

Chapter 11

Distributed MPC Via Dual Decomposition

B. Biegel, J. Stoustrup and P. Andersen

Abstract This chapter presents dual decomposition as a means to coordinate a number of subsystems coupled by state and input constraints. Each subsystem is equipped with a local model predictive controller while a centralized entity manages the subsystems via prices associated with the coupling constraints. This allows coordination of all the subsystems without the need of sharing local dynamics, objectives and constraints. To illustrate this, an example is included where dual decomposition is used to resolve power grid congestion in a distributed manner among a number of players coupled by distribution grid constraints.

11.1 Short Introduction

In this chapter we consider a number of dynamical subsystems; each subsystem has local inputs and states, a local objective function, and local state and input constraints. Moreover, global state and input constraints make the subsystems mutually dependent. The subsystems are not able (or willing) to share the local information; hence optimization of the operation of the subsystems cannot be performed centrally and a distributed approach is necessary.

We consider two small figurative examples to illustrate such global constraints causing coupling of the subsystems. As a first example, consider a number of subsystems that are dependent on the same shared limited resource: this could correspond

B. Biegel (✉) · J. Stoustrup · P. Andersen
Department of Electronic Systems, Automation and Control Aalborg University,
Aalborg, Denmark
e-mail: bbi@es.aau.dk

J. Stoustrup
e-mail: jakob@es.aau.dk

P. Andersen
e-mail: pa@es.aau.dk

to coupling input inequality constraints. In the second example, consider a number of producing and consuming subsystems in a setup where balance must exist: this could correspond to coupling state equality constraints. In both cases the optimization problem is to minimize the total objective while honoring both local and global constraints—without sharing local information.

This chapter presents an approach to solve this problem via dual decomposition: by associating each coupling constraint with a price, the subsystems can be managed by a central entity to reach the solution. This allows coordination of the individual subsystems without sharing local dynamics, constraints or objectives. Further, the final prices of the coupling constraints, the so-called *shadow prices*, will reveal the marginal cost that each agent is willing to pay for the shared resources. This allows the shadow prices to be used for economical settlement purposes between the subsystems.

Dual decomposition is a huge area of research and there exists a large amount of literature on the topic. Dual decomposition appeared already in 1960s where it was used for solving large-scale optimization problems [6, 9]. Also within the area of coordination of dynamic systems via dual decomposition, which is the topic of this chapter, large amounts of literature exists; some references for this are [7, 10–12]. In this chapter, we show the basic idea in using dual decomposition in the coordination of coupled dynamic subsystems.

11.2 Boundary Conditions

We consider N subsystems each described by a discrete linear time-invariant state space model. The states and inputs of subsystem i are denoted $\mathbf{x}_i(k) \in \mathbb{R}^{n_{x,i}}$ and $\mathbf{u}_i(k) \in \mathbb{R}^{n_{u,i}}$, respectively. The state space model is formulated as

$$\mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_i \mathbf{u}_i(k) \quad (11.1)$$

where $\mathbf{A}_i \in \mathbb{R}^{n_{x,i} \times n_{x,i}}$ is the state matrix and $\mathbf{B}_i \in \mathbb{R}^{n_{x,i} \times n_{u,i}}$ is the input matrix. Each subsystem is subject to state and input constraints:

$$\mathbf{x}_i(k) \in \mathcal{X}_i, \mathbf{u}_i(k) \in \mathcal{U}_i \quad (11.2)$$

where \mathcal{X}_i and \mathcal{U}_i are convex constraint sets with $0 \in \mathcal{X}_i, 0 \in \mathcal{U}_i$. The stage cost function of subsystem i is convex and denoted $\ell_i(\mathbf{x}_i(k), \mathbf{u}_i(k))$ and $\ell_i(0, 0) = 0$.

Taking a receding horizon control approach with a finite control horizon of N_c time samples and a prediction horizon of $N_p = N_c$ time samples, a local control strategy at subsystem i can be formulated as follows. Let \mathcal{K} be a set containing the current time sample k and the following $N_c - 1$ time samples: $\mathcal{K} = \{k, \dots, k + N_c - 1\}$, and let \mathcal{N} denote the set of all N subsystems: $\mathcal{N} = \{1, \dots, N\}$. Then we can formulate a decentralized model predictive control algorithm for subsystem i as in Algorithm 11.1.

Algorithm 11.1 Decentralized Model Predictive Control

1: Observe the current state $\mathbf{x}_i(k)$ and solve the optimization problem

$$\begin{aligned} & \text{minimize} && \sum_{\kappa \in \mathcal{K}} \ell_i(\mathbf{x}_i(\kappa + 1), \mathbf{u}_i(\kappa)) \\ & \text{subject to} && \mathbf{x}_i(\kappa + 1) = \mathbf{A}_i \mathbf{x}_i(\kappa) + \mathbf{B}_i \mathbf{u}_i(\kappa), \quad \forall \kappa \in \mathcal{K} \\ & && \mathbf{x}_i(\kappa + 1) \in \mathcal{X}_i, \quad \mathbf{u}_i(\kappa) \in \mathcal{U}_i, \quad \forall \kappa \in \mathcal{K} \end{aligned} \quad (11.3)$$

where the variables are $\mathbf{x}_i(k + 1 : k + N_c)$, $\mathbf{u}_i(k : k + N_c - 1)$ and $\mathbf{x}_i(k)$ is data. The solution is denoted $\mathbf{x}_i^*(k + 1 : k + N_c)$, $\mathbf{u}_i^*(k : k + N_c - 1)$.

2: Apply the first control input solution $\mathbf{u}_i^*(k)$ to subsystem i .

3: Increase k by one and repeat from 1.

Algorithm 11.1 is presented to illustrate the concept of receding horizon control as this control strategy forms the background for the method presented in this chapter. However, this algorithm is not applicable to the subsystems we have in scope: the N subsystems are not only subject to the local constraints 11.2, but also to global state and input constraints. Consider the following compact notation for inputs and states:

$$\mathbf{x}(k) = \left[\mathbf{x}_1(k)^T, \dots, \mathbf{x}_N(k)^T \right]^T \quad (11.4)$$

$$\mathbf{u}(k) = \left[\mathbf{u}_1(k)^T, \dots, \mathbf{u}_N(k)^T \right]^T \quad (11.5)$$

where $\mathbf{x}(k) \in \mathbb{R}^{n_x}$, $n_x = \sum_{i=1}^N n_{x,i}$, and $\mathbf{u}(k) \in \mathbb{R}^{n_u}$, $n_u = \sum_{i=1}^N n_{u,i}$. With this notation we can express the coupling constraints as

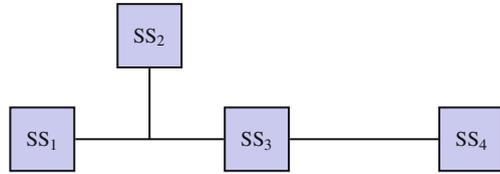
$$\mathbf{C}\mathbf{u}(k) \leq \mathbf{c}, \quad \mathbf{D}\mathbf{u}(k) = \mathbf{d}, \quad (11.6)$$

$$\mathbf{E}\mathbf{x}(k) \leq \mathbf{e}, \quad \mathbf{F}\mathbf{x}(k) = \mathbf{f}, \quad (11.7)$$

where \leq denotes componentwise inequality; $\mathbf{C} \in \mathbb{R}^{n_c \times n_u}$, $\mathbf{c} \in \mathbb{R}^{n_c}$, and $\mathbf{D} \in \mathbb{R}^{n_d \times n_u}$, $\mathbf{d} \in \mathbb{R}^{n_d}$ describe n_c input inequality constraints and n_d input equality constraints, respectively, while $\mathbf{E} \in \mathbb{R}^{n_e \times n_x}$, $\mathbf{e} \in \mathbb{R}^{n_e}$ and $\mathbf{F} \in \mathbb{R}^{n_f \times n_x}$, $\mathbf{f} \in \mathbb{R}^{n_f}$ describe n_e state inequality constraints and n_f state equality constraints, respectively. These types of constraints can for example express the previously described resource couplings or balancing couplings.

We illustrate this idea of coupled subsystems with a small figurative example. Consider $N = 4$ subsystems where subsystems 1, 2, and 3 share a limited resource while a production/consumption balance must exist between subsystems 3 and 4. This example can be visualized as in Fig. 11.1: subsystems 1, 2 and 3 are interconnected by a net of lines and subsystems 3 and 4 are interconnected by a single line representing the coupling constraints. In dual decomposition, each coupling constraint (each interconnection) will be associated with a price. These prices will be used to coordinate the subsystems to collectively honor the coupling constraints.

Fig. 11.1 Illustration of coupled subsystems: subsystems 1, 2, and 3 are coupled and subsystems 3 and 4 are coupled



Hereby the subsystems avoid sharing local information such as dynamics, objective and constraints. Two prices exist in the small example presented in Fig. 11.1: one for the coupling of subsystems 1, 2 and 3 and one for the coupling of subsystems 3 and 4.

Due to the coupling constraints 11.6 and 11.7, the subsystems depend on each other and must coordinate their actions to reach feasibility. In the following sections it will be shown that the subsystems can be coordinated via prices associated with the coupled resources by letting an external agent adjust these prices. It is therefore necessary to assume that each subsystems is able to establish a two-way communication link with such an external agent.

11.3 Description of the Approach

We only consider coupling constraints on the form $\mathbf{C}\mathbf{u}(k) \leq \mathbf{c}$, $\mathbf{C} \in \mathbb{R}^{n_c \times n_x}$, $\mathbf{c} \in \mathbb{R}^{n_c}$ in the following and neglect the three other constraints presented in 11.6 and 11.7. This simplification is made to ease the notation. It is, however, straightforward to follow the method presented in the following to include all four of the presented constraints.

Let

$$\ell(\mathbf{x}(k), \mathbf{u}(k)) = \sum_{i \in \mathcal{N}} \ell_i(\mathbf{x}_i(k), \mathbf{u}_i(k)) \quad (11.8)$$

be the sum of the N convex objective functions of the subsystems and thereby itself a convex function. Based on this, we formulate a Algorithm 11.2 as a control algorithm using the receding horizon approach with a finite control and prediction horizon of $N_c = N_p$ time samples. This algorithm can be applied if all information is available centrally (which is not the case in our setup).

The centralized optimization problem 11.9 is completely separable except for the last coupling constraint $\mathbf{C}\mathbf{u}(\kappa) \leq \mathbf{c}$. As the coupling constraints are affine, we are able to apply dual decomposition to eliminate the coupling (see, e.g., [5, 13]). This is exactly what we will do in the following.

First, we relax the coupling constraints by introducing the associated Lagrange multipliers; hereby the partial Lagrangian of problem 11.9 becomes

Algorithm 11.2 Centralized Model Predictive Control

1: Observe the current states $\mathbf{x}_i(k)$ for all subsystems $i \in \mathcal{N}$ and solve the centralized optimization problem

$$\begin{aligned} & \text{minimize} && \sum_{\kappa \in \mathcal{K}} \ell(\mathbf{x}(\kappa + 1), \mathbf{u}(\kappa)) \\ & \text{subject to} && \mathbf{x}_i(\kappa + 1) = \mathbf{A}_i \mathbf{x}_i(\kappa) + \mathbf{B}_i \mathbf{u}_i(\kappa), \quad \forall \kappa \in \mathcal{K}, i \in \mathcal{N} \\ & && \mathbf{x}_i(\kappa + 1) \in \mathcal{X}_i, \mathbf{u}_i(\kappa) \in \mathcal{U}_i, \quad \forall \kappa \in \mathcal{K}, i \in \mathcal{N} \\ & && \mathbf{C}\mathbf{u}(\kappa) \preceq \mathbf{c}, \quad \forall \kappa \in \mathcal{K} \end{aligned} \quad (11.9)$$

where the variables are

$$\boldsymbol{\eta}(k) = \left[\mathbf{x}(k + 1 : k + N_c)^\top, \mathbf{u}(k : k + N_c - 1)^\top \right]^\top$$

and $\boldsymbol{\eta}(k) \in \mathbb{R}^{N_c(n_x + n_u)}$ is used as a compact representation of states and inputs in the following.

2: Apply the first control input solution $\mathbf{u}_i^*(k)$, $\forall i \in \mathcal{N}$ to the N subsystems.

3: Increase k by one and repeat from 1.

$$L(\boldsymbol{\eta}(k), \boldsymbol{\Lambda}(k)) = \sum_{\kappa \in \mathcal{K}} \left(\ell(\mathbf{x}(\kappa + 1), \mathbf{u}(\kappa)) + \boldsymbol{\lambda}(\kappa)^\top (\mathbf{C}\mathbf{u}(\kappa) - \mathbf{c}) \right) \quad (11.10)$$

where $\boldsymbol{\lambda}(\kappa) \in \mathbb{R}^{n_c}$ are the Lagrange multipliers associated with the inequality constraint $\mathbf{C}\mathbf{u}(\kappa) \preceq \mathbf{c}$ and $\boldsymbol{\Lambda}(k) \in \mathbb{R}^{N_c n_c}$ is a compact representation of the Lagrange multipliers: $\boldsymbol{\Lambda}(k) = \boldsymbol{\lambda}(k : k + N_c - 1)$.

Define $g(\boldsymbol{\Lambda}(k))$ as the optimal value of the problem

$$\begin{aligned} & \text{minimize} && \sum_{\kappa \in \mathcal{K}} \left(\ell(\mathbf{x}(\kappa + 1), \mathbf{u}(\kappa)) + \boldsymbol{\lambda}(\kappa)^\top (\mathbf{C}\mathbf{u}(\kappa) - \mathbf{c}) \right) \\ & \text{subject to} && \mathbf{x}_i(\kappa + 1) = \mathbf{A}_i \mathbf{x}_i(\kappa) + \mathbf{B}_i \mathbf{u}_i(\kappa), \quad \kappa \in \mathcal{K}, i \in \mathcal{N} \\ & && \mathbf{x}_i(\kappa + 1) \in \mathcal{X}_i, \mathbf{u}_i(\kappa) \in \mathcal{U}_i, \quad \kappa \in \mathcal{K}, i \in \mathcal{N} \end{aligned} \quad (11.11)$$

where the variables are $\boldsymbol{\eta}(k)$. This problem is completely separable as both objective and constraints can be separated among the i subsystems. We see this clearly by separating the matrix \mathbf{C} into blocks

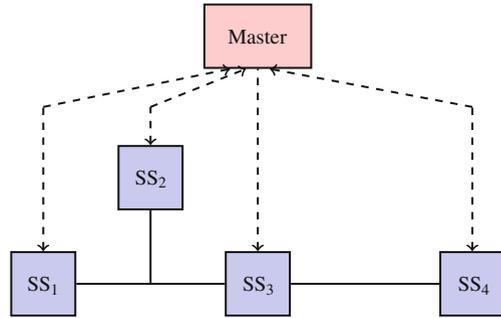
$$\mathbf{C} = [\mathbf{C}_1, \dots, \mathbf{C}_N] \quad (11.12)$$

where $\mathbf{C}_i \in \mathbb{R}^{n_c \times n_{u,i}}$ such that

$$\mathbf{C}\mathbf{u}(k) = \sum_{i \in \mathcal{N}} \mathbf{C}_i \mathbf{u}_i(k). \quad (11.13)$$

Evaluating a subgradient of $g(\boldsymbol{\Lambda}(k))$ can be done as follows. Solve problem 11.11 and let $\bar{\mathbf{u}}(\kappa)$ denote the optimal $\mathbf{u}(\kappa)$, $\forall \kappa \in \mathcal{K}$ for a given realization of $\boldsymbol{\Lambda}(k)$. By differentiation of the objective of problem 11.11 with respect to $\boldsymbol{\Lambda}(k)$ it is evident that a subgradient of $g(\boldsymbol{\Lambda}(k))$ can be described as

Fig. 11.2 Coupled subsystems interact with master: master broadcasts prices $\mathbf{\Lambda}(k)$ and subsystems respond by reporting how much they utilize the limited resources $\mathbf{C}_i \mathbf{x}_i(\kappa)$, $\forall \kappa \in \mathcal{K}$. The dashed lines indicate the necessary two-way communication links between subsystems and master



$$\left[(\mathbf{C}\bar{\mathbf{u}}(k) - \mathbf{c})^T, \dots, (\mathbf{C}\bar{\mathbf{u}}(k + N_c - 1) - \mathbf{c})^T \right]^T \in \partial(g)(\mathbf{\Lambda}(k)), \quad (11.14)$$

where $\partial(g)(\mathbf{\Lambda}(k))$ denotes the subdifferential of g at $\mathbf{\Lambda}(k)$.

We can formulate the dual of the original centralized problem 11.9 as

$$\begin{aligned} & \text{maximize} && g(\mathbf{\Lambda}(k)) \\ & \text{subject to} && \mathbf{\Lambda}(k) \geq 0 \end{aligned} \quad (11.15)$$

with variables $\mathbf{\Lambda}(k)$. Based on the above, we are able to solve the original problem 11.9 in a distributed manner. The key idea is to solve the primal problem 11.9 by solving its dual problem 11.15 using a projected subgradient method. In the subgradient method, steps of appropriate length are taken in the direction of a subgradient of the dual problem which corresponds to iteratively updating the Lagrange multipliers $\mathbf{\Lambda}(k)$. We can do this in a distributed manner as a subgradient of the dual problem 11.15 is given by 11.14 which is separable among the subsystems as $\mathbf{C}\bar{\mathbf{u}}(\kappa) = \sum_{i \in \mathcal{N}} \mathbf{C}_i \bar{\mathbf{u}}_i(\kappa)$. Algorithm 11.3 illustrates this. Note that we use the term *Master* to denote a centralized entity able to perform two-way communication with all subsystems (an interpretation of this master entity is presented in the example in the end of this chapter).

Figure 11.2 illustrates Algorithm 11.3: each interconnection of solid lines represents a coupling constraint while the dashed lines illustrate the necessary communication. This shows that the master needs information from each subsystem in order to update the prices and communicate these prices to the subsystems. It is important to note that the master needs no information of local subsystem constraints, objectives or dynamics; it is sufficient that the master knows how much the limited resources will be used at each subsystem under a sequence of different price realizations. Finally we note that the resulting algorithm using dual decomposition has a straightforward interpretation: in step 5 the master observes if the shared resources $\mathbf{u}(k : k + N_c - 1)$ are overutilized or underutilized. If the subsystems overutilize a limited resource, the associated price is increased; if the subsystems underutilize a shared resource, the associated price is decreased (while keeping it non-negative).

Algorithm 11.3 Distributed Model Predictive Control

-
- 1: Master initializes the prices (Lagrange multipliers) $\mathbf{\Lambda}(k) \succeq 0$.
 - 2: **repeat**
 - 3: Master broadcasts the current prices $\mathbf{\Lambda}(k)$ to all subsystems.
 - 4: Problem (11.1) is solved under the current $\mathbf{\Lambda}(k)$ distributedly by letting each subsystem $i \in \mathcal{N}$ *locally* solve the optimization problem

$$\begin{aligned}
& \text{minimize} && \sum_{\kappa \in \mathcal{K}} \left(\ell_i(\mathbf{x}_i(\kappa+1), \mathbf{u}_i(\kappa)) + \boldsymbol{\lambda}(\kappa)^T \mathbf{C}_i \mathbf{u}_i(\kappa) \right) \\
& \text{subject to} && \mathbf{x}_i(\kappa+1) = \mathbf{A}_i \mathbf{x}_i(\kappa) + \mathbf{B}_i \mathbf{u}_i(\kappa), \quad \forall \kappa \in \mathcal{K} \\
& && \mathbf{x}_i(\kappa+1) \in \mathcal{X}_i, \quad \mathbf{u}_i(\kappa) \in \mathcal{U}_i, \quad \forall \kappa \in \mathcal{K}
\end{aligned} \tag{11.16}$$

where the variables are $\mathbf{x}_i(\kappa+1)$, $\mathbf{u}_i(\kappa)$, $\forall \kappa \in \mathcal{K}$. The solution is denoted $\bar{\mathbf{x}}_i(k+1 : k+N_c)$, $\bar{\mathbf{u}}_i(k : k+N_c-1)$ and the vectors $\mathbf{C}_i \bar{\mathbf{u}}_i(\kappa) \in \mathbb{R}^{n_c}$, $\forall \kappa \in \mathcal{K}$ are determined locally at each subsystem and communicated to the master.

- 5: Master determines the violations $\mathbf{s}(\kappa) \in \mathbb{R}^{n_c}$ of the coupling inequality constraints: $\mathbf{s}(\kappa) = \sum_{i=1}^N \mathbf{C}_i \bar{\mathbf{u}}_i(\kappa) - \mathbf{c}$, $\forall \kappa \in \mathcal{K}$; $\mathbf{S}(k) = \mathbf{s}(k : k+N_c-1) \in \mathbb{R}^{N_c n_c}$ and assigns new prices via projection: $\mathbf{\Lambda}(k) := \max(0, \mathbf{\Lambda}(k) + \alpha \mathbf{S}(k))$.
 - 6: **until** $\max(\mathbf{S}(k)) \leq \epsilon$ or maximum number of iterations reached.
 - 7: Based on the final utilization of the input $\mathbf{u}_i(\kappa)$, $\forall \kappa \in \mathcal{K}$, the master determines limits $\bar{\mathbf{c}}_i$ assuring feasibility of the overall problem and communicates the limits to all subsystems.
 - 8: Each subsystem locally solves problem (11.3) with the additional constraint $\mathbf{C}_i \mathbf{u}_i(\kappa) \leq \bar{\mathbf{c}}_i$, $\forall \kappa \in \mathcal{K}$ and applies the first control input solution.
 - 9: Increase k by one and repeat from 1.
-

11.4 Theoretical Results Availability

In this section we briefly comment on the computational burden of the optimization algorithm and describe under what circumstances the algorithm will converge.

First, we note that the optimization problem of each subsystem in the distributed algorithm (Problem 11.16) is only slightly more complex than if the subsystem couplings were neglected problem (11.16). However, the complexity increases significantly as we are required to solve the distributed optimization problem (11.16) a number of times until convergence. Further we note that the update law of the master (Algorithm 11.3 step 5) requires only a single addition and multiplication operation. The computational burden of the master therefore scales well with the number of subsystems N .

A requirement for Algorithm 11.3 to converge is that we have no duality gap, i.e., the value of the primal and the dual solutions are identical. If the primal problem is convex, it often holds that the primal and dual solutions are identical but additional conditions are necessary to guarantee this. One such condition is Slater's condition [4, p. 226] which states that the primal and dual solutions are identical if the primal problem is convex and there exists a solution to the primal problem that is strictly feasible. In the formulation of problem (11.16), Slater's condition requires that a solution exists such that

$$\mathbf{x}_i(\kappa + 1) \in \mathbf{relint}\mathcal{X}_i, \quad \mathbf{u}_i(\kappa) \in \mathbf{relint}\mathcal{U}_i, \quad \forall \kappa \in \mathcal{K} \quad (11.17)$$

where $\mathbf{relint}\mathcal{X}_i$ denotes the relative interior of \mathcal{X}_i and is a set that contains all points which are not on the edge of \mathcal{X}_i , relative to the smallest set in which \mathcal{X}_i lies [1, p. 448]. Under this assumption, convergence can be guaranteed depending on the choice of step size in the subgradient method, which will be discussed in the following.

In the presented algorithm, a projected subgradient method is used to solve the constrained convex optimization problem 11.15. The subgradient method updates $\mathbf{\Lambda}(k)$ according to

$$\mathbf{\Lambda}(k) := \mathbf{P}(\mathbf{\Lambda}(k) - \alpha \mathbf{g}) \quad (11.18)$$

where \mathbf{P} is a projection of $\mathbf{\Lambda}(k)$ onto the feasible set $\{\mathbf{\Lambda}(k) \in \mathbb{R}^{N_c n_c} \mid \mathbf{\Lambda}(k) \succeq 0\}$ and \mathbf{g} is any subgradient to the dual problem and α is a (constant) step size. Using such constant step size assures that we will converge to a value that lies within some range of the optimum value. If the objective of problem 11.11 is differentiable, i.e., $\ell(\mathbf{x}(\kappa + 1), \mathbf{u}(\kappa))$, $\kappa \in \mathcal{K}$ is differentiable, the subgradient method will indeed converge to the optimum for sufficiently small α [3].

Another option is to let the step size vary with the iteration number j , hereby convergence to the optimal value can be guaranteed also for the case of a non-differentiable objective. One example is a non-summable diminishing step size

$$\lim_{j \rightarrow \infty} \alpha_j = 0, \quad \sum_{j=1}^{\infty} \alpha_j = +\infty \quad (11.19)$$

where α_j is the step size at iteration i ; this will guarantee that the subgradient method converges to the optimum [14, p. 215]. Other step size rules with same convergence result exist.

It is important to note that the subgradient method is chosen due to the fact that this allows us to decouple the problem. Other methods (such as second order methods) can provide much faster convergence than the subgradient method presented here. They are, however, not suitable for the type of decoupling presented in this chapter.

A final note concerns the convergence proofs of dual decomposition algorithms. Dual decomposition algorithms rely on subgradient methods as presented above. Generally, convergence proofs for gradient methods are based on the function value decreasing at each iteration; however, for subgradient methods this is not the case. In subgradient methods, the convergence proofs are generally based on the Euclidian distance to the optimal set by showing under what circumstances this distance will decrease at each iteration [3]. Therefore, the objective value can increase during the iterates in the subgradient method used in the algorithm; however, the distance to the optimal set will decrease at each iteration.

11.5 Application Results Availability

In this section, an application of distributed model predictive control via dual decomposition is presented. The example is taken from a *smart grid* setup where the basic idea is to increase the sustainability and stability of the electrical grid by utilizing the flexibility of the demand side (consumers) in the balancing efforts. Two main ideas of the smart grid concept are

- balancing of production and consumption by moving load temporally,
- avoiding distribution grid congestion by moving load temporally or spatially.

In this example, we address these two topics at an overall level.

Consider a number of balancing responsible parties (BRPs) each responsible for a number of consumers under their jurisdiction; each consumer belongs to exactly one BRP. The BRPs buy energy at the day-ahead electricity market on behalf of the consumers. In the following, we illustrate how BRPs can utilize the flexibility of the consumers under their jurisdiction to minimize the imbalance between the purchased energy and the consumed energy thereby avoiding trading compensating balancing energy at unfavorable prices. Further, we show how the BRPs can be coordinated such that distribution grid congestion is avoided. Due to the very competitive electricity market, the BRPs are not willing to share local information such as objectives and states; therefore we use the dual decomposition approach presented in this chapter to resolve grid congestion. In this way, congestion management can be achieved without information sharing between the BRPs. Finally, we show how the dual decomposition method can be interpreted as a *distribution grid capacity market*. Throughout the following, the notation from the previous section will be used to the extent possible.

Consider a star topology distribution grid (no loops) consisting of n_f distribution lines of limited capacity. A total of N BRPs are active in the distribution grid and BRP number i is responsible for $n_{x,i}$ consumers. The setup is illustrated in Fig. 11.3 and discussed in detail in the sequel.

The $n_{x,i}$ consumers under BRP i are characterized by hourly energy consumptions $\mathbf{u}_i(k) + \tilde{\mathbf{u}}_i(k)$ where $\mathbf{u}_i(k) \in \mathbb{R}^{n_{x,i}}$ is the controllable (flexible) part of the consumption and $\tilde{\mathbf{u}}_i(k) \in \mathbb{R}^{n_{x,i}}$ is an uncontrollable base consumption assuming a sampling time of 1 hour. Due to the flexible consumption, the devices are able to *store* energy. We denote the amount of stored energy $\mathbf{x}_i(k) \in \mathbb{R}^{n_{x,i}}$ for the consumers under BRP i ; this may be energy stored as either heat, cold, energy in a battery, or similar. The stored energy depends on the controllable power consumption

$$\mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_i \mathbf{u}_i(k), \quad (11.20)$$

where $\mathbf{A}_i, \mathbf{B}_i \in \mathbb{R}^{n_{x,i} \times n_{x,i}}$ are diagonal with diagonal elements describing drain losses of each energy storage. The consumers are limited by power and energy constraints

$$0 \leq \mathbf{u}_i(k) + \tilde{\mathbf{u}}_i(k) \leq \mathbf{u}_i^{\max}, \quad \mathbf{x}_i^{\min} \leq \mathbf{x}_i(k) \leq \mathbf{x}_i^{\max} \quad (11.21)$$

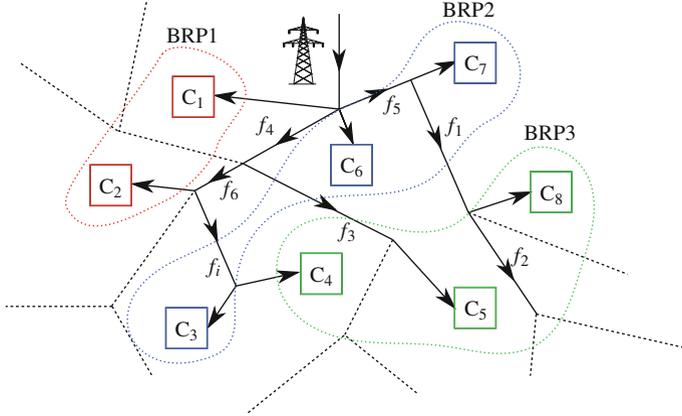


Fig. 11.3 Interconnected consumers under the jurisdiction of different BRPs sharing the same distribution grid (dotted lines indicate that only a small part of the total grid is shown)

where \mathbf{u}_i^{\max} , \mathbf{x}_i^{\min} , $\mathbf{x}_i^{\max} \in \mathbb{R}^{n_{x,i}}$ describe these limits. Consumer models described this way can be found for example in [8].

The consumers are powered through the distribution grid, as illustrated in Figure 11.3. Each BRP will contribute to the loading of the distribution lines. Let $\mathbf{r}_i(k) \in \mathbb{R}_+^{n_f}$ denote the partial flow caused by BRP i to the n_f distribution lines; these partial flows can by flow conversation be described as

$$\mathbf{r}_i(k) = \mathbf{R}_i (\mathbf{u}_i(k) + \tilde{\mathbf{u}}_i(k)) \quad (11.22)$$

where $\mathbf{R}_i \in \mathbb{R}^{n_f \times n_{x,i}}$ is given by

$$(\mathbf{R}_i)_{pq} = \begin{cases} 1 & \text{if consumer } q \text{ under BRP } i \text{ is supplied through link } p, \\ 0 & \text{otherwise.} \end{cases}$$

This simply states that the power to each consumer under BRP i must flow through a unique path of distribution lines; these paths are indicated in the \mathbf{R}_i matrix.

The distribution grid is protected from overcurrents by electrical fuses; hence, the distribution lines are subject to constraints. The total flows $\mathbf{f}(k) \in \mathbb{R}_+^{n_f}$ over the distribution lines and associated fuse limits can be expressed as

$$\mathbf{f}(k) = \sum_{i \in \mathcal{N}} \mathbf{r}_i(k), \quad \mathbf{f}(k) \leq \mathbf{f}^{\max} \quad (11.23)$$

where $\mathbf{f}^{\max}(k) \in \mathbb{R}_+^{n_f}$ denotes the limits of the fuses and \mathcal{N} is the set of all BRPs.

The BRPs buy energy at a day-ahead spot market for each hour of the following day. We denote the energy bought by BRP i at the day-ahead spot market $\mathbf{p}_i(k) \in \mathbb{R}$; this means that BRP i has bought the energy $\mathbf{p}_i(k)$ for the time interval from hour

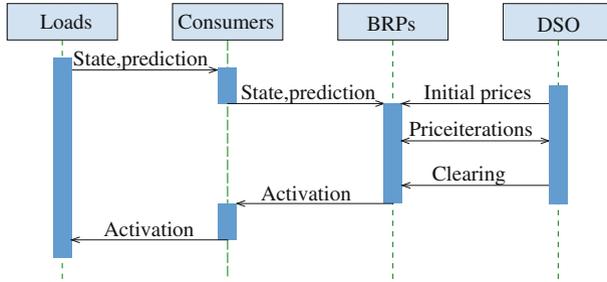


Fig. 11.4 Market interpretation of congestion alleviation via dual decomposition

k to $k + 1$. The objective of each BRP is to minimize the imbalance between the consumed energy $\mathbf{1}^T(\mathbf{u}_i(k) + \tilde{\mathbf{u}}_i(k))$ and the purchased energy $\mathbf{p}_i(k)$, i.e.,

$$\ell_i(\mathbf{u}_i(k)) = \|\mathbf{1}^T(\mathbf{u}_i(k) + \tilde{\mathbf{u}}_i(k)) - \mathbf{p}_i(k)\|_2^2, \tag{11.24}$$

where it is chosen to minimize the imbalance in the two-norm sense and where $\mathbf{1}$ denotes a vector of appropriate dimension with all entries equal to one. By keeping this imbalance small, the BPR minimizes the energy imbalances and thereby avoids trading balancing energy possibly at very unfavorable price.

The modeling reveals that the optimization problem is completely separable among the BRPs except for the coupling via the distribution line capacity constraints 11.23. We apply Algorithm 11.3 to the presented application example and obtain the Algorithm 11.4 when performing receding horizon control with a control horizon N_c and prediction horizon of $N_p = N_c$ samples.

Algorithm 11.4 shows that the congestion management via dual decomposition can be interpreted as a new distribution grid market where each distribution line is associated with a time-varying cost per unit flow. If the lines are not congested, the BRPs are free to use the lines at no cost; however, if congestion occurs, the master will adjust the price on the lines until the congestion is resolved.

The sequence diagram in Fig. 11.4 illustrates how this market can be imagined in an electrical power system setup. First, the individual loads communicate their flexibility (via states and predictions) to the individual consumers. Following, the consumers communicate the flexibility of all their respective loads to the corresponding BRP. Further, the BRPs are provided with initial prices on the distribution grid from the distribution grid operator (DSO) which has the role of the master. Based on this, a price iteration follows where the DSO adjusts the prices until all grid congestions are resolved. When the iteration is completed, the DSO clears the market by communicating final prices and line capacity limits for each BRP. Here it is important to note that the prices at the moment of the market clearing are *real prices* that will determine the economical settlement between the BRPs. From the perspective of a BRP, the prices on the distribution lines reveal the cost that the BRP will have to pay (or be paid) for using more (or less) of the line capacity.

Algorithm 11.4 Congestion Management Example

- 1: Master initializes the prices $\mathbf{\Lambda}(k) \geq 0$, $\mathbf{\Lambda}(k) = \boldsymbol{\lambda}(k : k + N_c - 1)$, where $\boldsymbol{\lambda}(k) \in \mathbb{R}^{n_f}$ and $\lambda_j(k)$ is the price associated with the capacity limit of distribution line j at time sample k .
- 2: **repeat**
- 3: Master broadcasts the current prices $\boldsymbol{\lambda}(\kappa)$, $\forall \kappa \in \mathcal{K}$ to the subsystems.
- 4: Each BRP locally solves the price dependent problem

$$\begin{aligned}
 & \text{minimize} \quad \sum_{\kappa \in \mathcal{K}} \left(\|\mathbf{1}^T(\mathbf{u}_i(\kappa) + \tilde{\mathbf{u}}_i(\kappa)) - \mathbf{p}_i(\kappa)\|_2^2 + \boldsymbol{\lambda}(\kappa)^T \mathbf{r}_i(\kappa) \right) \\
 & \text{subject to} \quad \mathbf{x}_i(\kappa + 1) = \mathbf{A}_i \mathbf{x}_i(\kappa) + \mathbf{B}_i \mathbf{u}_i(\kappa), \quad \forall \kappa \in \mathcal{K} \\
 & \quad \quad \quad 0 \leq \mathbf{u}_i(\kappa) + \tilde{\mathbf{u}}_i(\kappa) \leq \mathbf{u}_i^{\max}, \quad \forall \kappa \in \mathcal{K} \\
 & \quad \quad \quad \mathbf{x}_i^{\min} \leq \mathbf{x}_i(\kappa) \leq \mathbf{x}_i^{\max}, \quad \forall \kappa \in \mathcal{K} \\
 & \quad \quad \quad \mathbf{r}_i(\kappa) = \mathbf{R}_i(\mathbf{u}_i(\kappa) + \tilde{\mathbf{u}}_i(\kappa)), \quad \forall \kappa \in \mathcal{K}
 \end{aligned} \tag{11.25}$$

where the variables are $\mathbf{x}_i(k+1 : k + N_c)$, $\mathbf{u}_i(k : k + N_c - 1)$, $\mathbf{r}_i(k : k + N_c - 1)$. The solution is denoted $\bar{\mathbf{x}}_i(k+1 : k + N_c)$, $\bar{\mathbf{u}}_i(k : k + N_c - 1)$, $\bar{\mathbf{r}}_i(k : k + N_c - 1)$.

- 5: Each BRP reports local partial flows $\bar{\mathbf{r}}_i(\kappa)$ to the master. The master centrally determines line capacity violations $\mathbf{s}(\kappa) = \sum_{i \in \mathcal{N}} \bar{\mathbf{r}}_i(\kappa) - \mathbf{f}^{\max} \in \mathbb{R}^{n_f}$, $\forall \kappa \in \mathcal{K}$ where \mathbf{s}_j is the capacity violation of line j and $\mathbf{S}(k) = \mathbf{s}(k : k + N_c - 1) \in \mathbb{R}^{N_c n_f}$.
 - 6: Master updates prices $\mathbf{\Lambda}(k)$ via projection: $\mathbf{\Lambda}(k) := \max(0, \mathbf{\Lambda}(k) + \alpha \mathbf{S}(k))$. Again notice that this corresponds to increasing the cost on congested lines and reducing the price on lines where there is free capacity; however, always assuring non-negative line prices.
 - 7: **until** $\max(\mathbf{S}(k)) \leq \epsilon$ or maximum number of iterations reached.
 - 8: Master determines limits $\bar{\mathbf{c}}_i \in \mathbb{R}^{n_f}$ and communicates limits and final prices (shadow prices) to the BRPs.
 - 9: Each subsystem locally solves problem (11.25) with the additional constraint $\mathbf{r}_i(\kappa) \leq \bar{\mathbf{c}}_i$ and applies the first control input of the solution.
 - 10: Increase k by one and repeat from 1.
-

Finally, we consider a small numerical example to illustrate the price iteration. The example is kept at a conceptual level to clearly illustrate the concept. The details of the simulation are not presented here but can be found in [2]. Consider two BRPs responsible for one and two consumers, respectively, as presented in Fig. 11.5. The example is constructed with dynamics and objectives fitting the structure of Algorithm 11.4; we assume we are at time sample $k = 1$ and use a control horizon and prediction horizon of $N_c = N_p = 10$. Both BRP 1 and 2 desire to increase the controllable consumption in the first hours, and decrease the consumption in the later hours. If no action is taken, this will violate the capacity constraint on line 2: $\mathbf{f}_2 \leq \mathbf{f}_2^{\max}$. To remedy the problem without information sharing, Algorithm 11.4 is used. The DSO starts by publishing the initial prices $\mathbf{\Lambda}(1) = 0$ where after the two BRPs report back to the DSO how they will utilize the distribution grid under this price, by respectively sending $\bar{\mathbf{r}}_1$ and $\bar{\mathbf{r}}_2$ to the DSO. The DSO discovers that congestion will occur under the initial prices and therefore updates the prices according to $\mathbf{\Lambda}(1) := \mathbf{\Lambda}(1) + \alpha \mathbf{S}(1)$. The top plot of Fig. 11.6 shows the price adjustments, converging to the shadow prices $\mathbf{\Lambda}^*(1)$, optimally resolving the congestion (within the given horizon). The solid line shows the primal objective when using feasible flows, the dashed line is the dual objective, and the dotted line is the optimal value within the control horizon. The lower plot shows the iteration of the prices associated

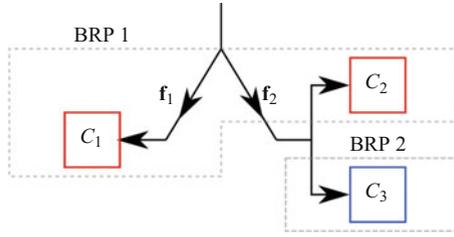


Fig. 11.5 Three consumers under the jurisdiction of two different BRPs sharing the same distribution grid

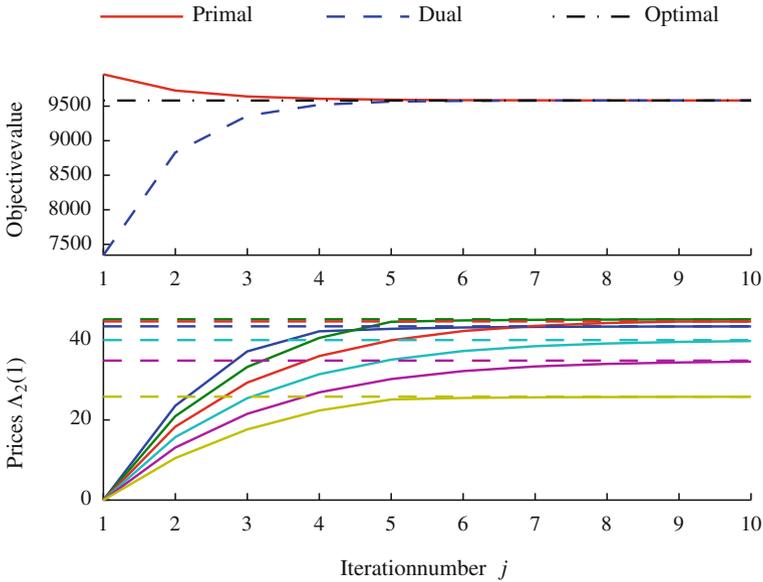


Fig. 11.6 Top: objective value progress. Bottom: convergence of $\lambda_2(1), \dots, \lambda_2(6)$ (solid lines) towards the shadow prices (dashed lines)

with capacity constraint at line 2 from time sample $k = 1$ to $k = 6$; the prices at time $k = 7$ to $k = 10$ remain at zero as there is no congestion at these hours.

The large benefit of resolving congestion management by prices is that the global economical optimum is reached within the control horizon N_c without the need of a centralized optimization. In the presented example, consumer 3 under PRB 2 is a storage of high quality (low drainage) while consumer 2 under BRP 1 is a storage of low quality (high drainage). In this market approach, this results in consumer 3 being the main user of the shared distribution line because BRP 2 is willing to pay a higher price for the line utilization due to the fact that he can profit much from this high quality storage. BRP 1, on the other hand, is willing to decrease the use of his

low quality storage by receiving a payment from BRP 2 as he is not able to profit much from his poor storage.

To illustrate the benefit of using the distribution grid market approach to resolve grid congestion, consider an alternative very simple strategy: congestion is simply resolved by splitting the capacity of the shared line equally among the players sharing the line. In this case, the high quality storage would be used less and the low quality storage would be used more. As a result, a larger amount of energy would be lost due to the higher utilization of the low quality storage; hence, we would not have reached the global economical optimum.

Acknowledgments The work is completed as a part of the project *iPower* and supported by the Danish government via the DSR-SPIR program 10-095378.

References

1. D.P. Bertsekas, *Nonlinear Programming* (Athena Scientific, Belmont, 1999)
2. B. Biegel, P. Andersen, J. Stoustrup, J. Bendtsen, *Congestion management in a smart grid via shadow prices*, in IFAC Power Plant and Power Systems Control, Toulouse, 2012
3. S. Boyd, L. Xiao, A. Mutapic, Notes on subgradient, methods (2007)
4. S. Boyd, L. Vandenberghe, *Convex Optimization* (Cambridge University Press, New York, 2004)
5. S. Boyd, L. Xiao, A. Mutapic, J. Mattingley, Notes on decomposition methods (2008)
6. G.B. Danzig, P. Wolfe, The decomposition algorithm for linear programming. *Econometrica* **29**, 767–778 (1961)
7. M.D. Doan, T. Keviczky, B. De Schutter, A dual decomposition-based optimization method with guaranteed primal feasibility for hierarchical MPC problems, in *Proceedings of the 18th IFAC World Congress*, pp. 392–397, Milan, Italy, August–September (2011)
8. K. Heussen, S. Koch, A. Ulbig, G. Andersson, Unified system-level modeling of intermittent renewable energy sources and energy storage for power system operation. *IEEE Syst. J.* **99**, 1 (2011)
9. H Everett III, Generalized Lagrange multiplier method for solving problems of optimum allocation of resources. *Oper. Res.* **11**(3), 399–417 (1962)
10. F.P. Kelly, A.K. Maulloo, D.K.H. Tan, Rate control in communication networks: shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.* **49**, 237–252 (1998)
11. A. Rantzer, Dynamic dual decomposition for distributed control. In American Control Conference, 2009. ACC '09, pp. 884–888, June 2009
12. Anders Rantzer, On prize mechanisms in linear quadratic team theory, in *Proceedings of 46th IEEE Conference on Decision and Control*, New Orleans, LA, December 2007
13. S. Samar, S. Boyd, D. Gorinevsky, Distributed estimation via dual decomposition, in *IEEE Multi-Conference on Systems and Control*, Denver, CO., USA, Kos, Greece, July 2008
14. N.Z. Shor, *Minimization Methods for Non-Differentiable Functions* (Springer, New York, 1979)