

Technical report 16-025

Multi-agent model predictive control based on resource allocation coordination for a class of hybrid systems with limited information sharing*

R. Luo, R. Bourdais, T.J.J. van den Boom, and B. De Schutter

If you want to cite this report, please use the following reference instead:

R. Luo, R. Bourdais, T.J.J. van den Boom, and B. De Schutter, “Multi-agent model predictive control based on resource allocation coordination for a class of hybrid systems with limited information sharing,” *Engineering Applications of Artificial Intelligence*, vol. 58, pp. 123–133, Feb. 2017.

Multi-Agent Model Predictive Control Based on Resource Allocation Coordination for a Class of Hybrid Systems with Limited Information Sharing

Renshi Luo^{a,*}, Romain Bourdais^b, Ton J.J. van den Boom^a, Bart De Schutter^a

^a*Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands*

^b*CentraleSupélec-IETR UMR 6164, Cesson-Sevigne, France*

Abstract

We develop a multi-agent model predictive control method for a class of hybrid systems governed by discrete inputs and subject to global hard constraints. We assume that for each subsystem the local objective function is convex and the local constraint function is strictly increasing with respect to the local control variable. The proposed multi-agent control method is based on a distributed resource allocation coordination algorithm and it only requires limited information sharing among the local agents of the subsystems. Thanks to primal decomposition of the global constraints, the distributed algorithm can always guarantee global feasibility of the local control decisions, even in the case of premature termination. Moreover, since the control variables are discrete, a mechanism is developed to branch the overall solution space based on the outcome of the resource allocation coordination algorithm at each node of the search tree. Finally, the proposed multi-agent control method is applied to the charging control problem of electric vehicles under constrained grid conditions. This case study highlights the effectiveness of the proposed method.

Keywords: Multi-agent control, Model predictive control, Limited information sharing, Resource allocation, Discrete inputs

1. Introduction

1.1. Multi-agent hybrid systems and their control

Multi-agent systems, like transportation systems, manufacturing systems, power systems, financial systems, are composed of multiple subsystems with interactions (Kantamneni et al., 2015). Multi-agent systems research is facing a variety of challenges (Leitão, 2009), of which a crucial one is to design mechanisms for coordinating agents that have limited information sharing with each other in order to protect confidential information of local subsystems while at the same time still aiming for global performance (Dutta et al., 2005). Typical global control goals for multi-agent systems involve synchronizing motions of agents, maximizing resource utility, and minimizing control costs. In order to achieve globally satisfactory performance, given limited information of other subsystems, the agents need to assist each other to make better decisions about their actions (Cao et al., 2013). Thanks to its straightforward design procedure, where hard system constraints are incorporated directly as inequalities in the formulation of the control problem, model predictive control has shown to be a promising control strategy

for multi-agent systems (Camponogara et al., 2002; Dunbar and Murray, 2006; Negenborn et al., 2008; Scattolini, 2009).

However, the cooperation among agents is made much more difficult when the individual agents have to regulate hybrid subsystems (Christofides et al., 2013) that contain both continuous components and discrete components, such as switches and overrides. In fact, this will result in having to solve mixed-integer programming problems in a distributed way, for which there has not yet been a universally successful algorithm (Frick et al., 2015). Moreover, many system-theoretic concepts and control strategies, such as model predictive control and Artificial Intelligence based control (Cai et al., 2014; Olaru et al., 2004), still require further examination and research in this setting (Mayne et al., 2000).

1.2. Multi-agent model predictive control for hybrid systems with global hard constraints

In this paper, we focus on a class of hybrid systems that are governed by discrete inputs and that are subject to global hard constraints. In particular, each subsystem is characterized by a convex objective function and a strictly increasing constraint function with respect to the local control variable. Besides, each subsystem only shares limited information with the external environment. Actually, such hybrid systems are ubiquitous, and an important example are a group of systems with on/off switches sharing a given amount of resources. More specifically,

*Corresponding author

Email addresses: r.luo@tudelft.nl (Renshi Luo),
Romain.Bourdais@centralesupelec.fr (Romain Bourdais),
a.j.j.vandenboom@tudelft.nl (Ton J.J. van den Boom),
b.deschutter@tudelft.nl (Bart De Schutter)

concrete real-life examples include the charging of a fleet of electric vehicles sharing a given power level provided by the grid, and the operation of a number of appliances sharing a given amount of energy in smart buildings. We aim to develop a multi-agent model predictive control method for such a class of hybrid systems based on a distributed resource allocation coordination algorithm.

A resource allocation is a plan for using the available resources to achieve goals for the future. In principle, such planning may be done by centrally scheduling the actions of the systems that require resources (Huang et al., 2013). However, for reasons of scalability and fast computation, it will not be tractable to schedule the actions of a large number of systems in a centralized way. Actually, the scheduling of the actions of the systems that require resources can be done in a distributed way based on the primal decomposition (Boyd and Vandenberghe, 2004) of the overall problem. More specifically, in primal decomposition, which is naturally applicable to resource allocation scenarios, the allocation of resources can be represented by auxiliary variables and these variables are optimized using a master problem (Palomar and Chiang, 2006). A resource allocation coordination algorithm that is based on the primal decomposition of the overall problem, has already been developed for continuous optimization problems with global capacity constraints by Cohen (1978).

In fact, a multi-agent control method based on the primal decomposition of the overall control problem will always guarantee the global feasibility of local control decisions. However, since all the control variables are discrete in the control setting considered in the paper, issues such as oscillatory behavior of the discrete decision variables could arise if the resource allocation coordination algorithm is applied directly. As a result, the global optimality of the algorithm cannot be guaranteed anymore.

In this paper, a smart mechanism based on the branch-and-bound paradigm (Lawler and Wood, 1966) is developed to improve the solution found when using the resource allocation coordination algorithm only. More specifically, this is achieved by building the search tree according to the outcome of the resource allocation coordination algorithm at each node and by returning the best solution found when the overall method stops.

In the literature, a mechanism has been proposed by Bourdais et al. (2012) to deal with the oscillatory behavior of the discrete decision variables. That mechanism is also based on the branch-and-bound paradigm but it uses a distributed algorithm based on the dual decomposition of the overall problem. Since in the dual decomposition approach constraints are relaxed and accounted for in the objective by using penalties for violations, it cannot be guaranteed that the constraints are always satisfied during iterations. Moreover, a general framework of embedded optimization based on the branch-and-bound paradigm has been presented by Frick et al. (2015) for model predictive control of hybrid systems, which includes the class of hybrid systems considered in this paper. However, that paper focuses

on building a problem-specific branch-and-bound search tree by pre-processing heuristics. So, to the authors' best knowledge, an online and problem-independent algorithm for the control of the class of hybrid systems considered here has not yet been proposed in the literature.

In our previous work (Luo et al., 2015a), we have integrated the resource allocation coordination method into the branch-and-bound paradigm. However, in (Luo et al., 2015a), we assumed the solver to have full information of the overall problem, and we did not consider protecting the confidential information of the local subproblems. In the contrast, the main contribution of this paper consists in reducing information sharing among local control agents, which helps protecting the confidential information of the local subproblems.

With respect to the literature, the multi-agent control method proposed in this paper only requires limited information sharing among local control agents. In addition, it guarantees the global feasibility of local control decisions and it is also able to efficiently search the overall solution space online by making use of the outcome of the resource allocation coordination algorithm at each node of the tree.

1.3. Outline

This paper is organized as follows. In Section 2, the considered class of hybrid systems with subsystems governed by discrete inputs and subject to global hard constraints is formalized. In Section 3, the resource allocation coordination algorithm based on a projected subgradient method is presented. In Section 4, we present the overall proposed multi-agent model predictive control method. In Section 5, we consider charging control of a fleet of electric vehicles as an application example of the proposed method and assess the performance of the proposed method in a simulation study. Section 6 summarizes the results of this paper and presents some ideas for future work.

2. Model predictive control for a class of hybrid systems

In this section, we focus on the control problem formulation of hybrid systems governed by discrete inputs and subject to global hard constraints. Assume that a large-scale hybrid system consists of N subsystems such that:

- each subsystem is controlled by a control agent
- each control agent has a dynamical model of its subsystem
- each control agent has to solve its local problem
- each agent does not have any information of the models and the local control problems of other subsystems
- subsystems together have to satisfy global hard constraints

2.1. Model of subsystem dynamics

Let the dynamics of subsystem i be given by the following deterministic discrete-time model with discrete inputs:

$$x_{i,k+1} = A_i x_{i,k} + B_i u_{i,k} \quad (1)$$

$$y_{i,k} = C_i x_{i,k} + D_i u_{i,k} \quad (2)$$

where at time step k , for subsystem i , $x_{i,k} \in \mathbb{R}^{n_{i,x}}$ is the local continuous state, $y_{i,k} \in \mathbb{R}^{n_{i,y}}$ the local continuous output, $u_{i,k} \in U_{i,k} \subset \mathbb{R}^{n_{i,u}}$ the local discrete input with $u_{i,k,v} \in U_{i,k,v}$, $U_{i,k,v} \subset \mathbb{R}$ a finite set of scalar values, and $U_{i,k} = U_{i,k,1} \times U_{i,k,2} \times \dots \times U_{i,k,n_{i,u}}$, and $A_i \in \mathbb{R}^{n_{i,x} \times n_{i,x}}$, $B_i \in \mathbb{R}^{n_{i,x} \times n_{i,u}}$, $C_i \in \mathbb{R}^{n_{i,y} \times n_{i,x}}$ and $D_i \in \mathbb{R}^{n_{i,y} \times n_{i,u}}$.

2.2. Model predictive control of a single subsystem

In model predictive control, at each control cycle, the control agent determines its local control input by computing the optimal control input sequence over a finite prediction horizon of N_p steps according to an objective function describing the control goals, subject to a model of the subsystem and operational constraints. After that, the control agent applies the first control input in that sequence to its subsystem and waits until the next control cycle starts. For the sake of simplicity of notation, in the following, a bold variable is used to denote the compact expression of variables over the prediction horizon, e.g., $\mathbf{x}_{i,k} = [x_{i,k}^T \ x_{i,k+1}^T \ \dots \ x_{i,k+N_p-1}^T]^T$.

Assume that the maximum amount of resources $\theta_{i,k+l} \in \mathbb{R}$ available to subsystem i for $l = 0, \dots, N_p - 1$ is given. Then control agent i makes a measurement of the local state $x_{i,k}$ of its subsystem at time step k . After elimination of $x_{i,k+l}$ and $y_{i,k+l}$ for $l = 1, 2, \dots, N_p$, the following optimization problem is then solved by control agent i :

$$\begin{aligned} \min_{\mathbf{u}_{i,k}} \quad & \sum_{l=0}^{N_p-1} J_i(u_{i,k+l}, x_{i,k}) \\ \text{subject to} \quad & G_i(u_{i,k+l}) \leq \theta_{i,k+l} \\ & u_{i,k+l} \in U_{i,k+l} \\ & \text{for } l = 0, \dots, N_p - 1 \end{aligned} \quad (3)$$

where $J_i : U_{i,k} \times \mathbb{R}^{n_{i,x}} \rightarrow \mathbb{R}$ is assumed to be a convex function w.r.t. $u_{i,k+l}$ for a given $x_{i,k}$ that gives the cost per prediction step and $G_i : U_{i,k} \rightarrow \mathbb{R}$ is assumed to be a monotonic strictly increasing function w.r.t. $u_{i,k+l}$ that gives the amount of resources consumed by subsystem i per prediction step.

2.3. Global constraints

Let r_{k+l} denote the total amount of resources available for all the subsystems at time step $k+l$. Then the global constraints over the prediction horizon are given by

$$\sum_{i=1}^N \theta_{i,k+l} = r_{k+l}, \quad \text{for } l = 0, \dots, N_p - 1 \quad (4)$$

2.4. Combined overall control problem

We aim to achieve global optimal system performance. Therefore, we define the combined overall control problem by aggregating the local control problems and including the global constraints (4), i.e.,

$$\begin{aligned} \min_{\mathbf{u}_k, \boldsymbol{\theta}_k} \quad & \sum_{i=1}^N \sum_{l=0}^{N_p-1} J_i(u_{i,k+l}, x_{i,k}) \\ \text{subject to} \quad & G_i(u_{i,k+l}) \leq \theta_{i,k+l} \\ & u_{i,k+l} \in U_{i,k+l} \\ & \sum_{i=1}^N \theta_{i,k+l} = r_{k+l} \\ & \text{for } i = 1, \dots, N \text{ and } l = 0, \dots, N_p - 1 \end{aligned} \quad (5)$$

where $\mathbf{u}_k = [\mathbf{u}_{1,k}^T \ \mathbf{u}_{2,k}^T \ \dots \ \mathbf{u}_{N,k}^T]^T$ and $\boldsymbol{\theta}_k = [\theta_{1,k}^T \ \theta_{2,k}^T \ \dots \ \theta_{N,k}^T]^T$.

Since each local agent does not have information of the local models and the local control problems of other agents, none of the agents has full information of the overall problem. Therefore, it is not possible to solve the combined overall control problem (5) in a centralized way. To deal with this issue, in the following sections, we will develop a multi-agent control method to solve the combined overall control problem by introducing a resource allocation coordinator that only requires very limited information from the local agents.

3. Resource allocation coordination

Before presenting the overall multi-agent model predictive control method, in the section, we first describe the resource allocation coordination algorithm on which the overall control method will be based. In the resource allocation coordination algorithm, the maximum amount of resources allocated to each subproblem is represented by an auxiliary variable and then the coordination is achieved by solving a master problem that optimizes all the auxiliary variables (Palomar and Chiang, 2006).

3.1. Primal decomposition

By dropping $x_{i,k}$ and the time step k and l , problem (5) can be simplified as

$$\begin{aligned} \min_{\mathbf{u}, \boldsymbol{\theta}} \quad & \sum_{i=1}^N J_i(u_i) \\ \text{subject to} \quad & G_i(u_i) \leq \theta_i, \quad i = 1, \dots, N \\ & u_i \in U_i, \quad i = 1, \dots, N \\ & \sum_{i=1}^N \theta_i = r \end{aligned} \quad (6)$$

Now, let us define

$$p_i(\theta_i) = \min_{u_i \in U_i, G_i(u_i) \leq \theta_i} J_i(u_i) \quad (7)$$

Then, problem (6) can be written (Boyd and Vandenberghe, 2004) as

$$\begin{aligned} \min_{\theta} \quad & \sum_{i=1}^N p_i(\theta_i) \\ \text{subject to} \quad & \sum_{i=1}^N \theta_i = r \end{aligned} \quad (8)$$

Problem (8) is called the master problem.

3.2. Optimization algorithm

Actually, if all optimization variables u_i are continuous, the master problem (8) can be solved efficiently by using a subgradient method, which is a simple iterative method for solving convex optimization problems (Shor, 2012). More specifically, given a convex problem with decision variable u , classical subgradient methods search for the solution to the problem by using the following iteration:

$$u^{(z+1)} = \Pi\left(u^{(z)} - \zeta^{(z)} h(u^{(z)})\right)$$

where z denotes the iteration step, $h(u^{(z)})$ denotes a subgradient of the objective function of the problem at $u^{(z)}$, $\zeta^{(z)}$ denotes the step size at step z , and $\Pi(\cdot)$ denotes the projection onto the constrained solution space.

It can be derived that in problem (7), a subgradient of $p_i(\theta_i)$ at θ_i is given by $-\lambda_i$, with λ_i the corresponding Lagrange multiplier to the constraint $G_i(u_i) \leq \theta_i$ (Bertsekas, 1999, Chapter 6.4.2). In particular, the projected subgradient method is given in (Cohen, 1978) as

$$\theta_i^{(z+1)} = \theta_i^{(z)} + \xi^{(z)} \left(\lambda_i^{(z)} - \frac{1}{N} \sum_{j=1}^N \lambda_j^{(z)} \right) \quad (9)$$

where $\xi^{(z)}$ is a diminishing step size that satisfies

$$\xi^{(z)} > 0, \quad \sum_{z=1}^{+\infty} \xi^{(z)} = +\infty, \quad \sum_{z=1}^{+\infty} (\xi^{(z)})^2 < +\infty \quad (10)$$

Note that in (9), $-\lambda_i^{(z)}$ is used as the subgradient of $p_i(\cdot)$ at $\theta_i^{(z)}$ and a projection is used to guarantee that the constraint $\sum_{i=1}^N \theta_i^{(z)} = r$ is satisfied for all iterations.

In fact, the corresponding λ_i to the constraint $G_i(u_i) \leq \theta_i$ in problem (7) can in general be computed by

$$\lambda_i = \begin{cases} -\frac{d_{i,\mathbf{J}}^T \cdot d_{i,\mathbf{G}}}{d_{i,\mathbf{G}}^T \cdot d_{i,\mathbf{G}}}, & \text{if } -\frac{d_{i,\mathbf{J}}^T \cdot d_{i,\mathbf{G}}}{d_{i,\mathbf{G}}^T \cdot d_{i,\mathbf{G}}} > 0 \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

where $d_{i,\mathbf{J}}$ and $d_{i,\mathbf{G}}$ are respectively the derivatives of $J_i(\cdot)$ and $G_i(\cdot)$ w.r.t. u_i^* with

$$u_i^* = \arg \min_{u_i \in U_i, G_i(u_i) \leq \theta_i} J_i(u_i)$$

The proof of (11) is given in Appendix A. Finally, the explicit modified resource allocation coordination algorithm for problems with discrete optimization variables is presented in Algorithm 1.

Inputs: r, N, ξ and J_i, G_i, U_i for all i

- i) Initialize $\theta_i^{(1)}$ for all i and set $z = 1$.
- ii) At iteration z , each agent i solves its local problem

$$u_i^{*,(z)} = \arg \min_{u_i \in U_i, G_i(u_i) \leq \theta_i^{(z)}} J_i(u_i)$$

and obtains $\lambda_i^{(z)}$ using (11). Note that U_i is a set of discrete values.

- iii) Update $\theta_i^{(z+1)}$ for all i using (9).

- iv) Stop if $|\theta_i^{(z+1)} - \theta_i^{(z)}| \leq \epsilon$ for all i or if the maximum number of iterations is reached; otherwise set $z \leftarrow z + 1$ and go back to step ii).

Outputs: $u_i^{*,(z)}$ and $\theta_i^{(z)}$

Algorithm 1: Modified resource allocation coordination algorithm for problems with discrete optimization variables

3.3. Problems arising when applied to optimization problems with discrete decision variables

Actually, if the problem (6) is strictly convex, the global optimum is always attained by using the resource allocation coordination algorithm presented above. However, even if $J_i(\cdot)$ and $G_i(\cdot)$ are strictly convex functions, the overall problem is still not convex if u_i is a discrete variable. In fact, in that case, if the resource allocation coordination algorithm is directly applied to the problem, the discrete decision variables may exhibit oscillatory behavior (i.e. the computed optimal values oscillate from one iteration to the next), and hence the global optimum may not be attained. To be more specific, a simple numerical example is given next to show the problem of directly applying the resource allocation coordination algorithm to an optimization problem with discrete decision variables.

Consider the following resource allocation problem:

$$\begin{aligned} \min_{u_1, u_2} \quad & (u_1 - 3)^2 + 2(u_2 - 2)^2 \\ \text{subject to} \quad & u_1 \in \{-1.5, 1.2, 2.4, 3.4, 4.5\} \\ & u_2 \in \{-1, 0.6, 2.5, 3.8, 4.2\} \\ & u_1 + u_2 \leq 4.5 \end{aligned}$$

The global optimum to this problem is $u_1^* = 1.2$ and $u_2^* = 2.5$. Figure 1 shows the behavior of the values of the discrete decision variables when the resource allocation coordination algorithm is directly applied to this simple example. It is clearly shown that the value of u_1 converges to 1.2 during the iterations while that of u_2 oscillates between 0.6 and 2.5. Therefore, the global optimum is only part of the oscillation.

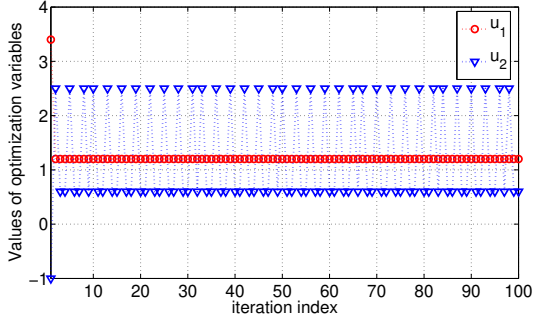


Figure 1: Example of oscillatory behavior in the case of discrete optimization variables

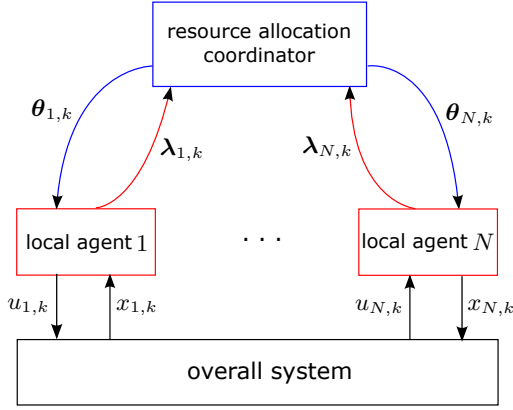


Figure 2: Communication structure of the multi-agent model predictive control method

4. Multi-agent model predictive control method based on resource allocation coordination

Although the values of the control decision variables may oscillate when the resource allocation coordination algorithm is applied to the combined overall control problem (5), the oscillations of the values of the control decision variables can be used as a guideline for branching the overall solution space. In this section, we develop a multi-agent control method for the combined overall control problem (5) by integrating the resource allocation coordination algorithm into a search tree building mechanism for the overall solution space. The communication structure used in the method is shown in Figure 2.

4.1. Resource allocation coordinator

The resource allocation coordinator is in charge of allocating the resource to subsystems. More specifically, receiving $\lambda_{i,k}^{(z)}$ for all i from local agents at iteration z , the resource allocation coordinator updates the amount of resource allocated to a subsystem i by

$$\theta_{i,k}^{(z+1)} = \theta_{i,k}^{(z)} + \xi^{(z)} \left(\lambda_{i,k}^{(z)} - \frac{1}{N} \sum_{j=1}^N \lambda_{j,k}^{(z)} \right), \quad \text{for } i = 1, \dots, N \quad (12)$$

with diminishing step size $\xi^{(z)}$ satisfying (10).

4.2. Local agent

Each local agent focuses on solving its local control problem and communicating the Lagrange multiplier associated with the local constraint to the coordinator. More specifically, given the allocated maximal amount $\theta_{i,k}^{(z)}$ of resources from the resource allocation coordinator, each local agent i solves its local control problem (3) to obtain $u_{i,k}^{*,(z)}$ and then calculates

$$\lambda_{i,k+l}^{(z)} = \begin{cases} -\frac{d_{i,k+l,\mathbf{J}}^T \cdot d_{i,k+l,\mathbf{G}}}{d_{i,k+l,\mathbf{G}}^T \cdot d_{i,k+l,\mathbf{G}}}, & \text{if } -\frac{d_{i,k+l,\mathbf{J}}^T \cdot d_{i,k+l,\mathbf{G}}}{d_{i,k+l,\mathbf{G}}^T \cdot d_{i,k+l,\mathbf{G}}} > 0 \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

where $d_{i,k+l,\mathbf{J}}$ and $d_{i,k+l,\mathbf{G}}$ for $l = 0, \dots, N_p - 1$ are respectively the derivatives of $J_i(\cdot)$ and $G_i(\cdot)$ w.r.t $u_{i,k+l}$ at $u_{i,k+l}^{*,(z)}$.

4.3. Multi-agent control procedure

The overall multi-agent control framework can be considered as a two-level control hierarchy, consisting of:

- the lower-level resource allocation coordination subprocedure
- the higher-level subprocedure to branch the solution space and to improve the current best solution to the overall control problem

More specifically, the overall procedure of the multi-agent control method is shown in Figure 3.

4.3.1. Lower-level subprocedure

The resource allocation coordination subprocedure (involving the coordinator and the local agents), the schematic representation of which is shown in Figure 2, can be explicitly described as follows:

- The coordinator proposes an initial plan for allocating resources to local agents and communicates the corresponding proposed amount $\theta_{i,k}^{(1)}$ of resources to each local agent i .
- At iteration z , each local agent i receives a proposed value $\theta_{i,k}^{(z)}$ from the coordinator and evaluates it by solving its local control problem (3). After that, agent i determines $\lambda_{i,k}^{(z)}$ using (13) and then communicates it to the coordinator, indicating how much the agent would benefit from extra resources.
- At iteration z , based on the proposed value $\theta_{i,k}^{(z)}$ communicated to each local agent i and $\lambda_{i,k}^{(z)}$ received from each local agent i , the coordinator proposes an updated plan of resource allocation $\theta_{i,k}^{(z+1)}$ for all agents using (12) and communicates it to the local agents.

iv) During the iterations, the evolution of local control decisions is checked (by the local agents). Depending on whether local control decisions oscillate, two cases can occur:

- In the case where oscillation of local control decisions is detected (see Remark 1), stop the lower-level subprocedure and return:
 - * the best proposal of resource allocation (with the lowest sum of the cost functions of local agents) so far
 - * the index of the local control decision variable for which oscillation is detected
 - * the discrete values between which the given control decision variable oscillates
- In the case where no oscillation of local control decisions is detected:
 - * if the allowed maximum number of iterations is reached or $|\theta_{i,k+l}^{(z+1)} - \theta_{i,k+l}^{(z)}| \leq \epsilon$ holds for all i and for all l , stop the subprocedure and return the best proposal of resource allocation (with the lowest sum of the cost functions of local agents) so far
 - * otherwise, set $z \leftarrow z + 1$ and go to step ii)

Remark 1.

The oscillation of a discrete decision variable $u_{i,k+l,v}$ is characterized by the following condition, the proof of which can be found in the technical report (Luo et al., 2015b):

$$u_{i,k+l,v}^{*,(z+1)} \neq u_{i,k+l,v}^{*,(z)}, \quad \text{sgn}(\Delta\theta_{i,k+l}^{(z+1)}) \neq \text{sgn}(\Delta\theta_{i,k+l}^{(z)}) \quad (14)$$

where $\Delta\theta_{i,k+l}^{(z+1)} = \theta_{i,k+l}^{(z+1)} - \theta_{i,k+l}^{(z)}$ and $\Delta\theta_{i,k+l}^{(z)} = \theta_{i,k+l}^{(z)} - \theta_{i,k+l}^{(z-1)}$. Therefore, we diagnose the oscillation of discrete decision variables by detecting the condition (14) for each i, l and v .

4.3.2. Higher-level subprocedure

The procedure to improve the current best solution, the schematic diagram of which is shown in Figure 3, is given by:

- a) call the lower-level subprocedure
- b) update the best solution to the overall problem found so far by comparing the corresponding values of objective function to the current best solution and the newly found solution by the lower-level subprocedure
- c) depending on whether local control decisions oscillate
 - if $u_{i,k+l,v}$ is found to oscillate between $\alpha \in U_{i,k+l,v}$ and $\beta \in U_{i,k+l,v}$, separate $U_{i,k+l,v}$ into two sets:

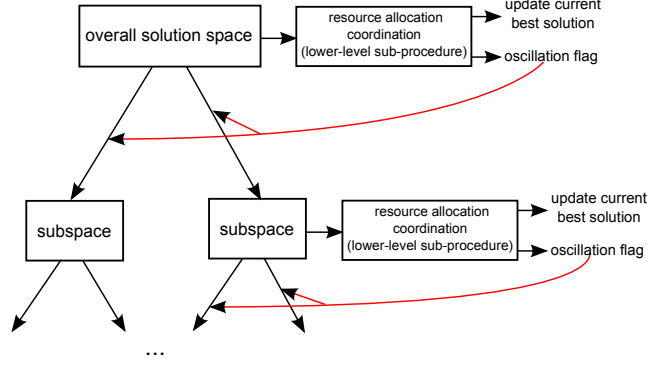


Figure 3: Schematic diagram of the higher-level subprocedure

Table 1: Nomenclature of the mathematical symbols

Symbol	Definition
N	number of subsystems
N_p	prediction horizon
$x_{i,k}$	local continuous state of subsystem i
$u_{i,k}$	local discrete input of subsystem i
$y_{i,k}$	local continuous output of subsystem i
$\theta_{i,k}$	amount of resources assigned to subsystem i
r_k	amount of resources available to the overall system
U_i	finite set of admissible control actions of subsystem i
J_i	local objective function of subsystem i
G_i	local constraint function of subsystem i
λ_i	Lagrange multiplier corresponding to local constraint
z	iteration step counter in the resource allocation coordination algorithm
$\xi^{(z)}$	diminishing step size in the iteration of the resource allocation coordination algorithm

- * set $U_{i,k+l,v}^{(1)} \leftarrow U_{i,k+l,v} \setminus \{\gamma \mid \gamma \in U_{i,k+l,v}, \gamma \leq \alpha\}$
- * set $U_{i,k+l,v}^{(2)} \leftarrow U_{i,k+l,v} \setminus \{\gamma \mid \gamma \in U_{i,k+l,v}, \gamma > \alpha\}$

and solve the two branches in parallel

- if no oscillation of local discrete decision variables is detected, stop and return the best solution found so far
- d) when the computation time or number of information exchanges between the coordinator and the local agents reaches a predefined upper bound or $U_{i,k+l,v}$ for all i, l, v in all calls of the lower-level sub-procedure has only one single element, stop the procedure.

Note that all important symbols used in the multi-agent model predictive control method have been listed in Table 1.

5. Charging control of electric vehicles

As an application example of the developed multi-agent model predictive control method, in this section we address the charging control of a fleet of electric vehicles under constrained grid conditions.

Due to their higher energy efficiency and lower emission of pollutants, electric vehicles are used more and more. Charging this increasing number of electric vehicles will inevitably cause additional load to the electrical power distribution grid (Fernandez et al., 2011; Hu et al., 2015). Therefore, a smart charging control strategy that balances the charging demands of electric vehicles is highly preferred by the distribution grid operators. So far, intelligent charging control of electric vehicles have been the topic of many researches (Mal et al., 2013; Nguyen et al., 2015; Saber and Venayagamoorthy, 2009) and have been addressed by using distributed integer linear optimization (Vujanic et al., 2016), sequential quadratic programming (Clement-Nyns et al., 2010; Hajimiragha et al., 2010), dynamic programming (Han et al., 2010), and heuristic methods (Saber and Venayagamoorthy, 2011). In this section, we assume that each electric vehicle is equipped with a charging controller and each controller only shares limited information with the external environment, and apply the developed multi-agent model predictive control method to the charging of a fleet of electric vehicles under constrained grid conditions.

We focus on the optimal charging control of a fleet of electric vehicles at a charging station within a given time period, e.g. a day. We assume that there is a charging point for each vehicle in the station. Given the profile of the electricity price, the arrival and the departure times of all electric vehicles at the charging station, and the maximum power limit provided by the grid, we aim to charge all electric vehicles up to the required levels while minimizing the total cost of electricity use.

5.1. Definitions

We define k as the discrete-time step counter, T as the length of the simulation time interval, with a typical value of 15 minutes, and k_d as the last step of the overall charging period. Then define N_v as the total number of electric vehicles under consideration. Define $T_{i,\text{arrival}}$, $T_{i,\text{departure}}$ as the arrival time and the departure time of electric vehicle i at the charging station, respectively. Without loss of generality, we assume $T_{i,\text{arrival}}$ and $T_{i,\text{departure}}$ are integer multiples¹ of T . Define $s_{i,k}$ as the state of charge of electric vehicle i at time kT and s_i^{req} as the required state of charge of electric vehicle i before departing from the charging station. Define C_i^p as the capacity of the battery of electric vehicle i and $d_{i,\text{tol}}$ as the allowed tolerance

¹If an electric vehicle arrives earlier or departs later than a sampling time instant, it will not be charging in the partial time slot within which it arrives or departs.

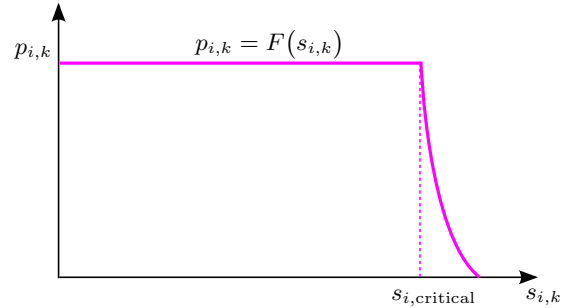


Figure 4: Electric vehicle charging using the CPCV option. With this option, the vehicle is first charging with constant power until the critical state of charge s_{critical} is reached. After that, it is charged with constant voltage until its battery is fully charged.

on the difference between the state of charge of electric vehicle i at its departure time and s_i^{req} . Besides, we assume the charging power of each electric vehicle within a simulation interval is constant and define $p_{i,k}$ as the amount of power consumed by electric vehicle i in the time interval $[kT, (k+1)T)$. Finally, we define $u_{i,k}$ as the binary control variable indicating whether electric vehicle i is charging in the time interval $[kT, (k+1)T)$.

5.2. Model of the charging of an individual electric vehicle

First, the amount of power consumed by an electric vehicle i within the time interval $[kT, (k+1)T)$ is given by

$$p_{i,k} = \begin{cases} F(s_{i,k}), & \text{if } u_{i,k} = 1 \\ 0, & \text{if } u_{i,k} = 0 \end{cases} \quad (15)$$

where the function $F(\cdot)$ describes how the amount of power consumed by electric vehicle i depends on its state of charge.

Next, the state of charge of electric vehicle i is updated by

$$p_{i,k} = F(s_{i,k}) \cdot u_{i,k} \quad (16)$$

$$s_{i,k+1} = s_{i,k} + \frac{p_{i,k} \cdot T}{C_i^p} \quad (17)$$

There are in general two charging options available for electric vehicle chargers (Marra et al., 2012), namely Constant Current-Constant Voltage (CCCV) and Constant Power-Constant Voltage (CPCV). In this paper, we assume all electric vehicles are charged using the CPCV option. More specifically, the profile of the function $F(\cdot)$ for the CPCV option is shown in Figure 4. Moreover, in order to make the model of the charging of an electric vehicle clearer, Figure 5 shows the dynamics of the state of charge of an electric vehicle under charging control.

5.3. Global constraints

At any time, the total amount of power consumed by all electric vehicles must not exceed the maximum amount

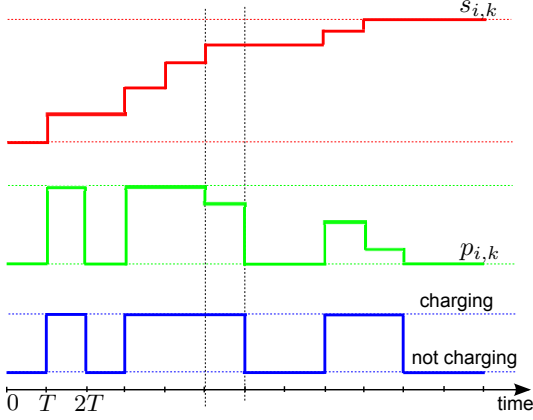


Figure 5: Charging profile of an electric vehicle under charging control

of power that can be provided by the grid. Therefore, the constraints imposed by the capacity of the grid on all electric vehicles are given by

$$\sum_{i=1}^{N_v} p_{i,k} \leq P_{k,\max}, \quad k = 1, \dots, k_d \quad (18)$$

where $P_{k,\max}$ denotes the maximum power limit provided by the grid, which can be steady or time-variant.

5.4. Charging cost

If the profile of the price of electricity is given, the total cost of charging all electric vehicles is given by:

$$J = \sum_{i=1}^{N_v} \sum_{k=k_{i,\text{arrival}}}^{k_{i,\text{departure}}-1} p_{i,k} \cdot T \cdot c_k \quad (19)$$

where c_k denotes the price of electricity in the time interval $[kT, (k+1)T)$ and

$$k_{i,\text{arrival}} = \frac{T_{i,\text{arrival}}}{T}, \quad k_{i,\text{departure}} = \frac{T_{i,\text{departure}}}{T}$$

5.5. Problem formulation

Normally, the state of charge of a battery is limited to the interval $[0.2, 0.9]$ (Marra et al., 2012). This mainly relates to battery life time aspects: charging the remaining 10%-20% before fully charged has shown to result in quicker battery degradation (Marra et al., 2012). According to (Marra et al., 2012), if the CPCV option is used, an electric vehicle can be charged with constant power up to more than 90% of the capacity of its onboard battery. As we adopt the CPCV option, if we define $s_{i,\text{critical}}$ as the critical state of charge of electric vehicle i , according to (Marra et al., 2012), we have $s_{i,\text{critical}} > 0.9$ for all i . In order to make the life time of a battery longer, the state of charge of the battery should be limited at most to 0.9. Therefore, we assume that the user always selects s_i^{req} such

that $s_i^{\text{req}} \leq 0.9$. Hence, assuming s_i^{req} is sufficiently smaller than $s_{i,\text{critical}}$ and T is not too large, we can assume that

$$s_{i,k} \leq s_{i,\text{critical}} \quad (20)$$

holds for all k and for all i . Then, the function $F(\cdot)$ with the CPCV option can be simplified as

$$F(s_{i,k}) = \begin{cases} p_{i,\text{constant}}, & \text{if } s_{i,k} \leq s_{i,\text{critical}} \\ 0, & \text{otherwise} \end{cases}$$

The charging model (16)-(17) of an electric vehicle can then be simplified to

$$s_{i,k_{i,\text{departure}}} = s_{i,k_{i,\text{arrival}}} + \sum_{k=k_{i,\text{arrival}}}^{k_{i,\text{departure}}-1} \frac{p_{i,\text{constant}} \cdot u_{i,k} \cdot T}{C_i^{\text{p}}}$$

and the constraint $s_{i,k_{i,\text{departure}}} \geq s_i^{\text{req}} - d_{i,\text{tol}}$, which presents the charging requirement for each electric vehicle i , can be written as

$$\frac{s_i^{\text{req}} - s_{i,k_{i,\text{arrival}}} - d_{i,\text{tol}}}{p_{i,\text{constant}} \cdot T} \cdot C_i^{\text{p}} \leq \sum_{k=k_{i,\text{arrival}}}^{k_{i,\text{departure}}-1} u_{i,k} \quad (21)$$

Further, since all the control variables are all binary values, the constraint (21) can be rewritten as

$$\sum_{k=k_{i,\text{arrival}}}^{k_{i,\text{departure}}-1} u_{i,k} \geq m_i \quad (22)$$

where m_i is an integer constant given by

$$m_i = \text{ceiling} \left(\frac{s_i^{\text{req}} - s_{i,k_{i,\text{arrival}}} - d_{i,\text{tol}}}{p_{i,\text{constant}} \cdot T} \cdot C_i^{\text{p}} \right)$$

Finally, including for all i the constraint (22) via penalty term to the objective function with a sufficiently large weight β_i and a normalization factor $1/(k_{i,\text{departure}} - k_{i,\text{arrival}})$, the optimal charging control problem of a fleet of electric vehicles under constrained grid conditions can be formulated as

$$\min_{\mathbf{u}} \sum_{i=1}^{N_v} \left(\frac{1}{J_{i,\text{cost,typical}}} \sum_{k=k_{i,\text{arrival}}}^{k_{i,\text{departure}}-1} p_{i,\text{constant}} \cdot T \cdot c_k \cdot u_{i,k} + \frac{\beta_i}{k_{i,\text{departure}} - k_{i,\text{arrival}}} \left| m_i - \sum_{k=k_{i,\text{arrival}}}^{k_{i,\text{departure}}-1} u_{i,k} \right| \right) \quad (23)$$

subject to $\sum_{i=1}^{N_v} p_{i,\text{constant}} \cdot u_{i,k} \leq P_{k,\max}, \quad k = 1, \dots, k_d$

where $\mathbf{u} = [\mathbf{u}_1^{\text{T}} \ \mathbf{u}_2^{\text{T}} \ \dots \ \mathbf{u}_{N_v}^{\text{T}}]^{\text{T}}$ with $\mathbf{u}_i = [u_{i,k_{i,\text{arrival}}} \ \dots \ u_{i,k_{i,\text{departure}}-1}]^{\text{T}}$, and $J_{i,\text{cost,typical}}$ is the typical value for the charging cost of electric vehicle i , which is used to

Table 2: Data for 5 electric vehicles charging with $P^{\max} = 8\text{kW}$ in Case 1. The unit of C^P is kWh and the unit of p_{constant} is kW.

i	k_{arrival}	$k_{\text{departure}}$	s_{initial}	s^{req}	C^P	p_{constant}
1	3	6	0.60	0.80	9	3.5
2	1	4	0.35	0.45	7.1	2.5
3	2	5	0.40	0.60	8	3
4	5	10	0.60	0.90	8.5	2.7
5	4	8	0.50	0.70	7.5	3.2

normalize the real charging cost of electric vehicle i . Alternatively, $J_{i,\text{cost,typical}}$ can be given by cost related to the average or maximum number of steps needed to charge from the current state of charge to the required level. Note that problem (23) is a specific case of the general combined control problem (5). Therefore, the proposed multi-agent control method is applicable to the optimal charging control problem of electric vehicles.

5.6. Numerical simulation study

We will now apply the proposed multi-agent control method to the charging control problem (23) of electric vehicles. In this simulation study, we consider two cases where respectively 5 and 20 electric vehicles need to be charged. With the simpler Case 1, we aim to show that the proposed multi-agent control method finds the globally optimal solution while with the more complicated Case 2, we aim to show the flexibility and effectiveness of the proposed method when limiting the computation and communication budget. The information of all vehicles in both cases is summarized in Tables 2 and 3, respectively.

The total number of binary control variables in Case 1 is $\sum_{i=1}^5 (k_{i,\text{departure}} - k_{i,\text{arrival}}) = 18$ while in Case 2 it is $\sum_{i=1}^{20} (k_{i,\text{departure}} - k_{i,\text{arrival}}) = 89$. The parameters used in the simulations are $T = 15$ min, $d_{i,\text{tol}} = 0.02$, and $\beta_i = 200$ for all i . Besides, for each electric vehicle i , $J_{i,\text{cost,typical}} = (k_{i,\text{departure}} - k_{i,\text{arrival}}) \cdot T \cdot \bar{c} \cdot p_{i,\text{constant}}$ with \bar{c} denoting the average price of electricity for the simulated period; the maximum number of iterations in the resource allocation coordination algorithm is set to 1000 and $\epsilon = 0.001$; the simulated period is 165 minutes and the profiles of the electricity price for Case 1 and Case 2 are shown in Figures 6 and 7. The simulations are performed using Matlab 2013b on a cluster computer consisting of 4 blades with 2 eight-core 3.3 GHz E5-2643 processors, and 64 GiB memory per blade.

In the simulation for Case 1, no limit is imposed on the maximum computation time or on the maximum number of information exchanges between the coordinator and the local control agents. The simulation results are summarized in Table 4. Note that the overall problem is a mixed integer linear programming problem and therefore it can be solved efficiently in a centralized way by using state-of-the-art solvers like CPLEX, GUROBI, MOSEK

Table 3: Data for 20 electric vehicles charging with $P^{\max} = 36\text{kW}$ in Case 2. The unit of C^P is kWh and the unit of p_{constant} is kW.

i	k_{arrival}	$k_{\text{departure}}$	s_{initial}	s^{req}	C^P	p_{constant}
1	3	6	0.60	0.80	9	3.5
2	1	4	0.35	0.45	7.1	2.5
3	2	5	0.40	0.60	8	3
4	5	10	0.60	0.90	8.5	2.7
5	4	8	0.50	0.70	7.5	3.2
6	3	9	0.30	0.50	7.8	3.5
7	2	7	0.45	0.75	8.3	2.9
8	1	5	0.60	0.80	8.6	3.1
9	2	6	0.40	0.65	7.7	3.4
10	6	10	0.50	0.75	8.8	3.7
11	4	8	0.65	0.85	8.6	3.2
12	2	6	0.43	0.63	7.5	2.4
13	1	5	0.38	0.58	8.2	3.1
14	2	7	0.55	0.85	8.8	2.8
15	2	6	0.40	0.60	7.6	3.1
16	1	9	0.45	0.65	7.9	3.3
17	3	9	0.50	0.80	8.4	2.8
18	3	8	0.60	0.85	8.5	3
19	2	6	0.55	0.70	7.6	3.1
20	7	11	0.60	0.85	8.7	3.3

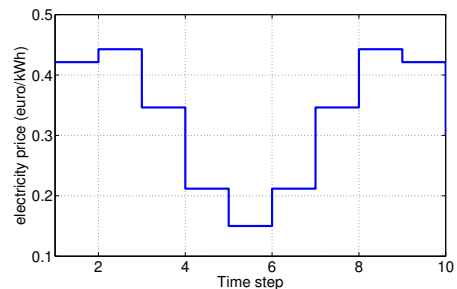


Figure 6: Profile of the electricity price for Case 1

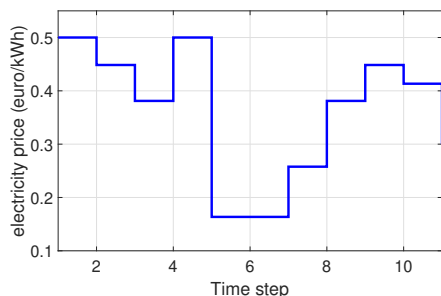


Figure 7: Profile of the electricity price for Case 2

Table 4: Simulation results for Case 1

	CPLEX	proposed multi-agent control method
J_{opt}	2.3907	2.3907

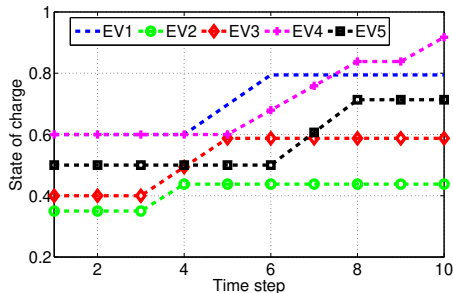


Figure 8: Charging dynamics of the 5 electric vehicles with the proposed multi-agent charging control method in Case 1

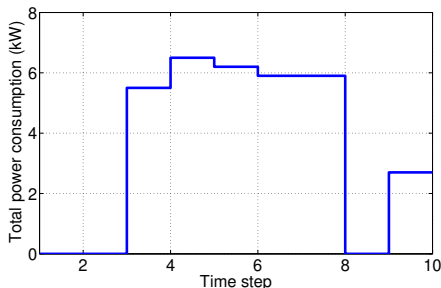


Figure 9: Total power consumption the 5 electric vehicles with the proposed multi-agent charging control method in Case 1

and XPRESS, which yield the globally optimal solution, provided the overall problem would be fully known by a single control agent. However, in the control setting considered in this paper, local control agents only share limited information with the coordinator. Therefore, neither the coordinator nor any of the local control agents have full information of the overall problem. In Table 4, the solution found by using CPLEX for this case is included to validate the solution found by the proposed multi-agent control method. In fact, from Table 4, it is clearly seen that the proposed multi-agent control method finds the globally optimal solution for Case 1. Besides, the computation time of the proposed method is 68.76 seconds. Finally, the charging dynamics and the total power consumption of the 5 electric vehicles with the proposed multi-agent charging control in this case are shown in Figures 8 and 9, respectively. It is clearly seen that the electric vehicles are charged up to the required level without exceeding the maximum power limit provided by the grid, and that they are charged as much as possible when the price of electricity is low.

In the simulations for Case 2, the proposed method without limiting the computation time or the number of information exchanges between the coordinator and the local control agents takes too much computation time. Therefore, we decided to use four alternatives that generate feasible solutions in a limited time:

- Alternative 1 (A1): $M_{\max} = 300000$; $t_{\max} = \infty$; depth-first search

Table 5: Simulation results for Case 2

	CPLEX	A1	A2	A3	A4
J_{opt}	9.1946	75.9733	9.6983	75.9733	9.6983

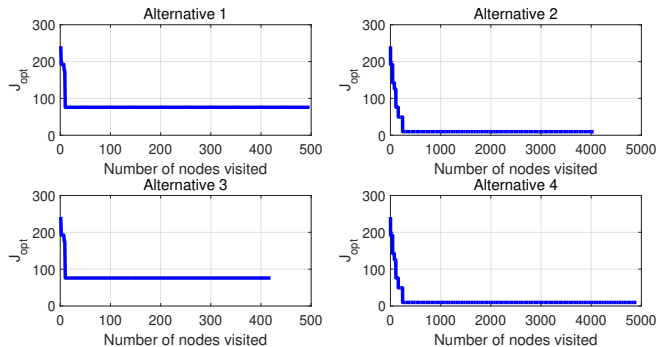


Figure 10: Evolution of J_{opt} as a function of the number of nodes visited in the search tree for the alternatives for Case 2

- Alternative 2 (A2): $M_{\max} = 300000$; $t_{\max} = \infty$; breadth-first search
- Alternative 3 (A3): $t_{\max} = 60000$ s; $M_{\max} = \infty$; depth-first search
- Alternative 4 (A4): $t_{\max} = 60000$ s; $M_{\max} = \infty$; breadth-first search

where t_{\max} denotes the maximum computation time (in seconds) and M_{\max} denotes the maximum number of information exchanges between the coordinator and local control agents. Note that in the depth-first search approach, the search tree is explored as far as possible along each branch before backtracking, while in breadth-first search, the search tree is explored as widely as possible on each level of nodes before moving to the next level.

The simulation results of using all the four variants of the proposed method are summarized in Table 5. Note that the result found by using CPLEX for this case is also included in Table 5 to help indicate how far the solutions found by the four alternatives are from the globally optimal one. Besides, Figure 10 shows the evolution of the values of the overall objective function as a function of the number of nodes that have been visited in the search tree for each of the alternatives. It can be seen from Table 5 that although the solutions found by the variants of the proposed multi-agent control method are not the same as the globally optimal solution, the best values of the overall objective function found by the second and the fourth alternatives are close to the globally optimal one. Moreover, given the same computation and communication budget, in the proposed multi-agent control method, breadth-first search generates better results than depth-first search.

Next, in order to show the effectiveness of the proposed multi-agent control method in balancing the quality of solution and the computation time, we conducted

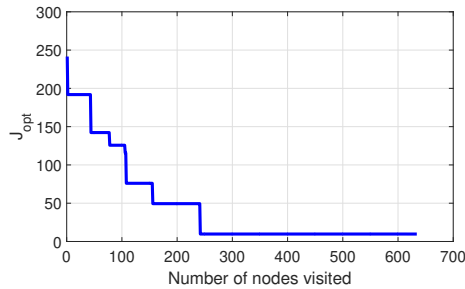


Figure 11: Evolution of J_{opt} as a function of the number of nodes visited in the search tree for Case 2 using breadth-first search with $t_{\text{max}} = 1200\text{s}$ and $M_{\text{max}} = \infty$

another simulation for Case 2 using breadth-first search, $M_{\text{max}} = \infty$ and a lower value for t_{max} of $t_{\text{max}} = 1200$ s (i.e. 20 minutes). The evolution of the values of the overall objective function in this simulation is shown in Figure 11. More precisely, the best value of the overall objective function found using breadth-first search in the given 20 minutes of computation time is 9.7270 and the corresponding charging dynamics and total power consumption of the 20 electric vehicles are shown in Figures 12 and 13, respectively. Note that in order to assess how far the behavior of the proposed controller is from optimality, we have compared the total power consumption of the electric vehicles with the proposed control method and with centralized optimal control using CPLEX in Figure 13. It is seen that the proposed multi-agent controllers show a behavior that is similar to the one of the centralized optimal controller with only slight differences. Besides, the best value of the overall objective function found using breadth-first search in 20 minutes of computation time is close to the globally optimal one and the electric vehicles are all charged up to the required levels without exceeding the maximum power limit imposed by the grid. Therefore, although each control agent only communicates very limited information (i.e. only the Lagrange multiplier associated with the charging power constraint) to the coordinator and only limited computation time is available, the proposed multi-agent control method can still find effectively balance the solution quality and the computation time.

Finally, as a first step to investigate the robustness of the proposed multi-agent control method, we consider that the power supply provided by the grid is disturbed and perform a closed-loop receding horizon control simulation for Case 2. More specifically, we consider the power supply predicted by the controllers is constantly $P_{k,\text{max}}^{\text{predicted}} = 36$ kW while the actual power supply at each step k is given by

$$P_{k,\text{max}}^{\text{actual}} = 36 + \omega \quad [\text{kW}]$$

where ω is a normally distributed pseudorandom number with mean 0 and standard deviation 5. The resulting power consumption of all electric vehicles with the proposed multi-agent charging control method, the predicted

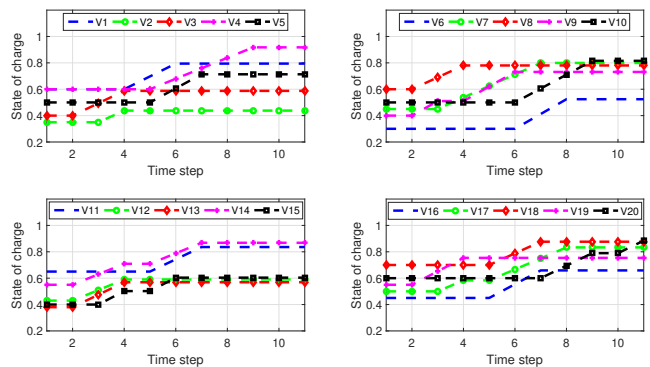


Figure 12: Charging dynamics of the 20 electric vehicles with the proposed multi-agent charging control method in Case 2 given 20 minutes of computation time

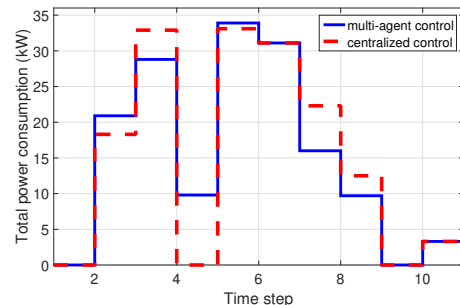


Figure 13: Total power consumption the 20 electric vehicles with the proposed multi-agent control method given 20 minutes of computation time and with centralized optimal control for Case 2

power supply, and the actual power supply are plotted in Figure 14. In order to highlight the effect of the disturbance on the actual power consumption of the vehicles, we also plot the power consumption of vehicles (indicated by red dashed line) in the undisturbed case. It is seen that when the actual power supply is higher than the level that is needed for the control action to be implemented, the vehicles are charged with the control action of the proposed method fully implemented. When the actual power supply is lower than the needed level, the control action of the proposed method is not fully implemented in order to respect the actual charging constraints. In that case, the control action of the proposed method is implemented in a way that the vehicles with higher charging-emergency-rate have higher priority to change. Note that in this simulation, the charging-emergency-rate $q_{i,k}$ of a vehicle i at a simulation step k is defined by

$$q_{i,k} = \frac{\text{remaining state of charge}}{\text{remaining charging time}} = \frac{s_i^{\text{req}} - s_{i,k}}{k_{i,\text{departure}} - k}$$

In this way, the actual power consumption is always lower than the actual power supply provided by the grid. Actually, the vehicles that are not charged due to insufficient power supply at certain time steps are charged at later time steps under the control of the proposed method.

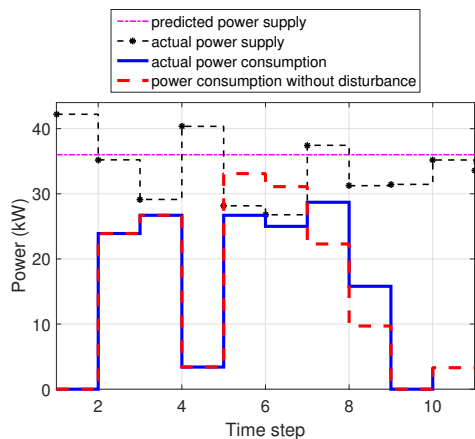


Figure 14: Power consumption of the 20 electric vehicles with the proposed multi-agent control method given 20 minutes of computation time in Case 2 when the power supply is disturbed

Besides, the value of the cost function for the disturbed case is 9.6948, which corresponds to a small performance drop of about 3.3% compared with the value 9.3865 of the cost function for the undisturbed case. Therefore, the proposed multi-agent control method is still working adequately when the power supply is disturbed.

6. Conclusions and future work

We have considered multi-agent model predictive control for a class of hybrid systems governed by discrete inputs and subject to global hard constraints where each subsystem has a local convex objective function and a strictly increasing constraint function. We focused on the scenario where each subsystem only shares limited information with the external environment, and we developed a novel multi-agent model predictive control method by integrating a distributed resource allocation coordination algorithm into a solution space branching mechanism. With the distributed resource allocation algorithm, the global feasibility of the local control decisions is always guaranteed. With the solution space branching mechanism, the search tree for the overall solution space is built smartly based on the outcome of the distributed resource allocation coordination algorithm. Results for the charging control of a fleet of electric vehicles in a simulation study show that the proposed multi-agent control method effectively balances the solution quality, the computation time, and the communication burden.

In our future work, we will focus on developing a distributed algorithm to compute the global lower bound of the overall objective function, and then combine that algorithm with the search tree building mechanism proposed in this paper. We will also perform more complex and more challenging case studies considering coordinating both charging-to-vehicle (G2V) and discharging-to-grid (V2G) of electric vehicles. In addition, we will

perform an extensive assessment including the stability and the robustness of the developed multi-agent control method.

Acknowledgments

This research was supported by Ammodo via the Van Gogh project VGP.14/47 and by the China Scholarship Council under Grant 201207090001.

Appendix A. Calculation of the Lagrange multiplier given the primal solution

If a problem is given by

$$\begin{aligned} \min_{u_i \in U_i} J_i(u_i) \\ \text{subject to } G_i(x_i) \leq \theta_i \end{aligned}$$

with $G_i(\cdot)$ a scalar function, then by introducing a Lagrange multiplier λ_i associated with the constraint $G_i(u_i) \leq \theta_i$, its dual problem is given by

$$\max_{\lambda_i \geq 0} \min_{u_i \in U_i} J_i(u_i) + \lambda_i (G_i(u_i) - \theta_i)$$

Letting u_i^* and λ_i^* be the solution to the dual problem and $d_{i,\mathbf{J}}$ and $d_{i,\mathbf{G}}$ be respectively the derivatives of $J_i(\cdot)$ and $G_i(\cdot)$ at u_i^* , then λ_i^* , $d_{i,\mathbf{J}}$ and $d_{i,\mathbf{G}}$ have to satisfy

$$d_{i,\mathbf{J}} + \lambda_i^* \cdot d_{i,\mathbf{G}} = 0$$

or equivalently

$$d_{i,\mathbf{J}}^T + \lambda_i^* \cdot d_{i,\mathbf{G}}^T = 0 \quad (\text{A.1})$$

Solving equation (A.1) and considering dual feasibility of λ_i^* yields

$$\lambda_i^* = \begin{cases} -\frac{d_{i,\mathbf{J}}^T \cdot d_{i,\mathbf{G}}}{d_{i,\mathbf{G}}^T \cdot d_{i,\mathbf{G}}}, & \text{if } -\frac{d_{i,\mathbf{J}}^T \cdot d_{i,\mathbf{G}}}{d_{i,\mathbf{G}}^T \cdot d_{i,\mathbf{G}}} > 0 \\ 0, & \text{otherwise} \end{cases}$$

Note that the dual variable λ_i is dual feasible if $\lambda_i \geq 0$ (Palomar and Chiang, 2006).

References

- Bertsekas, D.P., 1999. Nonlinear Programming. Athena Scientific .
- Bourdais, R., Guéguen, H., Belmiloudi, A., 2012. Distributed model predictive control for a class of hybrid system based on Lagrangian relaxation, in: Proceedings of the 4th Conference on Analysis and Design of Hybrid Systems, Eindhoven, The Netherlands. pp. 46–51.
- Boyd, S., Vandenberghe, L., 2004. Convex Optimization. Cambridge university press.
- Cai, X., Xie, L., Lu, P., Chen, J., 2014. Optimal selection of the decomposition structure based on ga for distributed model predictive control systems, in: Proceedings of 11th World Congress on Intelligent Control and Automation, Shenyang, China. pp. 4560–4565.
- Camponogara, E., Jia, D., Krogh, B., Talukdar, S., 2002. Distributed model predictive control. IEEE Control Systems Magazine 22, 44–52.

- Cao, Y., Yu, W., Ren, W., Chen, G., 2013. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial Informatics* 9, 427–438.
- Christofides, P.D., Scattolini, R., de la Peña, D.M., Liu, J., 2013. Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering* 51, 21–41.
- Clement-Nyns, K., Haesen, E., Driesen, J., 2010. The impact of charging plug-in hybrid electric vehicles on a residential distribution grid. *IEEE Transactions on Power Systems* 25, 371–380.
- Cohen, G., 1978. Optimization by decomposition and coordination: a unified approach. *IEEE Transactions on Automatic Control* 23, 222–232.
- Dunbar, W.B., Murray, R.M., 2006. Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica* 42, 549–558.
- Dutta, P.S., Jennings, N.R., Moreau, L., 2005. Cooperative information sharing to improve distributed learning in multi-agent systems. *Journal of Artificial Intelligence Research* 24, 407–463.
- Fernandez, L.P., Roman, T.G.S., Cossent, R., Domingo, C.M., Frias, P., 2011. Assessment of the impact of plug-in electric vehicles on distribution networks. *IEEE Transactions on Power Systems* 26, 206–213.
- Frick, D., Domahidi, A., Morari, M., 2015. Embedded optimization for mixed logical dynamical systems. *Computers & Chemical Engineering* 72, 21–33.
- Hajimiragha, A., Canizares, C.A., Fowler, M.W., Elkamel, A., 2010. Optimal transition to plug-in hybrid electric vehicles in ontario, canada, considering the electricity-grid limitations. *IEEE Transactions on Industrial Electronics* 57, 690–701.
- Han, S., Han, S., Sezaki, K., 2010. Development of an optimal vehicle-to-grid aggregator for frequency regulation. *IEEE Transactions on Smart Grid* 1, 65–72.
- Hu, J., Saleem, A., You, S., Nordström, L., Lind, M., Østergaard, J., 2015. A multi-agent system for distribution grid congestion management with electric vehicles. *Engineering Applications of Artificial Intelligence* 38, 45–58.
- Huang, C.J., Guan, C.T., Chen, H.M., Wang, Y.W., Chang, S.C., Li, C.Y., Weng, C.H., 2013. An adaptive resource management scheme in cloud computing. *Engineering Applications of Artificial Intelligence* 26, 382–389.
- Kantamneni, A., Brown, L.E., Parker, G., Weaver, W.W., 2015. Survey of multi-agent systems for microgrid control. *Engineering Applications of Artificial Intelligence* 45, 192–203.
- Lawler, E.L., Wood, D.E., 1966. Branch-and-bound methods: A survey. *Operations Research* 14, 699–719.
- Leitão, P., 2009. Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence* 22, 979–991.
- Luo, R., Bourdais, R., van den Boom, T., De Schutter, B., 2015a. Integration of resource allocation coordination and branch-and-bound, in: *Proceedings of the 54th IEEE Conference on Decision and Control*, Osaka, Japan. pp. 4272–4277.
- Luo, R., Bourdais, R., van den Boom, T., De Schutter, B., 2015b. Properties of applying a resource allocation coordination algorithm to optimization problems with discrete decision variables. Technical Report 15-025. Delft Center for Systems and Control, Delft University of Technology. Delft, The Netherlands.
- Mal, S., Chattopadhyay, A., Yang, A., Gadh, R., 2013. Electric vehicle smart charging and vehicle-to-grid operation. *International Journal of Parallel, Emergent and Distributed Systems* 28, 249–265.
- Marra, F., Yang, G., Traholt, C., Larsen, E., Rasmussen, C., You, S., 2012. Demand profile study of battery electric vehicle under different charging options, in: *Proceedings of the 2012 IEEE Power and Energy Society General Meeting*, San Diego, USA. pp. 1–7.
- Mayne, D., Rawlings, J., Rao, C., Scokaert, P., 2000. Constrained model predictive control: Stability and optimality. *Automatica* 36, 789–814.
- Negenborn, R.R., De Schutter, B., Hellendoorn, J., 2008. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. *Engineering Applications of Artificial Intelligence* 21, 353–366.
- Nguyen, H., Zhang, C., Mahmud, M., 2015. Optimal coordination of g2v and v2g to support power grids with high penetration of renewable energy. *IEEE Transactions on Transportation Electrification* 1, 188–195.
- Olaru, S., Thomas, J., Dumur, D., Buisson, J., 2004. Genetic algorithm based model predictive control for hybrid systems under a modified MLD form. *International Journal of Hybrid Systems* 4, 113–132.
- Palomar, D.P., Chiang, M., 2006. A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communications* 24, 1439–1451.
- Saber, A.Y., Venayagamoorthy, G.K., 2009. Optimization of vehicle-to-grid scheduling in constrained parking lots, in: *Proceedings of 2009 IEEE Power & Energy Society General Meeting*, Calgary, Canada. pp. 1–8.
- Saber, A.Y., Venayagamoorthy, G.K., 2011. Plug-in vehicles and renewable energy sources for cost and emission reductions. *IEEE Transactions on Industrial Electronics* 58, 1229–1238.
- Scattolini, R., 2009. Architectures for distributed and hierarchical model predictive control—A review. *Journal of Process Control* 19, 723–731.
- Shor, N.Z., 2012. *Minimization methods for non-differentiable functions*. Springer Science & Business Media .
- Vujanic, R., Esfahani, P.M., Goulart, P., Mariethoz, S., Morari, M., 2016. A decomposition method for large scale MILPs, with performance guarantees and a power system application. *Automatica* 67, 144–156.