Chapter 6
Distributed MPC Under Coupled Constraints Based on Dantzig-Wolfe Decomposition

R. Bourdais, J. Buisson, D. Dumur, H. Guéguen and P-D. Moroşan

Abstract In this chapter, we propose a distributed model predictive control scheme based on the Dantzig-Wolfe decomposition to control a collection of linear dynamical systems coupled by linear global constraints. The resulting structure is composed of one optimization agent for each system, and another one that has to ensure that the global constraints are fulfilled. The global solution of the problem is found in a finite number of iterations.

6.1 Short Introduction

This chapter deals with large-scale systems, composed with independent subsystems that share common resources. A typical example of that kind of system is the problem of energy management system for a residential house: heating systems, refrigerant systems, hot-water tank, washing-machine are independent equipments but the global available power is limited by a subscripted contract. A global objective is to minimize the total consumption of the building, while fulfilling the local objectives such as a

R. Bourdais · J. Buisson · H. Guéguen
IETR (UMR-CNRS 6164), SUPELEC, Cesson-Sévigné, France
e-mail: roman.bourdais@supelec.fr

J. Buisson
e-mail: jean.buisson@supelec.fr

H. Guéguen
e-mail: herve.gueguen@supelec.fr

D. Dumur
E3S, SUPELEC, Gif sur Yvette, France
e-mail: didier.dumur@supelec.fr

P-D. Moroşan
Acsystème, Rennes, France
e-mail: petru-daniel.morosan@acsysteme.com

good ambient air temperature, … Other examples of such a system could be a local residential area in which each house has its own behavior, or systems connected by smart grid.

The treatment of the optimization problem could be too complex with a global approach. Moreover, each time another subsystem is added, the whole controller synthesis has to be done again. For these reasons, a distributed structure is interesting. It leads to define smaller and less complex optimization problems as well as to ensure a modularity of the structure.

In the following, the whole optimization problem is linear: the local models and their constraints as well as the global objective and the global constraints. The special structure of all the constraints and the linear aspect of the problem allow to use the Dantzig-Wolfe decomposition. In a few words, the Dantzig-Wolfe decomposition is an algorithm which is well-known to simplify the complexity of a linear optimization problem when the matrix of constraints is block-angular. The initial problem is decomposed into several smaller linear problems, and an coordination mechanism is used such that the solution of the initial problem is found. In this chapter, as it will be explained, the distributed control structure will be based on this decomposition, which will be used not only to reduce the complexity of the global problem, but also to ensure the modularity.

6.2 Boundary Conditions

6.2.1 System Description

We suppose in this chapter that the studied system is composed of a collection of q independent discrete-time dynamical systems (S_i) that are interconnected by different input constraints. Their dynamical equation can be described as follows:

$$\begin{cases} \mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_i \mathbf{u}_i(k), \\ \mathbf{y}_i(k) \quad = \mathbf{C}_i \mathbf{x}_i(k). \end{cases} \tag{6.1}$$

In this equation, the local state $\mathbf{x}_i$ is in $\mathbb{R}^{n_{x_i}}$, the local control input $\mathbf{u}_i$ is in $\mathcal{U}_i \subset \mathbb{R}^{n_{u_i}}$ and the local output $\mathbf{y}_i$ is in $\mathbb{R}^{n_{y_i}}$. We suppose that the local input constraint set is defined by n_{c_i} linear constraints:

$$\mathcal{U}_i := \left\{ \mathbf{u}_i \in \mathbb{R}^{n_{u_i}} \quad \setminus \quad \forall n \in \{1, \dots, n_{c_i}\}, \quad \mathbf{e}_{i_n}^{\mathsf{T}} \mathbf{u}_i \leq d_{i_n} \right\} \tag{6.2}$$

This set can be also described by the aggregated notation:

$$\mathcal{U}_i := \left\{ \mathbf{u}_i \in \mathbb{R}^{n_{u_i}} \quad \setminus \quad \mathbf{E}_i \mathbf{u}_i \leq \mathbf{d}_i \right\}$$

Though all these systems are dynamically independent, we suppose that they are connected by n_G global linear input constraints. The global constraint set is defined as follows:

$$\mathcal{U}_G := \left\{ (\mathbf{u}_i)_{i=1,q} \in \mathbb{R}^{n_{u_1} + \cdots + n_{u_q}} \ \backslash \ \forall n \in \{1, \ldots, n_G\}, \ \sum_{i=1}^{q} \mathbf{h}_{i_n}^{\mathrm{T}} \mathbf{u}_i \leq g_n \right\}.$$

$$(6.3)$$

For the different inputs, we suppose that they fulfill the conditions given in Assumption 6.1, which are not restrictive as soon as we consider engineering systems. Besides, we suppose that there exists at least a collection of local inputs that satisfies the global constraints, this point is given in Assumption 6.1:

Assumption 6.1 Boundedness and existence of a global solution. *We suppose that:*

- *all the sets $\mathcal{U}_i$ are nonempty bounded convex sets.*
- *there exists $\mathbf{v} = (\mathbf{v}_i)_{i=1,q}$ such that $\mathbf{v} \in \mathcal{U}_G$ and for all $i \in \{1, \ldots, q\}$, $\mathbf{v}_i \in \mathcal{U}_i$.*

The objective in this paper is to implement a model predictive control for each systems, while taking in consideration the global constraints. This problem is formalized in the next section.

6.2.2 Control Objective

In predictive control, most of the results are provided for a quadratic cost which is interesting for different reasons such as the explicit solution or stability analysis. Here, the problem is tackled with the l_1-norm which is more representative of real costs (price, power consumption, …).

For a given prediction horizon N_p, we denote by $\mathbf{u}_i(k : k + N_p - 1)$ the vector of all the control inputs over the prediction horizon. $\mathbf{y}_i(k + 1 : k + N_p)$ is the predicted output, constructed from the state $\mathbf{x}_i(k)$ and according to the input vector $\mathbf{u}_i(k : k + N_p - 1)$. $\mathbf{y}_i(k + 1 : k + N_p)$ can be defined recursively by:

$$\mathbf{y}_i(k + l) = \mathbf{C}_i \mathbf{x}_i(k + l)$$
$$= \mathbf{C}_i \left(\mathbf{A}_i^{(l)} \mathbf{x}_i(k) + \sum_{j=1}^{l} \mathbf{A}_i^{(l-j)} \mathbf{B}_i \mathbf{u}_i(k + j - 1) \right) \qquad (6.4)$$

Then, this equation can be aggregated as follows:

$$\mathbf{y}_i(k + 1 : k + N_p) = \mathbf{M}_i \mathbf{x}_i(k) + \mathbf{N}_i \mathbf{u}_i(k : k + N_p - 1) \qquad (6.5)$$

The expressions of the matrices $\mathbf{M}_i$ and $\mathbf{N}_i$ are directly derived from equation (6.4). Let us denote with $\mathbf{y}_{\mathrm{ref},i}(k + 1 : k + N_p)$ the vector of the future reference.

The objective function for each system (S_i) can be stated as follows:

$$J_i(\mathbf{x}_i(k), \mathbf{u}_i(k : k + N_p - 1)) = \left\| \mathbf{y}_{ref,i}(k + 1 : k + N_p) - \mathbf{y}_i(k + 1 : k + N_p) \right\|_1$$
$$+ \alpha_i \left\| \mathbf{u}_i(k : k + N_p - 1) \right\|_1. \tag{6.6}$$

α_i is a positive weight factor. From these local objective functions, the global problem can be derived, integrating the global constraints. This leads to:

Problem 6.1 *Centralized Problem.* Given all the current states and the future references $\left(\mathbf{x}_i(k), \mathbf{y}_{ref,i}(k + 1 : k + N_p) \right)_{i=1,q}$

$$\min_{(\mathbf{u}_i(k:k+N_p-1))_{i=1,q}} \sum_{i=1}^{q} J_i(\mathbf{x}_i(k), \mathbf{u}_i(k : k + N_p - 1)), \tag{6.7}$$

subject to:

- the local dynamical equations (6.1),
- the local constraints, $\forall i \in \{1, \ldots, q\}$:

$$\forall l \in \{1, \ldots, N_p\}, \quad \mathbf{u}_i(k + l - 1) \in \mathcal{U}_i,$$

- and the global constraints,

$$\forall l \in \{1, \ldots, N_p\}, \quad (\mathbf{u}_i(k + l - 1))_{i=1,q} \in \mathcal{U}_G.$$
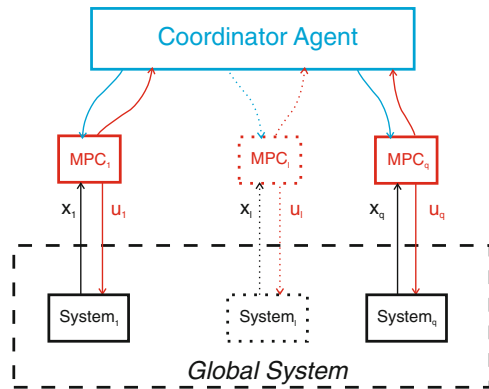
6.2.3 Control Structure

Because of the very particular structure of the global system (all the local systems are independent), the distribution of the optimization problem can be very interesting. The resulting control structure is based on $q + 1$ optimization agents. One will be dedicated to each system (S_i), and one which will be in charge of the fulfillment of the global constraints, called the coordinator agent. Concerning the communication protocols, the coordinator agent will exchange data with all the other agents, but the other agents will not communicate with each other. The control structure is presented in Fig. 6.1. All the interactions will be formalized in the next section.

In this chapter, we suppose that all the exchanges are perfect between the agents, this leads to the following assumption:

Assumption 6.2 *Communication and data exchange.* For the communication protocol, we suppose that:

Fig. 6.1 Distributed structure

- there is no delay induced by the communications.
- there is no loss of information during the exchanges.

Now that the global problem and the control structure are presented, the different steps of the resolution will be specified in the following section.

6.3 Description of the Approach

The principle of the decomposition is to use an iterative procedure at each time step (k). Then, for each iterative step, three actions are sequentially achieved:

- A first one is done by the coordinator agent which has to solve its own optimization problem. Its solution is then sent to each other agent.
- A second action in which all the local agents solve, in parallel, their optimization problem. Their solution is sent to the coordinator agent.
- A last step, in which a decision is taken by the coordinator agent. Two cases can occur: either the convergence is reached and the procedure is stopped, or the convergence is not reached, and then a new iteration has to be done.

In the first two steps, different optimizations have to be solved. The distribution of the initial problem relies on linear decomposition technique, and all the agents solve their problem using a linear solver. As it can be noticed, the local cost function (6.6) is the sum of l_1-norm functions. It has to be rewritten into a linear form. This point is developed in the next paragraph, before the formalization of all the optimization problems and the interactions between the agents.

6.3.1 Exact Linearization of the Local Problem

As the function (6.6) is a sum of l_1-norm functions, it is also a sum of absolute values. To linearize the problem, the idea is to introduce two auxiliary variables for each absolute value.

For a better understanding, let us consider the case where $n_{u_i} = n_{y_i} = 1$ and the prediction horizon $N_p = 1$.

The local minimization problem is the following one:

$$\min_{u_i(k)} |y_{ref,i}(k+1) - y_i(k+1)| + \alpha_i |u_i(k)|, \tag{6.8}$$

subject to

$$u_i(k) \in \mathcal{U}_i. \tag{6.9}$$

Then by introducing 4 positive auxiliary variables $\underline{\mu_i}(k), \overline{\mu_i}(k), \underline{\eta_i}(k), \overline{\eta_i}(k)$, the previous problem is equivalent to:

$$\min_{u_i(k), \underline{\mu_i}(k), \overline{\mu_i}(k), \underline{\eta_i}(k), \overline{\eta_i}(k)} \left(\underline{\eta_i}(k) + \overline{\eta_i}(k) \right) + \alpha_i \left(\underline{\mu_i}(k) + \overline{\mu_i}(k) \right), \tag{6.10}$$

subject to

$$\mathbf{E}_i u_i(k) \leq \mathbf{d}_i \tag{6.11}$$

$$y_i(k+1) + \underline{\eta_i}(k) - \overline{\eta_i}(k) = y_{ref,i}(k+1) \tag{6.12}$$

$$u_i(k) + \underline{\mu_i}(k) - \overline{\mu_i}(k) = 0 \tag{6.13}$$

$$\underline{\mu_i}(k), \overline{\mu_i}(k), \underline{\eta_i}(k), \overline{\eta_i}(k) \geq 0 \tag{6.14}$$

Sketches of proof can be found in [3], in which another formulation is proposed to linearize a l_1-norm predictive control.

6.3.2 Local Agent Optimization Problem

This transformation is exactly the same in the general case. Just to have an estimation of the number of variables involved for each subsystem optimization, the initial problem has $n_{u_i} \cdot N_p$ variables and $n_{c_i} \cdot N_p$ constraints with a nonlinear cost function whereas the rewritten problem has $(3n_{u_i} + 2n_{y_i}) \cdot N_p$ variables and $(n_{c_i} + n_{u_i} + n_{y_i}) \cdot N_p$ constraints but with a linear cost function.

Then, let us denote with $\mathbf{w}_i(k : k + N_p - 1)$ the vector of all the variables of the rewritten local optimization problem, with

$$\mathbf{w}_i(k : k + N_{\mathrm{p}} - 1) = \begin{pmatrix} \mathbf{u}_i(k : k + N_{\mathrm{p}} - 1) \\ \mu_i(k : k + N_{\mathrm{p}} - 1) \\ \overline{\mu_i}(k : k + N_{\mathrm{p}} - 1) \\ \eta_i(k : k + N_{\mathrm{p}} - 1) \\ \overline{\eta_i}(k : k + N_{\mathrm{p}} - 1) \end{pmatrix}$$

As we will see in the following, from an iteration to the next one during the distribution process, all the constraints of the local problem remain the same, but the linear cost function can change. At an iteration p, let us denote with $\mathbf{k}_i^{(p)\mathrm{T}} \mathbf{w}_i(k : k + N_{\mathrm{p}} - 1)$ this cost function.

Then, the local problem can be stated as follows:

Problem 6.2 *Rewritten Local Optimization Problem.* For a time step k, and at an iteration p, the local optimization problem is:

$$\min_{\mathbf{w}_i(k:k+N_{\mathrm{p}}-1)} \quad \mathbf{k}_i^{(p)\mathrm{T}} \mathbf{w}_i(k : k + N_{\mathrm{p}} - 1), \tag{6.15}$$

subject to

$$\overline{\mathbf{E}}_i \mathbf{w}_i(k : k + l - 1) \le \overline{\mathbf{d}}_i(k). \tag{6.16}$$

It is important to notice that in this local problem, the matrix $\overline{\mathbf{E}}_i$ is independent of the time contrary to the vector $\overline{\mathbf{d}}_i(k)$, which has to be calculated at each time step and depends on the future target $\mathbf{y}_{\mathrm{ref},i}(k + 1 : k + N_{\mathrm{p}})$ and the current state $\mathbf{x}_i(k)$. Their expressions are the following:

$$\overline{\mathbf{E}}_i = \begin{bmatrix} diag(\mathbf{E}_i) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{N}_i & \mathbf{0} & \mathbf{0} & \mathbf{I} & -\mathbf{I} \\ -\mathbf{N}_i & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} \\ \mathbf{I} & \mathbf{I} & -\mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{I} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \end{bmatrix}, \quad \overline{\mathbf{d}}_i(k) = \begin{bmatrix} \mathbf{d}_i \\ \vdots \\ \mathbf{d}_i \\ \hline \mathbf{y}_{\mathrm{ref},i}(k + 1 : k + N_{\mathrm{p}}) - \mathbf{M}_i \mathbf{x}_i(k) \\ -\mathbf{y}_{\mathrm{ref},i}(k + 1 : k + N_{\mathrm{p}}) + \mathbf{M}_i \mathbf{x}_i(k) \\ \hline \mathbf{0} \\ \hline \mathbf{0} \\ \hline \mathbf{0} \\ \hline \mathbf{0} \\ \hline \mathbf{0} \\ \hline \mathbf{0} \end{bmatrix} \tag{6.17}$$

During the iteration process, the cost function $\mathbf{k}_i^{(p)\mathrm{T}} \mathbf{w}_i(k : k + N_{\mathrm{p}} - 1)$ will change from an iteration to another one. This is the way that the global constraints are going to be taken into account. When there is no global constraint, the cost function of the rewritten local optimization problem would be $\mathbf{k}_{i,0}^{\mathrm{T}} \mathbf{w}_i(k : k + N_{\mathrm{p}} - 1)$, with:

$$\mathbf{k}_{i,0}^{\mathrm{T}} = \begin{bmatrix} \mathbf{0}^{\mathrm{T}} \,|\, \alpha_i \mathbf{1}^{\mathrm{T}} \,|\, \alpha_i \mathbf{1}^{\mathrm{T}} \,|\, \mathbf{1}^{\mathrm{T}} \,|\, \mathbf{1}^{\mathrm{T}} \end{bmatrix} \tag{6.18}$$

As it will be explained in the following, this vector will be used during the exchanges from the local agents to the coordinator agent.

6.3.3 The Coordinator Agent Problem

The main objective of the coordinator agent is to guarantee that the global constraints are fulfilled. This is ensured by different optimizations that will be done iteratively. From an iteration to the next one, a new optimization variable is added. The way this new optimization variable is integrated, on the cost function and on the constraints, only depends on the solutions of the local optimizations that have been done at the previous iteration. But before starting, the initialization requires N_0 different admissible points for the local agents. More precision on the initialization will be given in a next paragraph.

Then, at iteration p, the coordinator agent will have $v^{(p)}$ optimization variables, with $v^{(p)} = N_0 + p - 1$. For $s = \{1, \ldots, v^{(p)}\}$, let us denote with $\mathbf{w}_i^{(s)}(k : k + N_p - 1)$ the s-th admissible point of the local agent.[1]

Then, we define the following variables, $\forall s \in \{1, \ldots, v^{(p)}\}$,

$$z^{(s)}(k) = \sum_{i=1}^{q} \mathbf{k}_{i,0}^{\mathrm{T}} \mathbf{w}_i^{(s)}(k : k + N_p - 1) \tag{6.19}$$

$$r_{n,l}^{(s)}(k) = \sum_{i=1}^{q} \mathbf{h}_{i_n}^{\mathrm{T}} \mathbf{u}_i^{(s)}(k + l - 1), \quad \forall n \in \{1, \ldots, n_G\}, \forall l \in \{1, \ldots, N_p\} \tag{6.20}$$

From this, the coordinator agent has to solve its own problem, which can be stated as follows:

Problem 6.3 *Coordinator Optimization Problem*. At an iteration p, given the variables $z^{(s)}(k)$ and $r_{n,l}^{(s)}(k)$, with $s \in \{1, \ldots, v^{(p)}\}$, the optimization problem is:

$$\min_{(v_s)_{s=1,v^{(p)}}} \sum_{s=1}^{v^{(p)}} z^{(s)}(k) v_s \tag{6.21}$$

with respect to

---

[1] For $s \in \{1, \ldots, N_0\}$, this point is given by the initialization, and for $s \in \{N_0 + 1, \ldots, N_0 + p - 1\}$ the point is the result of a local optimization problem.

$$\forall n \in \{1, \ldots, n_G\}, \forall l \in \{1, \ldots, N_p\}, \sum_{s=1}^{v^{(p)}} r_{n,l}^{(s)}(k) v_s \leq g_n \quad : \lambda_{n,l} \quad (6.22)$$

$$\sum_{s=1}^{v^{(p)}} v_s = 1 \quad : \sigma \quad (6.23)$$

$$\forall s \in \{1, \ldots, v^{(p)}\}, \quad v_s \geq 0 \quad (6.24)$$

The coordinator agent has to find not only the solution v_s but also the dual variable values $\lambda_{n,l}$, associated to the linear constraints.[2] (6.22) and the dual variable value σ associated to the linear constraint (6.23).

6.3.4 Iterative Mechanism

Each iteration consists in the following three steps:

1. The coordination agent optimization. At iteration p, it has to solve its problem 6.3. Let us denote with $(v_s^{(p)})_{s=1,v^{(p)}}$ the solution of this problem, and $\lambda_{n,l}^{(p)}$ and $\sigma^{(p)}$ the associated optimal dual variable values.
   The values of the dual variables $\lambda_{n,l}^{(p)}$ are then sent to each local agent, and will be used to define a new cost function $\mathbf{k}_i^{(p)\mathrm{T}} \mathbf{w}_i(k : k + N_p - 1)$, which is calculated using:

$$\mathbf{k}_i^{(p)} = \mathbf{k}_{i,0} + \begin{bmatrix} -\left(\sum_{n=1}^{n_G} \lambda_{n,1}^{(p)} \mathbf{h}_{i_n}\right) \\ \vdots \\ -\left(\sum_{n=1}^{n_G} \lambda_{n,N_p}^{(p)} \mathbf{h}_{i_n}\right) \\ \hline 0 \\ \hline 0 \\ \hline 0 \\ \hline 0 \end{bmatrix} \quad (6.25)$$

The important point is that the second term of this sum can be calculated by coordinator agent and given numerically to the local agent. This term is a penalization cost vector whose objective is to take into account the global constraints in the local optimization.

---

[2] $\forall n \in \{1, \ldots, n_G\}, \forall l \in \{1, \ldots, N_p\}.$

2. In parallel, each optimization agent has to solve its own problem 6.2 with the new cost function $\mathbf{k}_i^{(p)\mathrm{T}}\mathbf{w}_i(k : k + N_\mathrm{p} - 1)$ to find its solution $\mathbf{w}_i^{(v^{(p)}+1)}(k : k + N_\mathrm{p} - 1)$, which has to be sent to the coordinator agent.

3. The convergence checking is done after these two steps by comparing two values:

   - the value of the dual variable $\sigma^{(p)}$
   - the value of the sum of all the local optimization values, calculated as follows:

$$\psi^{(p)} = \sum_{i=1}^{q} \mathbf{k}_i^{(p)\mathrm{T}} \mathbf{w}_i^{(v^{(p)}+1)}(k : k + N_\mathrm{p} - 1) \tag{6.26}$$

The value $\psi^{(p)}$ obtained by Eq. (6.26) can be seen as the cost function resulting from a relaxed optimization problem with less constraint than the initial one. Two cases can occur:

3a. If $\psi^{(p)} < \sigma^{(p)}$, a new iteration has to be done. The coordinator agent will use the local solutions $\mathbf{w}_i^{(v^{(p)}+1)}(k : k + N_\mathrm{p} - 1)$ to calculate new variable values $z^{(v^{(p)}+1)}(k)$ and $r_{n,l}^{(v^{(p)}+1)}(k)$, according to Eqs. (6.19) and (6.20).

3b. In the other case, (i.e. $\psi^{(p)} \geq \sigma^{(p)}$), the optimal solution of the initial problem has been achieved. More precisely, the local optimal solution can be computed as follows:

$$\mathbf{w}_i^{\star}(k : k + N_\mathrm{p} - 1) = \sum_{s=1}^{v^{(p)}} v_s^{(p)} \mathbf{w}_i^{(s)}(k : k + N_\mathrm{p} - 1), \tag{6.27}$$

which is a convex combination of all the previous solutions done for the different cost function.

As usual in model predictive control, only the first input of the control sequence is brought to the system, which is $\mathbf{u}_i^{\star}(k)$.

6.3.5 Initialization

This point is maybe the main difficulty and drawback of the method. To initialize the algorithm, it is necessary to provide enough different solutions to the master problem such that it can be feasible. From a theoretical point of view, only two different solutions are enough, but only if they are good ones …they cannot be chosen randomly, but such that the resulting master problem is feasible. From a practical point of view, they can be provided using engineering considerations, but also could be the results of different local optimization done with arbitrary cost functions. But

anyway, there exist different techniques such as the one proposed in [5] to develop an always feasible coordinator agent problem.

6.3.6 Overview of the Procedure

From all the previous considerations, the iterative procedure of the distributed MPC based on the Dantzig-Wolfe decomposition can be recapitulated as in Algorithm 6.1. This algorithm will be called at each time step k.

6.3.7 A Remark on Modularity

Among the mechanisms used during the distribution procedure, it is important to notice that each local agent only interacts with the coordinator agent but not with other local agents. Then, adding a new subsystem implies a new local agent which is integrated in the structure, but it only communicates with the coordinator agent without any modification on the other agents. Moreover the complexity of the coordinator optimization problem is not increasing, because it only changes the way the coefficients $z^{(s)}(k)$ and $r_{n,l}^{(v^{(s)})}(k)$ are calculated. All these considerations ensure the modularity of the approach.

6.4 Theoretical Results Availability

The two major advantages of this method are that:

- The global solution will be found
- It requires a finite number of iterations.

The presented methodology is very close to the Dantzig-Wolfe decomposition procedure, which has been firstly proposed in [6]. The conditions of the convergence is linked to the fact that at each step, all the solutions of the optimization problems are finite, and there is no unfeasible problem.

For all the local agent, as from an optimization to the next one, only the cost function changes, it is always possible to solve the problem and the solution is finite. This point is ensured by the Assumption 6.1. For the coordinator agent problem, as soon as the initialization is well tuned, the problem cannot be unfeasible anymore. Because of Assumption 6.2, in which all the communications are supposed to be perfect, the iterative process is then ensured to converge to the optimal solution.

All the mathematical concepts that are used to prove the convergence of the decomposition can be found in the different following references [1, 2, 4], in which many different analyzes have been provided.

Algorithm 6.1 dMPC Algorithm

1: **Initialization:**

    Step 1a: Local agent, in parallel:

- Acquisition of $\mathbf{x}_i(k)$, $\mathbf{y}_{\mathrm{ref},i}(k)$
- Calculation of $\bar{\mathbf{d}}_i(k)$
- Determination of N_0 admissible points $\mathbf{w}_i^{(s)}(k : k + N_{\mathrm{p}} - 1)$
- Transmission of the data to the coordinator agent.

    Step 1b: Coordination agent:

- $p = 1$, $\upsilon^{(p)} = N_0$.
- Calculation of $z^{(s)}(k)$ and $r_{n,l}^{(\upsilon^{(s)})}(k)$ for $s \in \{1, \dots, \upsilon^{(p)}\}$

2: **Iteration process: Coordinator Agent Optimization**

- Solving the coordinator agent problem, to get $(v_s^{(p)})_{s=1,\upsilon^{(p)}}$ the solution of this problem, and $\lambda_{n,l}^{(p)}$ and $\sigma^{(p)}$ the associated optimal dual variable values.
- Calculation of the cost penalization vector for each agent, using the $\lambda_{n,l}^{(p)}$ values, and transmission of this vector.

3: **Iteration process: Local Agent Optimizations, in parallel**

- Solving the local optimization problem, to get the solution $\mathbf{w}_i^{(\upsilon^{(p)}+1)}(k : k + N_{\mathrm{p}} - 1)$.
- Transmission of the solution to the coordinator agent.

4: **Iteration process: Coordinator agent, Convergence checking**

- Calculation of $\psi^{(p)}$.
- Test $\psi^{(p)} < \sigma^{(p)}$

    Case 1: YES. Then:

- Calculation of $z^{(\upsilon^{(p)}+1)}(k)$ and $r_{n,l}^{(\upsilon^{(p)}+1)}(k)$
- $p \leftarrow p + 1$, $\upsilon^{(p)} \leftarrow \upsilon^{(p)} + 1$
- GO BACK to Step 2.

    Case 2: NO. Then:

- Send $(v_s^{(p)})_{s=1,\upsilon^{(p)}}$ to all the agent.
- GO TO Step 5.

5: **Local Agent, Execution, in parallel**

- Calculation of $\mathbf{w}_i^{\star}(k : k + N_{\mathrm{p}} - 1)$
- Applying the first control input $\mathbf{u}_i^{\star}(k)$
- END OF Procedure

6.5 Application Results Availability

In this chapter, we propose a distributed structure to control different systems when they are connected by global input constraints. This is the easiest case to use the Dantzig-Wolfe decomposition. This can be extended to a more general class of systems, when there exist global constraints on the outputs or on the states. In this case, the main difficulty is linked to the feasibility of the problem, which is not ensured anymore if the constraints are hard one. Different methods can be used to soften these constraints and then to use the decomposition.

In this chapter, all the systems are dynamically independent. In a more general case, if the systems are connected by the inputs, states or outputs, the methodology can be adapted to take into account these interactions. This can be done using nested iterations and a communication-based algorithm, such the one proposed in [9]. The major drawback of these nested iterations is the fact that the algorithm converges but not to the optimal solution of the problem. Anyway, this method has been used with success on building temperature regulation, where it has been shown that the suboptimality is not really significant.

In a case where all the systems are connected by common input variables, the problem can also be distributed using the dual of the Dantzig-Wolfe decomposition, the Benders' decomposition. This has been used to regulation temperature in building in the multi-source case [10].

If the first applications of the method dealt with temperature regulation in buildings [8], it also has been extended to energy management system in residential house to develop an adjustable structure to optimize the energy bill [7].

6.6 Conclusions

In this chapter, a distributed structure has been proposed to control a collection of linear dynamical systems coupled by linear global constraints. The particular structure of the constraints set allows the use of the Dantzig-Wolfe decomposition to define the control architecture. The two major advantages of this method are first the fact that the solution found during the iteration process is the global solution. The second one is link to the modularity induced by the control structure which allow adding or removing subsystems in the structure without any modification for the other local controllers. The methodology is proposed for a collection of independent subsystems, a current direction for future works is how to generalize this method to interconnected subsystems.

References

1. M.S. Bazaraa, J.J. Jarvis, H.D. Sherali. Linear Programming and Network Flows, (Wiley, New York, 1990)
2. S.P. Bradley, A.C. Hax, T.L. Magnanti, *Applied Mathematical Programming* (Addison-Wesley Publishing Company, Reading, 1997)
3. E.F. Camacho, C. Bordons, *Model Predictive Control in the Process Industry*, 2nd edn. (Springer-Verlag, London, England, 2004)
4. V. Chvátal, *Linear Programming* (Freeman and Company, New York, 1983)
5. A.J. Conejo, E. Castillo, R. Mínguez, R. García-Bertrand. *Decomposition Techniques in Mathematical Programming*. (Springer, London, 2006)
6. G.B. Dantzig. *Linear Programming and Extensions.* (Princeton University Press, NJ, 1963)
7. A. Lefort, R. Bourdais, G. Ansannay-Alex, H. Guéguen. Planification de la consommation énergétique d'un bâtiment par une méthode d'optimisation linéaire distribuée, *In Proceedings of CIFA 2012*, (2012)
8. P-D. Moroşan, R. Bourdais, D. Dumur, J. Buisson. Distributed model predictive control for building temperature regulation. In *American Control Conference*, pp. 3174–3179, Jun. 2010
9. P.-D. Moroşan, R. Bourdais, D. Dumur, J. Buisson. In Distributed MPC for multi-zone temperature regulation with coupled constraints, *IFAC World Congress*, Aug, 2011
10. P.-D. Moroşan, R. Bourdais, D. Dumur, J. Buisson, A distributed MPC strategy based on Benders' decomposition applied to multi-source multi-zone temperature regulation. J. Process Control **21**(5), 729–737 (2011)