcal{K}_d$ using sparse reverse Cuthill-McKee algorithm
3. Identify separable blocks from KKT matrix $\mathcal{K}_d$
4. Determine the sub-vectors $\tilde{x}_i, (i = 1, \ldots, p)$ based on the identified separable blocks

Note that for revealing the separable blocks, it is sufficient
to set $N = 2$ (prediction horizon) in deriving the KKT matrix $K_c$.

IV. DISTRIBUTED SUBCONTROLLERS SYNTHESIS

A. Control synthesis procedure

With the provided system decomposition method, one can successfully partition a large-scale system into $p$-subsystems. Now, raises the issue of synthesizing the controllers for the identified $p$-subsystems in a coordinated way. The main idea and tool used to address this issue is an optimization procedure, the so-called optimality condition decomposition ([13]), which can be viewed as a modified Lagrangian relaxation. Thanks to the decomposition of $x$ into $p$-subvectors, i.e., $x \sim \bar{\bar{x}} = [\bar{x}_1^T, \ldots, \bar{x}_p^T]^T$, the centralized problem (3) can be written explicitly as

$$\begin{align*}
\mathcal{P}_c : \quad & \min_{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p} f(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) \\
& \text{subject to} \\
& \tilde{h}_1(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) = 0 \\
& \vdots \\
& \tilde{h}_p(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) = 0 \\
& x_i^{\min} \leq \bar{x}_i \leq x_i^{\max} \quad (i = 1, 2, \ldots, p)
\end{align*}$$

The constraint $\tilde{h}_i(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) = 0$ describes the overall system dynamics and is nothing more than the original constraint $h(x_1, \ldots, x_n)$ expressed as a function of the subvectors of the $p$ subsystems of the decomposition. This constraint naturally decomposes into $p$ vector-valued constraints describing the dynamics of the $p$ subsystems $S_i(i = 1, \ldots, p)$ and how each subsystem $S_i$ interacts with other subsystems $S_j(j \neq i)(j = 1, \ldots, p)$. The optimization problem (14) reads therefore as

$$\begin{align*}
\mathcal{P}_c : \quad & \min_{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p} f(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) \\
& \text{subject to} \\
& \tilde{h}_1(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) = 0 \\
& \vdots \\
& \tilde{h}_p(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p) = 0 \\
& x_i^{\min} \leq \bar{x}_i \leq x_i^{\max} \quad (i = 1, 2, \ldots, p)
\end{align*}$$

where the $\tilde{h}_i's$, $i = 1, \ldots, p$, are $n_h(i)$-dimensional vector-valued constraints with $\sum_{i=1}^p n_h(i) = n_h$. These $p$ vector-valued constraints $h_i(i = 1, 2, \ldots, p)$ are actually coupling constraints that if they were removed, the resulting optimization problem would be easier to solve. The iterative procedure underlying the optimality condition decomposition for solving problem (15) is to enforce “some separable approximation” for both $f$ and $h_i(i = 1, 2, \ldots, p)$ by fixing some of the decision variables in these functions to their last computed values (indicated by overlining) to obtain the following $p$ subproblems solved in parallel:

$$\begin{align*}
\mathcal{P}_i : \quad & \min_{\bar{x}_i} f(\bar{x}_i, \ldots, \bar{x}_i, \bar{x}_{i+1}, \ldots, \bar{x}_p) \\
& \text{subject to} \\
& \tilde{h}_1(\bar{x}_1, \ldots, \bar{x}_{i-1}, \bar{x}_i, \bar{x}_{i+1}, \ldots, \bar{x}_p) = 0 \\
& x_i^{\min} \leq \bar{x}_i \leq x_i^{\max} \quad (i = 1, 2, \ldots, p)
\end{align*}$$

In (16), $\lambda_j(j = 1, \ldots, p)$ denotes the dual variable (Lagrange multiplier) associated to the constraint $\tilde{h}_j$ and fixed at its last computed value in an iterative scheme. Note that the procedure looks like a modified Lagrangian relaxation, but it has the computational advantage that a single iteration for each subproblem $\mathcal{P}_i$ can be performed before updating all the variables. Clearly, the benefits of this formulation rely in removing the coupling in the constraints from the original large-scale problem. Let $\bar{x}^* = [\bar{x}_1^*, \ldots, \bar{x}_p^*]^T$ denotes the vector of optimal decision variables of the $p$ subproblems (16), then the steps of this distributed control synthesis procedure can be summarized in the following algorithm:

Algorithm 2 OCD-based distributed control synthesis

Require: Input data: $f; \tilde{h} = [\tilde{h}_1, \ldots, \tilde{h}_p]^T; x^{\min}, x^{max}$. Initial guess $\bar{x}_i(i = 1, \ldots, p)$

Ensure: Result $\bar{x}^* = [\bar{x}_1^*, \ldots, \bar{x}_p^*]^T$

1) Solve in parallel the optimization problems $\mathcal{P}_i, (i = 1, \ldots, p)$ given in (16) and obtain primal and dual search directions (the Newton step) $\delta_i$ and $\delta_i$ for the given iteration

2) Update the primal and dual variables: $\bar{x}_i = \bar{x}_i + \delta_i$;

3) Go to step 6 if variables do not change significantly in two consecutive iterations, otherwise continue

4) Broadcast the updated variables on the communication network

5) Update the coupling constraints in (16) and go to step 1

6) Implement $\bar{x}^* \leftarrow [\bar{x}_1^*, \ldots, \bar{x}_p^*]^T$}

B. Convergence analysis

Observe that, by the structure of problems $\mathcal{P}_i$ in (16) built upon fixing all variables but the $i^{th}$ decision subvector $\bar{x}_i$, the original problem has been decomposed to yield a separable problem that approximates it. Indeed, problem $\mathcal{P}_i$ can be rewritten as

$$\begin{align*}
\mathcal{P}_i : \quad & \min_{\bar{x}_i} \phi_i(\bar{x}_i) \\
& \text{subject to} \\
& \tilde{f}_i(\bar{x}_i) = 0 \\
& x_i^{\min} \leq \bar{x}_i \leq x_i^{\max} \quad (i = 1, 2, \ldots, p)
\end{align*}$$

by setting

$$\phi_i(\bar{x}_i) = f(\bar{x}_1, \ldots, \bar{x}_{i-1}, \bar{x}_i, \bar{x}_{i+1}, \ldots, \bar{x}_p)$$

and

$$\tilde{f}_i(\bar{x}_i) = \tilde{h}_i(\bar{x}_1, \ldots, \bar{x}_{i-1}, \bar{x}_i, \bar{x}_{i+1}, \ldots, \bar{x}_p) = 0$$
The optimization problems (17) are thus identical to the following overall optimization problem
\[
\begin{align*}
\min_{\tilde{x}_1, \ldots, \tilde{x}_p} & \quad \sum_{i=1}^p \phi(\tilde{x}_i) \\
\text{s.t.} & \quad \tilde{f}_i(\tilde{x}) = [\tilde{f}_i^T(\tilde{x}_1), \ldots, \tilde{f}_i^T(\tilde{x}_p)]^T = 0
\end{align*}
\]
(20)

Let \( \mathcal{X}_i \) be the KKT matrix of the optimization problem \( \mathcal{P}_i \) for \( i = 1, \ldots, p \) in (17), then the KKT matrix of the overall problem (20) is the diagonal matrix
\[
\tilde{\mathcal{X}} = \begin{pmatrix}
\mathcal{X}_1 & 0 & \cdots & 0 \\
0 & \mathcal{X}_2 & 0 & \vdots \\
\vdots & 0 & \ddots & 0 \\
0 & \cdots & 0 & \mathcal{X}_p
\end{pmatrix}
\]
(21)
The Newton steps in solving iteratively optimization problem (20) are given by the solution of the system of linear equations (see (11))
\[
\tilde{\mathcal{X}}_k \pi^k = -\gamma^k
\]
(22)
where \( \pi^k = (\tilde{\mathcal{X}}^{k+1} - \tilde{\mathcal{X}}^k) / \tilde{\mathcal{X}}_k^2 \) is the Newton step at iteration \( k \), matrix \( \mathcal{K}_k \) is the KKT matrix (21) evaluated at \( \xi^k = (\tilde{\mathcal{X}}^k)^T, (\tilde{\mathcal{X}}^k)^T \) and
\[
\gamma^k = \left( \nabla \phi(\tilde{x}^k) + J_\lambda(\tilde{x}^k) \tilde{\lambda} \right)
\]
(23)

Now, let \( x^* \) denotes the optimal solution of the centralized problem (3) and let \( \xi^* = [x^T, \lambda^*]^T \) be the corresponding primal-dual point. For this optimal point, we make the following assumptions:

**Assumption 2:** The Jacobian of the constraints, \( \nabla h \), has full row rank at \( x^* \).

**Assumption 3:** Matrices \( \mathcal{K}_k \) are non-singular for any \( \xi^* \) and the sequence \( \{ \mathcal{K}_k \} \) converges to a non-singular matrix \( \mathcal{K}^* = \lim_{\xi \to \xi^*} \mathcal{K}_k \).

Let denote by \( \mathcal{K}_d \) the evaluation of KKT matrix (13) at the optimal primal-dual point \( \xi^* \), then clearly \( \mathcal{K}_d \pi^* = -\gamma^* = 0 \). We are now in a position to state the convergence result in the following proposition.

**Proposition 1:** Under Assumptions 1, 2, and 3, if at the optimal point \( \xi^* \) of the centralized problem (3) it holds that
\[
\left\| I - (\mathcal{K}^*)^{-1} \mathcal{K}_d \right\| < 1
\]
(24)
then the procedure used in Algorithm 2 converges globally to \( x^* \).

**Proof:** Let \( \xi^k \) be an iterate in a neighborhood of \( \xi^* \), then under Assumption 1 the following Taylor series expansion holds
\[
\gamma^k = \gamma^* + \mathcal{K}_d (\xi^k - \xi^*) + \text{higher order terms in } (\xi^k - \xi^*)
\]
(25)
with \( \gamma^* = 0 \). Since \( \xi^{k+1} = \xi^k + \pi^k \) and \( \pi^k = -A_k^{-1} \gamma^k \), we have
\[
\xi^{k+1} - \xi^* = (I - (\mathcal{K}^*)^{-1} \mathcal{K}_d) (\xi^k - \xi^*) + \text{h.o.t in } (\xi^k - \xi^*)
\]
(26)
which implies that
\[
\left\| \xi^{k+1} - \xi^* \right\| \leq \left\| I - (\mathcal{K}^*)^{-1} \mathcal{K}_d \right\| \left\| \xi^k - \xi^* \right\| + \text{h.o.t in } (\xi^k - \xi^*)
\]
(27)
The convergence to zero of the sequence \( \{ \xi^k - \xi^* \} \) results from the above inequality with the norm condition (24) ■

**V. A SIMPLE ILLUSTRATIVE EXAMPLE**

The above distributed model predictive control is being applied for controlling a large-scale building benchmark, but for lack of space in this paper this benchmark will not be presented here and will be reported elsewhere. Nevertheless, to grasp the presented approach, we present below a very simple example which can be solved manually to illustrate the decomposition technique of a system and the synthesis of the optimal distributed decisions. Consider the following optimization problem
\[
\mathcal{P}_c : \quad \min_{x} f(x)
\]
(28)
s.t. \( h(x) = 0 \)
with
\[
f(x) = 2x_1^2 + x_2^2 - 20x_1 - 16x_2
\]
(29)
\[
h(x) = \left( \begin{array}{c}
x_1^2 + x_2^2 - 12.5572 \\
x_1 + x_2 - 5
\end{array} \right) = 0
\]
(30)
Note that the constraint \( h(x) \) is nonlinear here. The centralized solution of the problem is easily computed as
\[
x^* = \left( \begin{array}{c}
2.6692 \\
2.3309
\end{array} \right)
\]
(31)
The system decomposition is trivial here but we will still illustrate it with algorithm 1. We first derive the KKT matrix \( \mathcal{K}_c \) from the Lagrangian of the problem
\[
\mathcal{K}_c = \left( \begin{array}{cc}
4 + 2\lambda_1 & 0 \\
0 & 2(1 + \lambda_1)
\end{array} \right)
\]
(32)
\[
\begin{pmatrix}
2x_1 & 1 \\
2x_2 & 1
\end{pmatrix}
\]

Using the sparsity pattern command of MATLAB, "spy", we visualize the structure of \( \mathcal{K}_c \) in figure 1 (left image). This KKT matrix is then transformed into an equivalent KKT matrix \( \mathcal{K}_d \) with reduced bandwidth as displayed in figure 1 (right image) by the reverse Cuthill-McKee reordering algorithm using MATLAB command "symrcm". The resulting permutation vector is \( p = [1432] \). The explicit expression of KKT matrix \( \mathcal{K}_d \) is
\[
\mathcal{K}_d = \left( \begin{array}{cc}
4 + 2\lambda_1 & 2x_1 \\
2x_1 & 0
\end{array} \right)
\]
(33)
\[
\begin{pmatrix}
2x_2 & 0 \\
0 & 2(1 + \lambda_1)
\end{pmatrix}
\]
Two blocks are trivially identified here with the new decision vector \( \tilde{x} \) consisting of subvectors (here scalars) \( \tilde{x}_1 = x_1 \) and
Fig. 1. Original KKT matrix $\mathcal{K}_c$ and its equivalent reordering $\mathcal{K}_d$

\[ \tilde{x}_2 = x_2, \text{ that is, } \tilde{x} \text{ equals to the original vector } x = [x_1 \, x_2]^T. \]

The associated subsystems are $\tilde{h}_1(\tilde{x}) = h_1(x)$ for subvector $\tilde{x}_1 = x_1$ and $\tilde{h}_2(\tilde{x}) = h_2(x)$ for subvector $\tilde{x}_2 = x_2$. From this system decomposition, the synthesis of the coordinated distributed decision variables is performed by solving the two parallel subproblems (see (16))

\[
\begin{align*}
\min_{x_1} & \quad (2x_1^2 + \tilde{x}_2^2 - 20x_1 - 16\tilde{x}_2) + \tilde{\lambda}_2(x_1 + \tilde{x}_2 - 5) \\
\text{s.t.} & \quad x_1^2 + \tilde{x}_2^2 = 12.5572
\end{align*}
\]

\[
\min_{x_2} \quad (2\tilde{x}_1^2 + x_2^2 - 20\tilde{x}_1 - 16x_2) + \tilde{\lambda}_1(\tilde{x}_1^2 + x_2^2 - 12.5572) \\
\text{s.t.} & \quad \tilde{x}_1 + x_2 = 5
\]

where $\tilde{x}_i, \tilde{\lambda}_i \ (i = 1, 2)$ are values computed at the last iteration. Using available solvers, the solutions of subproblems (34) and (35) are,

\[
\begin{align*}
\begin{bmatrix} x_1^* \\ \tilde{x}_1^* \end{bmatrix} &= \begin{bmatrix} 2.66 \\ -2.97 \end{bmatrix} \\
\begin{bmatrix} x_2^* \\ \tilde{x}_2^* \end{bmatrix} &= \begin{bmatrix} 2.33 \\ 25.22 \end{bmatrix}
\end{align*}
\]

For this simple example, we see that the distributed decision variables have converged to those of the centralized problem.

VI. CONCLUSION

A new methodology to distributed model predictive control has been presented which aims at achieving control performance almost similar to that of a targeted centralized model predictive control. The methodology consists of two successive steps: a first step, based on an insight into the structure of Karush-Kuhn-Tucker matrices arising from the necessary optimality conditions, allows a decomposition of a large-scale system into interacting subsystems. The second step concerns the synthesis of coordinated distributed subcontrollers of the decomposed system. The main theoretical tool for this second step is the optimality condition decomposition technique which is similar to a modified Lagrangian relaxation with the benefits that it ensures the convergence of the distributed decision variables towards the optimal decision variables of the targeted centralized problem. A simple example is used to illustrate the different steps from system decomposition to the synthesis of decision variables. The DMPC approach of this paper was motivated by control problems and energy efficiency issues of large-scale non-residential buildings. A complete study and simulations illustrating the application of the developed DMPC methodology to buildings will be reported in a follow-on paper to be published elsewhere.

VII. ACKNOWLEDGMENTS

This work was partially supported by Energy-In-Time project funded by the European Union within the 7th Framework Program FP7-NMP, Sub-program EeB.NMP.2013-4: Integrated control systems and methodologies to monitor and improve building energy performance.

REFERENCES

[10] Petru-Daniel Morosan, Romain Bourdais, Didier Dumur, Jean Buisson, Distributed model predictive control based on Benders decomposition applied to multisource multizone building temperature regulation, American Control Conference (ACC), 2010