Optimization and Engineering
https://doi.org/10.1007/s11081-020-09506-x

RESEARCH ARTICLE

Hierarchical decompositions for MPC of resource constrained control systems: applications to building energy management

Eduardo Camponogara[1] · Helton Scherer[2] · Lorenz Biegler[3] · Ignacio Grossmann[3]

Received: 29 December 2019 / Revised: 19 April 2020 / Accepted: 19 April 2020

Abstract

Energy management can play a significant role in energy savings and temperature control of buildings, which consume a major share of energy resources worldwide. Model predictive control (MPC) has become a popular technique for energy management, arguably for its ability to cope with complex dynamics and system constraints. The MPC algorithms found in the literature are mostly centralized, with a single controller collecting signals and performing the computations. However, buildings are dynamic systems obtained by the interconnection of subsystems, with a distributed structure which is not necessarily explored by standard MPC. To this end, this work proposes hierarchical decompositions to split the computations between a master problem (*centralized component*) and a set of decoupled subproblems (*distributed components*) which brings about organizational flexibility and distributed computation. Three general methods are considered for hierarchical control and optimization, namely bilevel optimization, Benders and Lagrangean decomposition. Results are reported from a numerical analysis of the decompositions and a simulated application to the energy management of a building, in which a limited source of chilled water is distributed among HVAC units.

Keywords Bilevel optimization · Benders decomposition · Lagrangean decomposition · Predictive control · Linear systems · HVAC

1 Introduction

Studies show that energy consumption in buildings accounts for roughly 40% of the worldwide energy demand, and more than half can be attributed to Heating, Ventilation, and Air Conditioning (HVAC) systems (D&R International Ltd 2009;

✉ Eduardo Camponogara
eduardo.camponogara@ufsc.br

Extended author information available on the last page of the article

Published online: 19 May 2020

Pérez-Lombard et al. 2008). Such a share of the energy market is driving investments towards energy efficient buildings, including materials with better insulation, use of renewable energy, and control and optimization technology to manage the energy resources (Escrivá-Escrivá et al. 2010). For the latter, Model Predictive Control (MPC) has become a popular technique, arguably for its ability to handle complex dynamics and constraints, but also for optimizing objective functions that account for thermal comfort and energy savings.

Álvarez et al. (2013) presented a predictive controller to manage the energy consumption rate in a building while maximizing user comfort. Their predictive controller promoted thermal comfort by operating HVAC units smartly, achieved by optimizing a suitable cost function with a Lagrangean dual method. Moroşan et al. (2010) designed a distributed MPC algorithm to reduce energy consumption of multizone buildings, but without compromising the thermal comfort of users. Despite halting iterations before convergence to keep computational cost low, the distributed algorithm achieved satisfactory performance. Scherer et al. (2013, 2015) developed a distributed interior-point algorithm for MPC to manage a limited energy source, while regulating temperature in a building to optimize user comfort. The distributed algorithm takes advantage of the energy resource constraint to allow a high degree of concurrency in the computations. Castilla et al. (2014) brought about a two-layer control system for HVAC units, whereby the bottom layer consists of PID controllers and the upper layer is a non-linear MPC. The control strategy maintains thermal comfort conditions in a bioclimatic building, considering the effects on users productivity and energy consumption.

More recently, renewed interest in the dynamic control of buildings has emerged. Maasoumy et al. (2014) brought forth methodologies to handle model uncertainty in MPC for optimizing building energy consumption. The MPC methodologies were compared against a nominal controller, indicating that performance gains can be achieved by the robust controllers for a significant degree of uncertainty on state estimation and model parameters. Salakij et al. (2016) presented a prediction model for the heat and moisture transfer in buildings, which was validated and then combined with a model-based predictive controller. A simulation analysis showed that the MPC control of HVAC units can achieve better performance than traditional control methods, in terms of energy consumption and building comfort conditions for users. Yu et al. (2017) validated the MPC approach of Salakij et al. (2016) by physical experiments in a climate-controlled chamber, showing that MPC can be effective and reduce the number of sensors required.

The literature on building energy management is mostly on centralized MPC, with a single controller collecting measurements and handling the computations. However, modern buildings are dynamic systems obtained by interconnecting distributed dynamic systems, each representing a room and the associated thermal control equipment. Moroşan et al. (2010) and Scherer et al. (2015) proposed distributed MPC to take advantage of this distributed structure. Nevertheless, a fully distributed approach is somewhat complex to implement due to the need of strong coordination.

In this paper, we propose hierarchical decompositions to split the computations between a master problem (*centralized component*) and a set of subproblems (*distributed components*), which for being fully or highly decoupled, could be solved with multi-core and parallel hardware. Despite this potential of parallel computation, hierarchical and distributed algorithms are often slower than their centralized counter-parts. Thus, the main benefit of a hierarchical decomposition is not computational, but rather organizational as it facilitates the expansion and reconfiguration of the control system, a feature that stems from the simple coordination scheme and reduced information communication. Herein, hierarchical decompositions are developed for MPC of dynamically decoupled systems that share resources, such as electric power or chilled/hot water, being a suitable framework for energy building management. Three general methods are considered for hierarchical control and optimization, namely bilevel optimization, Benders and Lagrangean decomposition.

1.1 Contribution

To better relate this work with models and algorithms from the literature, we state below the contributions of this paper:

1. A distributed reformulation of resource constraints that enables the decomposition of the augmented Lagrangean (of the resource-constrained MPC problem) into a set of loosely coupled subproblems for distributed computation.
2. A Lagrangean decomposition for solving the MPC problem with convergence guarantee, assuming problem convexity, and the use of the nonlinear Gauss–Seidel algorithm to solve the dual function in a decoupled manner.
3. A numerical analysis and qualitative comparison of three hierarchical decompositions, namely Bilevel, Benders, and Lagrangean.
4. A simulated study of MPC with hierachical decomposition for energy management and temperature control in a building.

1.2 Organization

The paper is organized as follows. Section 2 presents the MPC problem for resource constrained dynamic systems, along with reformulations that are suitable for the decompositions. Section 3 discusses the implementation and reports numerical results of the decompositions for the control of dynamic systems. Section 4 presents simulation results of an application to the energy management of a building, in which a limited source of chilled water is distributed to HVAC units.

2 MPC formulation

We start by stating an MPC problem for the control of independent dynamic subsystems that share limited resources—the baseline problem. Compact reformulations are proposed for the solution with bilevel optimization, Benders and Lagrangean decomposition. For the latter, we present a distributed representation of the resource constraints.

2.1 Baseline problem

MPC uses a process model to convert a dynamic control problem into a series of static optimization problems, which are solved over time overlapping prediction horizons (Camacho and Bordons 2004). At the current time k, feedback is obtained by measuring the system state $x(k)$ and then an optimization problem is solved over the prediction horizon, but only the control signal $u(k)$ for the current time is applied to the process. At the next sample time, $k+1$, the system state $x(k+1)$ is sampled and the process repeated.

It is worth mentioning that an incremental model in the state space was used for the MPC formulation. A regular state-space formulation in MPC strategy can lead to steady-state errors due to modeling errors or plant model mismatches. With an incremental formulation, the MPC problem achieves offset-free tracking of constant references for systems without a zero at the origin (Ruscio 2013; Camacho and Bordons 2004). From a practical point of view, it is convenient to think about reducing the control effort to avoid excessive operations of the manipulators, and consequently, equipment wear. This formulation also brings the control increment decision variable, Δu_m, given by the relation $\Delta u_m(k) = u_m(k) - u_m(k-1)$, and a penalty for control signal variation in the cost function.

Let $\mathcal{M} = \{1, \dots, M\}$ be the set of subsystems, $\mathcal{R} = \{1, \dots, R\}$ be the set of resources, N_u define the control horizon, and N_x establish the prediction horizon for the outputs. Adjustable prediction and control horizons offer flexibility for control design. For instance, the control horizon can be shorter than the prediction horizon to simplify computations, and also because the controls are updated every time the horizon is rolled forward.

The MPC problem is given by:

$$\begin{aligned} P : \min J = & \sum_{m=1}^{M} \sum_{j=1}^{N_x} \|y_m(k+j|k) - w_m(k+j)\|^2_{Q_m} \\ & + \sum_{m=1}^{M} \sum_{j=0}^{N_u-1} \|\Delta u_m(k+j|k)\|^2_{W_m} \end{aligned} \tag{1a}$$

while, for each subsystem $m \in \mathcal{M}$, being subject to:

$$x_m(k+j|k) = A_m^j x_m(k) + \sum_{l=1}^{j} A_m^{j-l} B_m \Delta u_m(k+l-1|k),\ j = 1, \dots, N_x \tag{1b}$$

$$y_m(k+j|k) = C_m x_m(k+j|k),\ j = 1, \dots, N_x \tag{1c}$$

$$u_m(k+j|k) = \begin{cases} u_m(k+j-1|k) + \Delta u_m(k+j|k), & j = 0, \dots, N_u - 1 \\ u_m(k+j-1|k), & j = N_u, \dots, N_x - 1 \end{cases} \tag{1d}$$

$$u_m(k-1|k) = u_m(k-1) \tag{1e}$$

$$y_m^{\min} \leq y_m(k+j|k) \leq y_m^{\max},\ j = 1, \dots, N_x \tag{1f}$$

$$u_m^{\min} \leq u_m(k+j|k) \leq u_m^{\max},\ j = 0, \dots, N_u - 1 \tag{1g}$$

$$\Delta u_m^{\min} \leq \Delta u_m(k+j|k) \leq \Delta u_m^{\max},\ j = 0, \dots, N_u - 1 \tag{1h}$$

and with limited availability of resource $r = 1, \dots, R$, at time step $j = 0, \dots, N_x - 1$:

$$\sum_{m=1}^{M} s'_{r,m} u_m(k+j|k) \leq s_r^{\max}(k+j) \tag{1i}$$

in which, for each subsystem $m \in \mathcal{M}$:

- $x_m(k)$ is the system state at time k, and $x_m(k+j|k)$ is the state prediction for time $k+j$ calculated with the information available until time k.
- $y_m(k+j|k)$ is the predicted output for time $k+j$.
- $w_m(k+j)$ defines the desired output trajectory.
- $u_m(k+j|k)$ is the future control input and $\Delta u_m(k+j|k)$ is the future control increment.
- $Q_m = Q'_m$ and $W_m = W'_m$ are positive definite matrices that penalize the errors on trajectory tracking and control variation, respectively.
- $x_m(k)$ and $u_m(k-1)$ are known values with the initial conditions.
- A_m, B_m, and C_m are system matrices of conformable dimensions.
- $\|x\|_Q = \sqrt{x'Qx}$ is the vector norm induced by a positive definite matrix Q.
- $y_m^{\min}$, $y_m^{\max}$, $u_m^{\min}$, $u_m^{\max}$, $\Delta u_m^{\min}$, and $\Delta u_m^{\max}$ impose bounds on outputs, control signals, and control increments, respectively.
- $s_r^{\max}(k)$ is the resource r available at time k, and $s_{r,m}$ defines the rate of consumption by subsystem m.

Notice that the MPC problem is of cooperative nature (Scattolini 2009), meaning that it seeks a Pareto solution induced by a weighted sum of the subsystem objectives, with relative subsystem costs given by the matrices Q_m and W_m.

Without the resource constraints (1i), the subsystems would be fully decoupled and could be controlled independently. A strategy for distributed optimization consists in obtaining an approximation problem to render the subsystems decoupled, which can then be optimized with strategies such as coordinate descent. Approximations can be obtained with the augmented Lagrangean and the barrier function: the first leads to a dual method whose outer loop updates the Lagrange multipliers and penalty factor (Bertsekas 1995), whereas the second is a primal method with an outer loop that updates the centralization parameter (Boyd and Vandenberghe 2004; Camponogara and Scherer 2011). Such methods decouple the constraints; however, the subsystems become coupled in the objective through the quadratic penalty of the augmented Lagrangean or the barrier function.

Our work investigates other means to decompose the MPC problem into a family of subproblems and to enable a high degree of decoupling. We consider bilevel optimization (Colson et al. 2007), Benders (Benders 1962) and Lagrangean decomposition (Guignard and Kim 1987; Terrazas-Moreno et al. 2011). Hereafter the condition "$|k$" will be omitted for the sake of simplicity.

2.2 Compact formulation

Problem P is recast in a compact, more convenient form as follows:

$$\min_{z} \; f = \sum_{m=1}^{M} f_m(z_m) \tag{2a}$$

$$\text{s.t. for } m = 1, \ldots, M \; : \tag{2b}$$

$$h_m(z_m) = 0 \tag{2c}$$

$$g_m(z_m) \leq 0 \tag{2d}$$

$$g(z) \leq s \tag{2e}$$

in which $z_m = (x_m(k+j), y_m(k+j), u_m(k+j), \Delta u_m(k+j) \; : \; \forall j)$ is a vector with all the variables of subsystem m, $s = (s_r^{\max}(k+j) \; : \; \forall r, j)$ is a vector with the resources available over time, and $z = (z_m \; : \; m \in \mathcal{M})$ collects all the subsystem variables. Further, the vector functions h_m, g_m, and g represent respectively the equalities (1b)–(1e), inequalities (1f)–(1h), and the resource inequalities (1i).

2.3 Bilevel formulation

Let $ss = (ss_{r,m}(k+j) \; : \; \forall r, j, m)$ be a vector of resource allocations, where $ss_{r,m}(k+j)$ is the resource r allocated to subsystem m, at time $k+j$. Then, P can be framed as a bilevel optimization problem (Colson et al. 2007; Chen et al. 2014), with the master problem given by

$$U \; : \; \min_{ss} \; f(ss) = \sum_{m=1}^{M} f_m(z_m(ss_m)) \tag{3a}$$

$$\text{s.t. for } r = 1, \ldots, R, \, j = 0, \ldots, N_x - 1 \; :$$
$$\sum_{m=1}^{M} ss_{r,m}(k+j) \leq s_r^{\max}(k+j) \tag{3b}$$

$$ss_{r,m}(k+j) \geq 0, \; \forall m \in \mathcal{M} \tag{3c}$$

in which $ss_m = (ss_{r,m}(k+j) : \forall r, j)$ are the resources allocated to subsystem m, such that $ss = (ss_1, \dots, ss_M)$. The lower-level problem L consists of solving a set of subproblems $\{L_m\}$, one for each $m \in \mathcal{M}$, being defined by

$$L_m(ss_m) : \ z_m(ss_m) = \arg\min_{z_m} \ f_m(z_m) \tag{4a}$$

$$\text{s.t.} : \ h_m(z_m) = 0 \tag{4b}$$

$$g_m(z_m) \leq 0 \tag{4c}$$

$$\begin{gathered}\text{For all } r = 1, \dots, R, j = 0, \dots, N_x - 1 : \\ s'_{r,m} u_m(k+j|k) \leq ss_{r,m}(k+j).\end{gathered} \tag{4d}$$

Remark 1 The objective function $f(ss)$ of the upper-level problem (3) is nonincreasing.

If the resource allocation increases from ss to $\widehat{ss} = ss + \Delta ss$, $\Delta ss \geq 0$, then the feasible set of the constraint (4d) may expand, while the other constraints of (4) are unaffected. Thus, f_m cannot increase and so does not $f(ss)$ increase.

2.4 Benders decomposition

Here, we begin by introducing the subproblems rather than the master problem for the Generalized Benders decomposition (Geoffrion 1972). For a feasible resource allocation ss, the *optimality* subproblem is given by

$$BO(ss) : \ bo(ss) = \min_{z} \ \sum_{m=1}^{M} f_m(z_m) \tag{5a}$$

$$\begin{gathered}\text{s.t.} : \ \text{for } m = 1, \dots, M : \\ h_m(z_m) = 0\end{gathered} \tag{5b}$$

$$g_m(z_m) \leq 0 \tag{5c}$$

$$R_m z_m - S_m ss_m \leq 0 \tag{5d}$$

in which R_m and S_m are suitable matrices[1] that define the constraints (5d). Notice that $bo(ss)$ induces an upper bound for a feasible resource allocation ss, meaning

[1] In the particular problem of concern, S_m is the identity matrix and $R_m z_m$ is effectively $\widetilde{R}_m u_m$ for a suitable matrix $\widetilde{R}_m$, since only the terms $s'_{r,m} u_m(k+j)$ are needed.

a vector ss that satisfies constraints (3b)–(3c) and also renders $BO(ss)$ feasible. Clearly, $bo(ss)$ can be computed in parallel as follows,

$$bo(ss) = \sum_{m=1}^{M} bo_m(ss_m) \tag{6}$$

for which

$$BO_m(ss_m) \,:\; bo_m(ss_m) = \min_{z_m} \; f_m(z_m) \tag{7a}$$

$$\text{s.t.} \,:\; h_m(z_m) = 0 \tag{7b}$$

$$g_m(z_m) \leq 0 \tag{7c}$$

$$R_m z_m - S_m ss_m \leq 0 \tag{7d}$$

At iteration p of the Benders algorithm, let $ss^{(p)}$ be the resource allocation vector and assume that $BO(ss^{(p)})$ is feasible. Let $\mu_m^{(p)}$ be the Lagrange multipliers associated with the resource constraint (7d), at the optimal solution $z_m^{(p)}$. Then, the following optimality cut can be added to the Benders master problem:

$$\alpha_{\mathrm{B}} \geq \sum_{m \in \mathcal{M}} f_m(z_m^{(p)}) + \sum_{m \in \mathcal{M}} \mu_m^{(p)\prime} \left[R_m z_m^{(p)} - S_m ss_m \right] \tag{8}$$

with α_{B} being the lower bound for the overall objective of P. Due to the complementary conditions, the local constraints (7b) and (7c) do not play a part in the Benders cut. The decision space of the Benders master problem consists of the allocation vector ss and the lower bound α_{B}.

For an infeasible resource allocation, the *feasibility* subproblem is solved:

$$BF(ss) \,:\; bf(ss) = \min_{\gamma \geq 0, z} \; \gamma \tag{9a}$$

$$\begin{aligned} \text{s.t.} \,:\; & \text{for } m = 1, \ldots, M \,: \\ & h_m(z_m) \leq \gamma \cdot e_m \end{aligned} \tag{9b}$$

$$-h_m(z_m) \leq \gamma \cdot e_m \tag{9c}$$

$$g_m(z_m) \leq \gamma \cdot e_m \tag{9d}$$

$$R_m z_m - S_m ss_m \leq \gamma \cdot e_m \tag{9e}$$

with $e_m = (1, 1, \ldots, 1)$ being a vector of suitable dimension and γ a nonnegative scalar. The optimal γ is obtained by solving an auxiliary subproblem only for the infeasible BO_m's. Let $\mathcal{M}_{\text{infeas}} = \{m \in \mathcal{M} \,:\, BO_m(ss_m) \text{ is infeasible}\}$. Then, $BF(ss)$ is solved as follows:

$$bf(ss) = \max\{bf_m(ss_m) \,:\, m \in \mathcal{M}_{\text{infeas}}\} \tag{10}$$

with the following subproblem solved for all $m \in \mathcal{M}_{\text{infeas}}$:

$$BF_m(ss_m) \,:\, bf_m(ss_m) = \min_{\gamma_m \geq 0, z_m} \;\gamma_m \tag{11a}$$

$$\text{s.t.} \,:\, h_m(z_m) \leq \gamma_m \cdot e_m \tag{11b}$$

$$- h_m(z_m) \leq \gamma_m \cdot e_m \tag{11c}$$

$$g_m(z_m) \leq \gamma_m \cdot e_m \tag{11d}$$

$$R_m z_m - S_m ss_m \leq \gamma_m \cdot e_m \tag{11e}$$

Notice that if ss_m is feasible for BO_m, then the corresponding BF_m will have an optimal value $bf(ss_m) = 0$ and the feasibility subproblem is implicitly solved. Assume that $ss^{(p)}$ is an infeasible allocation at iteration p. Let $\mathcal{M}^{(p)} \subset \mathcal{M}$ be the subset of subproblems such that $bf_m(ss_m^{(p)}) = bf(ss^{(p)})$. Then, the infeasibility cut is obtained as follows:

$$\begin{aligned}\sum_{m \in \mathcal{M}^{(p)}} \Big\{ & \mu_{m,b}^{(p)}{}' h_m(z_m^{(p)}) - \mu_{m,c}^{(p)}{}' h_m(z_m^{(p)}) \\ & + \mu_{m,d}^{(p)}{}' g_m(z_m^{(p)}) + \mu_{m,e}^{(p)}{}' \big[R_m z_m^{(p)} - S_m ss_m\big] \Big\} \leq 0\end{aligned} \tag{12}$$

where $z_m^{(p)}$ is the solution to $BF_m(ss_m^{(p)})$ and $\mu_{m,t}^{(p)}$ is the respective Lagrange multipliers.

At iteration p an optimality cut is obtained by solving $BO(ss^{(p)})$, or else a feasibility cut by solving $BF(ss^{(p)})$. Let $\mathcal{O}^{(p)}$ and $\mathcal{F}^{(p)}$ be the indices of the iterations for which an optimality and feasibility cut was respectively produced. Then, the Benders master problem at iteration p can be stated as follows:

$$BM(p) \,:\, \min_{ss \geq 0, \alpha_{\text{B}}} \;\alpha_{\text{B}} \tag{13a}$$

while being subject to:

$$R \cdot ss \leq s \tag{13b}$$

$$\alpha_{\text{B}} \geq \sum_{m \in \mathcal{M}} f_m(z_m^{(i)}) + \sum_{m \in \mathcal{M}} \mu_m^{(i)\prime} \big[R_m z_m^{(i)} - S_m ss_m\big], \; i \in \mathcal{O}^{(p)} \tag{13c}$$

$$\begin{aligned}\sum_{m \in \mathcal{M}^{(i)}} \Big\{ & \mu_{m,b}^{(i)}{}' h_m(z_m^{(i)}) - \mu_{m,c}^{(i)}{}' h_m(z_m^{(i)}) \\ & + \mu_{m,d}^{(i)}{}' g_m(z_m^{(i)}) + \mu_{m,e}^{(i)}{}' \big[R_m z_m^{(i)} - S_m ss_m\big] \Big\} \leq 0, \quad i \in \mathcal{F}^{(p)}\end{aligned} \tag{13d}$$

with R being a suitable matrix that expresses the resource constraint (3b). $BM(p)$ is clearly a linear program.

2.5 Distributed constraint representation

In general, the augmented Lagrangean has some advantages in relation to the classical Lagrangean; the latter may have a dual gap and require a procedure to recover primal feasibility. Also the augmented Lagrangean enables the computation of a sequence of Lagrangean multipliers converging to the optimum of convex differentiable problems. However, the induced coupling from the relaxed constraint prevents the solution of the relaxed problem in a decomposed way, which in the case at hand are the resource constraints.

Fortunately, the resource constraints (1i) can be represented in a distributed manner. Let $\widehat{s}_{r,1}(k) = s_r^{\max}(k)$ be the available amount of resource r at time k. Now consider the following family of constraints for $m = 1, \dots, M$:

$$s'_{r,m} u_m(k) + \widehat{s}_{r,m+1}(k) = \widehat{s}_{r,m}(k) \tag{14a}$$

$$\widehat{s}_{r,m+1}(k) \geq 0 \tag{14b}$$

in which $\widehat{s}_{r,m}(k)$ is the resource received by subsystem m, which is partly used by itself and partially transferred to the next subsystem in $\widehat{s}_{r,m+1}(k)$. By adding up the constraints (14a), we obtain,

$$\begin{aligned}
&\sum_{m=1}^{M} s'_{r,m} u_m(k) + \sum_{m=1}^{M} \widehat{s}_{r,m+1}(k) = \sum_{m=1}^{M} \widehat{s}_{r,m}(k) \\
&\iff \sum_{m=1}^{M} s'_{r,m} u_m(k) + \left[\sum_{m=2}^{M} \widehat{s}_{r,m}(k) + \widehat{s}_{r,M+1}(k) \right] = \widehat{s}_{r,1}(k) + \sum_{m=2}^{M} \widehat{s}_{r,m}(k) \\
&\iff \sum_{m=1}^{M} s'_{r,m} u_m(k) + \widehat{s}_{r,M+1}(k) = \widehat{s}_{r,1}(k) \\
&\iff \sum_{m=1}^{M} s'_{r,m} u_m(k) \leq s_r^{\max}(k)
\end{aligned} \tag{15}$$

as $\widehat{s}_{r,M+1}(k) \geq 0$, which shows that the distributed constraint family (14) is equivalent to the original resource constraint (1i).

2.6 Lagrangean decomposition formulation

The distributed structure of the constraint family (14) enables the application of a Lagrangean decomposition, which consists in duplicating the variables that couple the subsystems and introducing consistency constraints (Guignard and Kim 1987). These variables are also known as consensus variables, as their values must agree to ensure a consistent solution for the entire system, a condition that can be imposed by constraints.

Let $\widehat{s}_{r,m}(k)$ be modeled by two variables (Boyd et al. 2011), namely $\widehat{s}^{\text{in}}_{r,m}(k)$ to represent the resource available from the perspective of subsystem m, and $\widehat{s}^{\text{out}}_{r,m-1}(k)$ to represent the residual resource made available to subsystem m by subsystem $m-1$, if $m \geq 2$, whose values should agree for consistency. For subsystem $m = 1$, $\widehat{s}^{\text{in}}_{r,m}(k)$ is the total resource $s^{\max}_r(k)$ available. Then, a compact formulation of P is obtained,

$$\min_{\widehat{z}} \; f = \sum_{m=1}^{M} f_m(z_m) \tag{16a}$$

$$\begin{aligned} &\text{s.t. for } m = 1, \ldots, M : \\ &h_m(z_m) = 0 \end{aligned} \tag{16b}$$

$$g_m(z_m) \leq 0 \tag{16c}$$

$$\widehat{g}(\widehat{z}) \leq 0 \tag{16d}$$

$$\widehat{h}(\widehat{z}) = 0 \tag{16e}$$

with $\widehat{z}_m = (z_m, \widehat{s}^{\text{in}}_{r,m}(k+j), \widehat{s}^{\text{out}}_{r,m}(k+j) \; : \; \forall r, j)$, $\widehat{z} = (\widehat{z}_m \; : \; m \in \mathcal{M})$, and the constraints $\widehat{h}$ and $\widehat{g}$ given by:

$$\begin{aligned} &\text{For all } m \in \mathcal{M},\; r \in \mathcal{R},\; j \in \mathcal{N} : \\ &s'_{r,m} u_m(k+j) + \widehat{s}^{\text{out}}_{r,m}(k+j) = \widehat{s}^{\text{in}}_{r,m}(k+j) \end{aligned} \tag{17a}$$

$$\widehat{s}^{\text{in}}_{r,m}(k+j), \widehat{s}^{\text{out}}_{r,m}(k+j) \geq 0 \tag{17b}$$

$$\begin{aligned} &\text{For all } r \in \mathcal{R},\; j \in \mathcal{N} : \\ &\widehat{s}^{\text{in}}_{r,1}(k+j) = s^{\max}_r(k+j) \end{aligned} \tag{17c}$$

$$\text{For all } m \in \mathcal{M} \setminus \{M\},\ r \in \mathcal{R},\ j \in \mathcal{N}: \\ \widehat{s}^{\text{out}}_{r,m}(k+j) = \widehat{s}^{\text{in}}_{r,m+1}(k+j) \tag{17d}$$

in which $\mathcal{R} = \{1, \dots, R\}$ and $\mathcal{N} = \{0, \dots, N_x - 1\}$.

The reformulation (16) and (17) of P enables a parallel implementation of the augmented Lagrangean algorithm, which is developed in the following section.

3 Solution methods

Three hierarchical methods are presented for dynamic optimization of the resource constrained systems. All methods allow parallel and distributed computation, whereby a master problem coordinates the solution of subproblems that can be solved concurrently or simultaneously.

3.1 Bilevel optimization

The bilevel formulation (3) and (4) for the MPC problem can be solved in two ways:

1. The first strategy makes explicit the first-order KKT conditions of the lower-level problems in the master problem. The resulting problem is a nonconvex NLP though, due to the complementary conditions involved in the inequality constraints and their respective Lagrange multipliers. Such a problem is a special case of Mathematical Programming with Equilibrium Constraints (MPEC) (Colson et al. 2007).
2. The second approach stems from the optimization of the master in the space of resource allocations ss, followed by the optimization of the lower-level subproblems and the computation of sensitivities $\partial f_m / \partial ss$ to guide the upper level algorithm.

The MPEC approach is not desirable for being more complex than the original problem (a quadratic problem, QP). Instead, we adopt the two-level approach in which the master optimizes the resource allocations with the aid of the subproblem sensitivities. Besides enabling the parallel solution of the subproblems, the sensitivities can be easily calculated because the objective function of the master is a weighted sum of the objectives of the subproblems. Therefore, the master problem U given by Eq. (3) can be solved by any gradient-based method, here solved by the interior-point method (IP) of Wächter and Biegler (2006).

For the application of the bilevel optimization approach, the computation of the derivatives remains to be explained. The sensitives can be conveniently computed from the solution of the following variation of the lower-level subproblem,

$$L_m(ss^\star_m) : \; z_m(ss^\star_m) = \arg\min_{z_m} \; f_m(z_m) \tag{18a}$$

$$\text{s.t.} : \; h_m(z_m) = 0 \tag{18b}$$

$$g_m(z_m) \leq 0 \tag{18c}$$

$$\begin{aligned} &\text{For all } r \in \mathcal{R}, j \in \mathcal{N} : \\ &\quad s'_{r,m} u_m(k+j|k) \leq s_{r,m}(k+j) \end{aligned} \tag{18d}$$

$$s_{r,m}(k+j) - ss^\star_{r,m}(k+j) = 0 \tag{18e}$$

such that $\partial f(ss_m)/\partial ss_{r,m}(k+j)|_{ss=ss^\star} = \lambda_{r,m}(k+j)$ in which $\lambda_{r,m}(k+j)$ is the Lagrange multiplier associated with constraint (18e) for the KKT conditions at the solution $z_m(ss^\star_m)$. Notice that the allocation vector $ss^\star_m$ is informed by the master problem, which is then taken as fixed in (18), whereas ss_m are local variables.

The bilevel approach, just described, assumes that the initial allocation ss renders the subproblems $\{L_m(ss_m)\}$ feasible, otherwise the sensitivities would not be defined. However, infeasibility could be handled with an ℓ_1 penalty based on a constraint relaxation, akin to the phase I strategy used in convex optimization (Boyd and Vandenberghe 2004). The phase I strategy defines the subproblem L_m much like the feasible subproblem of the Benders decomposition, whereby the slack variable γ_m is minimized and the constraints are like in BF_m. The objective function of the master would be the sum of the γ_m variables. If a solution ss is found to drive the master's objective to zero, then ss is a feasible allocation which can be used as a starting point.

3.2 Benders decomposition

The hierarchical approach derived from Generalized Benders decomposition (Geoffrion 1972; Benders 1962) consists of solving the master problem, updating the lower bound, and solving the subproblems. If the latter are all feasible, and solutions are obtained, then the upper bound is updated and an optimality cut is produced. Otherwise, the upper bound is not updated and the feasibility subproblems must be solved to generate a feasibility cut. The process is repeated until convergence is achieved.

Algorithm 1 formalizes the Benders decomposition. The algorithm does not require a feasible starting point, since it can produce a feasible solution if one exists with the aid of the feasibility cuts.

Algorithm 1: Benders Decomposition Algorithm

```
input: initial lower bound $lb^{(0)} := -\infty$ and upper bound $ub^{(0)} := \infty$, and tolerance
       $\tau > 0$;
$\mathcal{O}^{(0)} := \emptyset$, $\mathcal{F}^{(0)} := \emptyset$ ;
$p := 0$;
repeat
    Solve the master problem $BM(p)$ given in (13) to obtain a solution $ss^{(p)}$ and
     objective $\alpha_{\mathrm{B}}^{(p)}$;
    Update lower bound: $lb^{(p+1)} := \max\{lb^{(p)}, \alpha_{\mathrm{B}}^{(p)}\}$;
    Solve $\{BO_m(ss_m^{(p)})\}_{m \in \mathcal{M}}$ and obtain a solution $z^{(p)}$ for $BO(ss^{(p)})$;
    if $z^{(p)}$ is feasible for $BO(ss^{(p)})$ then
        Update upper bound: $ub^{(p+1)} := \min\{ub^{(p)}, f(z^{(p)})\}$;
        Add an optimality cut (8) to $\mathcal{O}^{(p)}$ and obtain $\mathcal{O}^{(p+1)}$;
        Let $\mathcal{F}^{(p+1)} := \mathcal{F}^{(p)}$;
    else
        Keep upper bound: $ub^{(p+1)} := ub^{(p)}$;
        Solve $\{BF_m(ss_m^{(p)})\}_{m \in \mathcal{M}}$ and obtain a solution $(ss^{(p)}, \gamma^{(p)})$ for $BF(ss^{(p)})$;
        Add a feasibility cut (12) to $\mathcal{F}^{(p)}$ and obtain $\mathcal{F}^{(p+1)}$;
        Let $\mathcal{O}^{(p+1)} := \mathcal{O}^{(p)}$;
    $p := p + 1$;
until $(ub^{(p)} - lb^{(p)}) \leq \tau$;
output: $z^{(p)}, lb^{(p)}, ub^{(p)}$
```

3.3 Lagrangean decomposition

The reformulation (16) and (17) of the MPC problem couples the subsystems only through the consistency constraint (17d). The idea here is to dualize (17d) to obtain an approximation problem and solve it with the augmented Lagrangean algorithm. The augmented Lagrangean of (16) and (17), obtained by dualizing (17d), leads to the Lagrangean function $l(\lambda, \mu)$ computed as follows

$$
\begin{aligned}
\mathcal{L}(\lambda, \mu) \, : \; l(\lambda, \mu) = \min_{\hat{z}} l(\hat{z}) = & \sum_{m=1}^{M} f_m(z_m) \\
- \sum_{m \in \mathcal{M} \setminus \{M\}} \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}} & \lambda_{r,m}(k+j) \left[\hat{s}_{r,m}^{\text{out}}(k+j) - \hat{s}_{r,m+1}^{\text{in}}(k+j) \right] \\
+ \frac{\mu}{2} \sum_{m \in \mathcal{M} \setminus \{M\}} \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}} & \left\| \hat{s}_{r,m}^{\text{out}}(k+j) - \hat{s}_{r,m+1}^{\text{in}}(k+j) \right\|^2
\end{aligned}
\tag{19}
$$

while being subject to constraints (16b)–(16d) and (17a)–(17c), in which $\lambda = (\lambda_{r,m}(k+j) \, : \, m \in \mathcal{M} \setminus \{M\}, \, r \in \mathcal{R}, \, j \in \mathcal{N})$ is the vector of Lagrange multipliers associated with (17d) and μ is the penalty factor.

Remark 2 The augmented Lagrangean problem $\mathcal{L}(\lambda, \mu)$ is convex, since its objective is a convex function and the constraints are all affine or linear.

Let us reformulate $\mathcal{L}$ in an equivalent form that brings about a distributed structure. The dual function (19) can be recast as

$$l(\hat{z}) = \sum_{m=1}^{M-1} \underbrace{\left[f_m(\hat{z}_m) - f_{m,m+1}(\hat{z}_m, \hat{z}_{m+1})\right]}_{l_m(\hat{z}_m, \hat{z}_{m+1})} + f_M(\hat{z}_M) \tag{20a}$$

in which

$$\begin{aligned} f_{m,m+1} = \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}} \Big\{ & \lambda_{r,m}(k+j)\left[\hat{s}^{\text{out}}_{r,m}(k+j) - \hat{s}^{\text{in}}_{r,m+1}(k+j)\right] \\ & -(\mu/2)\left\|\hat{s}^{\text{out}}_{r,m}(k+j) - \hat{s}^{\text{in}}_{r,m+1}(k+j)\right\|^2 \Big\} \end{aligned} \tag{20b}$$

which shows that subsystem m is only coupled to subsystem $m-1$ (if $m > 1$) and to $m+1$ (if $m < M$), as they are not coupled by constraints. We can harness the distributed structure with the *Nonlinear Gauss–Seidel* (GS) algorithm (Bertsekas and Tsitsiklis 1997) to yield a high degree of parallelism. Clearly $\mathcal{L}$ can be framed as the problem

$$\min_{\hat{z} \in \Omega} l(\hat{z}) = \sum_{m \in \mathcal{M} \setminus \{M\}} l_m(\hat{z}_m, \hat{z}_{m+1}) + l_M(\hat{z}_M) \tag{21}$$

where $\Omega \subseteq \mathfrak{R}^n$ is the feasible set, such that $\Omega = \Omega_1 \times \cdots \times \Omega_M$ and Ω_m is the feasible set for subproblem m. The GS algorithm updates the decision vector $\hat{z}$, at iteration p, coordinate by coordinate:

$$\begin{aligned} \hat{z}^{(p+1)}_m &= \arg \min_{\hat{z}_m \in \Omega_m} \; l(\hat{z}^{(p+1)}_1, \dots, \hat{z}^{(p+1)}_{m-1}, \hat{z}_m, \hat{z}^{(p)}_{m+1}, \dots, \hat{z}^{(p)}_M) \\ &= \arg \min_{\hat{z}_m \in \Omega_m} \; -\left\{ f_{m-1,m}(\hat{z}^{(p+1)}_{m-1}, \hat{z}_m) : m > 1 \right\} \\ &\quad + f_m(\hat{z}_m) - \left\{ f_{m,m+1}(\hat{z}_m, \hat{z}^{(p)}_{m+1}) : m < M \right\} \end{aligned} \tag{22}$$

Notice that when a subsystem m is being updated, the most recent information about the other variables is used. The GS iterative process converges under the following conditions (Bertsekas and Tsitsiklis 1997):

Theorem 1 *Assume that:*

- *$\Omega_1, \Omega_2, \dots, \Omega_M$ are nonempty closed convex subsets of $\mathfrak{R}^{n_1}, \mathfrak{R}^{n_2}, \dots, \mathfrak{R}^{n_M}$ respectively, in which n_m is the dimension of $\hat{z}_m$ and $n = n_1 + n_2 + \cdots + n_M$.*
- *$l : \mathfrak{R}^n \rightarrow \mathfrak{R}$ is continuously differentiable and convex on the set $\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_M$.*
- *for each m, l is a strictly convex function of $\hat{z}_m$, when the values of the other components of $\hat{z}$ are held constant.*

Let $\{\widehat{z}^{(p)}\}_p$ *be the sequence generated by the nonlinear Gauss–Seidel algorithm, assumed to be well defined. Then, every limit point of* $\{\widehat{z}^{(p)}\}_p$ *minimizes* l *over* Ω.

It can be noticed in the iterative process (22) that the update of the variables $\widehat{z}_m$ only affects, or otherwise are affected by, the variables $\widehat{z}_{m-1}$ and $\widehat{z}_{m+1}$. This structure allows the set $\mathcal{M}$ of subsystems to be partitioned in two sets, $\mathcal{M}_{\text{odd}} = \{m \in \mathcal{M} : m \text{ is an odd number}\}$ and $\mathcal{M}_{\text{even}} = \{m \in \mathcal{M} : m \text{ is an even number}\}$, which can be optimized in parallel by the GS algorithm. Thus, the GS iterative process can be carried out, according to the iteration number p, as follows:

- if the iteration p is an odd number:

$$\begin{cases} \widehat{z}_m^{(p+1)} &= \underset{\widehat{z}_m \in \Omega_m}{\arg\min}\; l(\widehat{z}_1^{(p)}, \ldots, \widehat{z}_{m-1}^{(p)}, \widehat{z}_m, \widehat{z}_{m+1}^{(p)}, \ldots, \widehat{z}_M^{(p)}),\ \forall m \in \mathcal{M}_{\text{odd}} \\ \widehat{z}_m^{(p+1)} &= \widehat{z}_m^{(p)},\ \forall m \in \mathcal{M}_{\text{even}} \end{cases} \tag{23a}$$

- if the iteration p is even:

$$\begin{cases} \widehat{z}_m^{(p+1)} &= \widehat{z}_m^{(p)},\ \forall m \in \mathcal{M}_{\text{odd}} \\ \widehat{z}_m^{(p+1)} &= \underset{\widehat{z}_m \in \Omega_m}{\arg\min}\; l(\widehat{z}_1^{(p)}, \ldots, \widehat{z}_{m-1}^{(p)}, \widehat{z}_m, \widehat{z}_{m+1}^{(p)}, \ldots, \widehat{z}_M^{(p)}),\ \forall m \in \mathcal{M}_{\text{even}} \end{cases} \tag{23b}$$

Algorithm 2 formalizes the augmented Lagrangean method. Being a dual method, a feasible starting point is readily obtained and a primal lower bound is predicted at each iteration.

Algorithm 2: Augmented Lagrangean Algorithm

```
input: initial Lagrange multipliers $\lambda^{(0)}$, penalty $\mu^{(0)}$, and update factor $\beta > 1$;
$q := -1$;
repeat
    $q := q + 1$;
    Solve $\mathcal{L}(\lambda^{(q)}, \mu^{(q)})$ with the GS algorithm (23) to obtain a solution $\widehat{z}^{(q)}$ with
      objective $l^{(q)}$;
    for $m \in \mathcal{M}$ do
        for $r \in \mathcal{R}$ do
            for $j \in \mathcal{N}$ do
                $\lambda_{r,m}^{(q+1)}(k+j) := \lambda_{r,m}^{(q)}(k+j) - \mu^{(q)} \cdot \left[\widehat{s}_{r,m}^{\text{out},(q)}(k+j) - \widehat{s}_{r,m+1}^{\text{in},(q)}(k+j)\right]$;
    $\mu^{(q+1)} := \beta \cdot \mu^{(q)}$;
until convergence of $\{\widehat{z}^{(q)}\}$;
output: $\widehat{z}^{(q)}, l^{(q)}$
```

3.4 Numerical experiments

The experiments aim to validate the hierarchical approaches and obtain insights into their performance.

Instances of the MPC problem were obtained in a random fashion by sampling the parameters from uniform distributions. Specifically, the dynamic matrices were sampled from a uniform distribution in the range [0, 1], with all subsystems having the following dimensions: $x_m \in \mathbb{R}^2$, $y_m \in \mathbb{R}$, and $u_m, \Delta u_m \in \mathbb{R}^2$. The error cost matrix Q_m was the identity I, while the control variation cost matrix W_m was $0.1I$. The number of subsystems were chosen from the set $\{20, 40, 80, 120\}$. The control and prediction horizons had the same length, being chosen from the set $\{4, 6, 8, 10, 12\}$. Bounds for the variables were set as follows: $y^{\min} = -4$, $y^{\max} = 4$, $u^{\min} = 0$, $u^{\max} = 3$, $\Delta u^{\min} = -3$, and $\Delta u^{\max} = 3$. One resource was shared by the subsystems with availability $s^{\max} = 2$ for all sample times.

The algorithms were implemented in the Julia language (Bezanson et al. 2017) using the solvers IPOPT and Gurobi, which offer a high level interface for solving optimization problems. The implementations were based on a single processor, with parallel iterations emulated serially. A maximum computation time of 600 s was set for all algorithms. Details are given below:

- *Bilevel* The bilevel approach was relatively simple to implement because the subproblems $L_m(ss^\star_m)$ are QPs derived from the MPC problem, as defined in Eq. (18). The subproblems were solved with Gurobi. The master problem was solved with IPOPT, for which we provided the objective derivatives from the Lagrange multipliers of the subproblems. The tolerance was set at 10^{-4} for both solvers.
- *Benders* The Benders approach was arguably the most complex to implement. Besides keeping track of Lagrange multipliers for the constraints, two types of subproblems were implemented: one for the optimality cut which produces an upper bound when feasible, and the other for the feasibility cut. The master and subproblems were solved by Gurobi with a tolerance of 10^{-4}. The duality gap tolerance was set at $\tau = 10^{-4}$.
- *Lagrangean* Taking advantage of the problem structure, the Lagrangean dual (21) was solved with the Gauss–Seidel method (23) implemented in the Julia language. The subproblems in (23) were solved with IPOPT for each subsystem m, while holding the neighboring variables fixed. The parameters for the algorithm were: $\lambda^{(0)} = 0$, $\beta = 1.02$, $\mu^{(0)} = 0.1$, and a tolerance for infeasibility of 10^{-5}. The Lagrangean approach was relatively simple to implement.

The results of the numerical experiments appear in Table 1 regarding the objective value, and in Table 2 the results regarding the computational time. The column labeled $f^\star$ gives the optimal objective, which was obtained by solving the MPC problem in a centralized manner using Gurobi.

To illustrate the performance of the approaches, we present the trajectory of their iterations for the problem instances with $M = 40$ subsystems. For the bilevel approach, Fig. 1 shows the percent deviation of the solution with respect to the

Table 1 Computational analysis of hierarchical decompositions: objective function

M	N_x	$f^\star$	Decompositions		
			Bilevel	Lagrange	Benders
20	4	0.1328	0.1328	0.1328	0.1856
	6	0.5196	0.5196	0.5196	0.5939
	8	4.3114	4.3114	4.3114	4.4041
	10	32.7168	32.7168	32.7168	32.8272
	12	226.7523	226.7523	226.7523	226.8803
40	4	0.6702	0.6702	0.6702	0.8087
	6	2.7449	2.7450	2.7450	3.2688
	8	21.1918	21.1918	21.1918	21.9595
	10	161.0574	161.0574	161.0574	162.0576
	12	1171.3187	1171.3187	1171.3187	1172.5401
80	4	1.2777	1.2777	1.2777	1.7551
	6	5.4551	5.4551	5.4551	6.3938
	8	42.3531	42.3531	42.3531	43.7159
	10	321.3196	321.3196	321.3196	323.1018
	12	2333.7309	2333.7310	2333.7310	2335.9122
120	4	2.8080	2.8080	2.8080	3.6255
	6	9.1553	9.1554	9.1553	10.9149
	8	64.4063	64.4067	64.4063	67.1715
	10	482.7852	482.8046	482.8047	486.5243
	12	3501.3631	3501.3631	3501.3838	3506.0097

optimum as a function of CPU time (in seconds). For the Benders decomposition, Fig. 2 provides the percent deviation of the primal and dual solutions. For the Lagrangean decomposition, Fig. 3 shows the relative deviation of the dual solution to the optimum and its infeasibility measure, being defined as the ℓ_1 norm of the consistency constraints (17d).

3.5 Discussion

The implementations and the numerical experiments elicit some remarks:

- The bilevel optimization was the simplest to implement, since the solution of the subproblems and sensitivity computations were transparent for the master solver. The approach consistently converged to the optimum with the given tolerance, as illustrated in Fig. 1. The bilevel approach was somewhat slower than the Lagrangean decomposition, but much faster than Benders.
- The Benders decomposition was the most complex strategy for implementation, as it requires to solve the optimality and feasibility problems. This approach was

Table 2 Computational analysis of hierarchical decompositions: CPU time (in seconds)

M	N_x	Centralized	Decompositions		
			Bilevel	Lagrange	Benders
20	4	0.0699	1.89	0.0155	600
	6	0.0142	1.92	0.0245	600
	8	0.0202	2.49	0.0340	600
	10	0.0273	3.41	0.0457	600
	12	0.0342	3.10	0.0579	600
40	4	0.0145	45.28	7.05	600
	6	0.0394	66.02	13.79	600
	8	0.0547	49.80	22.70	600
	10	0.0544	197.85	37.96	600
	12	0.0654	199.10	72.68	600
80	4	0.0280	119.62	14.42	600
	6	0.0518	115.65	28.09	600
	8	0.0924	184.68	55.06	600
	10	0.1193	335.89	158.28	600
	12	0.1424	248.19	207.26	600
120	4	0.0600	138.66	21.87	600
	6	0.0913	118.01	41.17	600
	8	0.1260	189.86	80.62	600
	10	0.2037	600.00	256.60	600
	12	0.2476	442.79	367.49	600

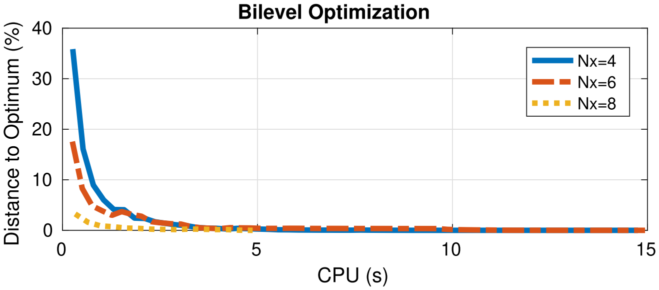

Fig. 1 Solution trajectories obtained by the bilevel approach for the system with $M = 40$ subsystems, considering a varying length N_x for the prediction horizon

the least efficient, reaching the maximum CPU limit without reaching the tolerance for the primal-dual gap, and often not reaching the optimum. A possibility to reduce computational time would be to solve the dual of the master, which could be warm started since cuts render suboptimal the incumbent dual solution, rather than primal infeasible. Figure 2 illustrates the slow convergence, specially regarding the lower bound.

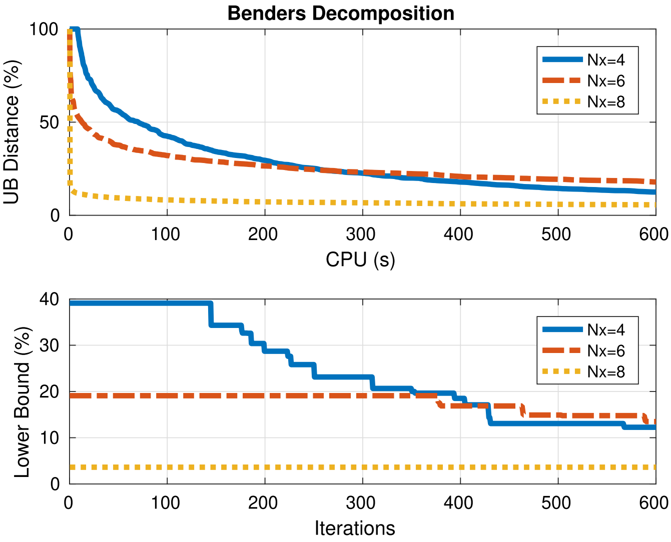

Fig. 2 Solution trajectories obtained by Benders decomposition for the system with $M = 40$ subsystems, considering a varying length N_x for the prediction horizon

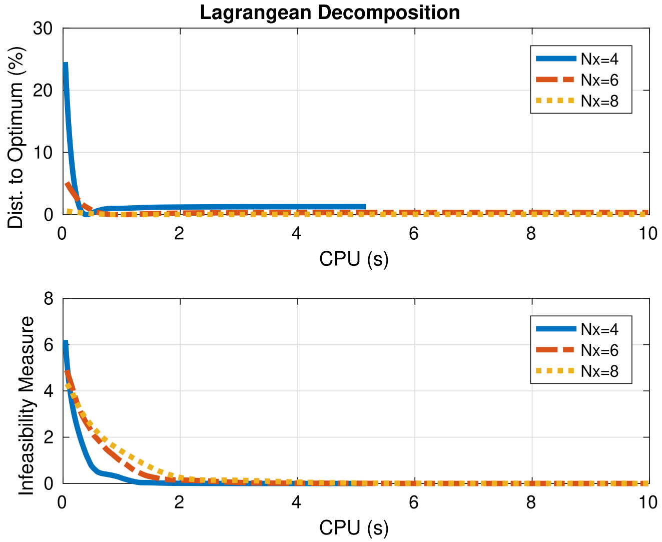

Fig. 3 Solution trajectories obtained by Lagrangean decomposition for the system with $M = 40$ subsystems, considering a varying length N_x for the prediction horizon

The generation of multiple Benders cuts as proposed by You and Grossmann (2013), one for each subsystem, was applied but its performance proved to be inferior to the standard optimality and feasibility cuts.

- The Lagrangean decomposition was relatively simple to implement. It converged to the optimum at a faster pace than the other approaches, with further iterations needed to drive the consistency constraint towards feasbility. This behavior is showcased in Fig. 3.
- Not surprisingly, the solution time of the centralized problem is considerably lower than the decomposition approaches as shown in Table 2. The overhead on communications and the solution of multiple problems imposes additional computational costs.

  The benefit of the hierarchical decompositions is organizational, as they enable the control system to be reconfigured locally and expanded with reduced coordination. Notice that the signals communicated between the master problem and subproblems are relatively simple, consisting of resource allocations or Lagrange multipliers (from master to subsystems), and sensitivities, cuts or resource usage (from subsystems to master). The master problem does not need to have detailed information on the subproblems.

4 Application to HVAC systems

This section reports on results yielded by the hierarchical approach applied to a representative case study of energy management. The problem concerns the distribution of chilled water to HVAC units of a building in order to promote thermal comfort.

4.1 CIESOL case study

For the purpose of evaluation and illustration of the methods developed, a practical study was carried out in the system represented in Fig. 4. This system is a model of the solar plant located at the Campus of the University of Almería, in the South East of Spain, being part of Centro de Investigación de Energía Solar (CIESOL), as described by Pasamontes et al. (2009).

This system involves the operation of the air-conditioning plant in an efficient building, for which the energy is generated by a solar plant and which should be distributed to a series of HVAC systems (each room of the building). In a simplified way, water fluid is used to transfer heat between the supply and consumers.

The circuit consists of a pipe that connects the absorption machine to the HVAC subsystems of the CIESOL building. Near the absorption machine there is a pump which delivers hot or chilled water, and in each room there is a heat exchanger, here considered a HVAC unit, that has a controlled valve which is used to regulate the flow in the HVAC unit. It should be mentioned that the cycle of the fluid energy is closed, that is, the fluid returns to the pump after passing through the HVAC systems. Physical and operational details of this air-conditioning system are found in Pasamontes et al. (2009) and Castilla et al. (2011).

Considering only the set of rooms of the building, the flow of water is divided by the control system to promote thermal comfort to the users. In each room, there is a heat exchanger (fancoil) governed by the following dynamics (Álvarez et al. 2007):

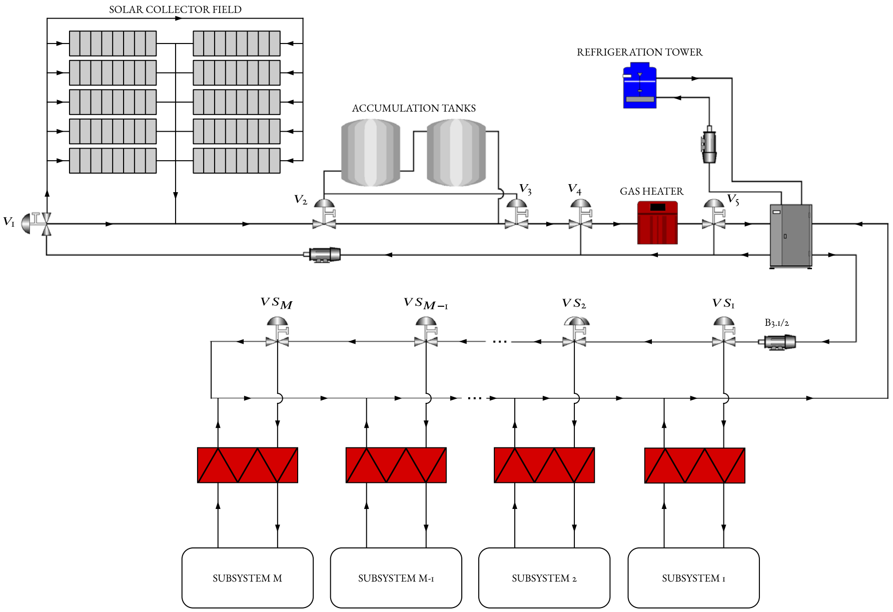

Fig. 4 Solar cooling/heating installation scheme. (Extracted from Scherer et al. 2013)

$$\frac{dT_{w,o}}{dt} = -\dot{q}_w \frac{(T_{w,o} - T_{w,i})}{V_w} + \frac{1}{\tau_w}(T_{a,imp} - T_{w,o}) \tag{24a}$$

$$\frac{dT_{a,imp}}{dt} = -v_a \frac{(T_{a,imp} - T_{a,ret})}{L} - \frac{1}{\tau_a}(T_{a,imp} - T_{w,o}) \tag{24b}$$

These equations show that the fancoil rate of chilled water ($\dot{q}_w$) and fan speed (v_a) can be manipulated to induce thermal comfort inside the room. Figure 5 illustrates the fancoil. More specifically, the HVAC is modeled as a parallel-flow of water and air, which both enter the exchanger at the same time and travel through the fancoil

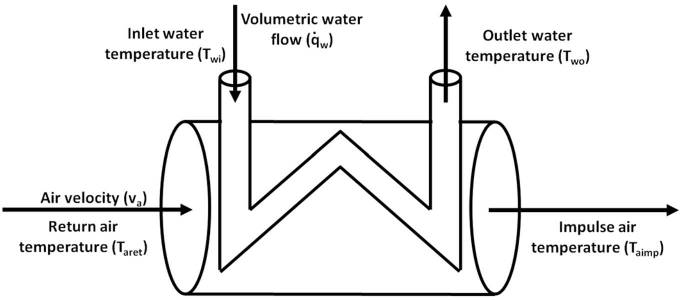
Inlet water temperature (T_{wi})
Volumetric water flow ($\dot{q}_w$)
Outlet water temperature (T_{wo})
Air velocity (v_a)
Return air temperature (T_{aret})
Impulse air temperature (T_{aimp})

Fig. 5 Schematic of HVAC fancoil. (Extracted from Scherer et al. 2013)

in parallel. The subscript w refers to the variables associated with *water* and a with *air*, and the subscripts i, o, *imp*, and *ret* mean *input*, *output*, *impulse*, and *return*, respectively.

In this case, the rooms are connected in series and the available resource may be depleted, reducing cooling efficiency in the environments located farther away from the pump. For the controller development, the linearized model of this system was adopted, considering the ratio given by the input water flow ($\dot{q}_w$) and output air temperature ($T_{a,imp}$). For this specific case, the linearization considered constant the input variables $\dot{q}_w = 2.5$ m^3/s, $T_{w,i} = 8$ °C, $T_s = 3$ s (sample time), and $V_{imp} = 0.4$ m^3/s. The system is linearized using these constant values and an operating point within 100 s after the start of operation, allowing for a steady state. After the state-space matrices are obtained, an expansion is carried out to accommodate the system outputs at the state vector, leaving the model in the desired shape for the application framed in the MPC model of Eq. (1).

4.2 Simulation experiments

The objective function adopted seeks to minimize the error of reference temperature tracking, while minimizing the flow use of the plant. The set of constraints must encompass maximum and minimum limits of output, control, and control variation. For the experimental set-up, the resource consists of the available rate of chilled water, modeled as $s_r^{\max}(k)$, which vary over time depending on the prevailing conditions of the solar plant. Putting together the dynamic models for each HVAC unit, the constraints on outputs and control signals, and the resource constraints, we arrive at the MPC formulation given by Eq. (1).

The MPC problem was solved using the bilevel decomposition, for being relatively simple to implement and its good performance in the numerical experiments. The Lagrange decomposition could be selected as well, since it consistently reached a nearly optimal solution after a few iterations. Actually, because these approaches produce optimal solutions, their control performance should approach the performance obtained by solving the MPC problem in a centralized fashion. The bilevel decomposition was implemented in Matlab using the YALMIP toolbox (Löfberg 2004), and `fmincon` and `quadprog` to solve the master and subproblems. The algorithm was sufficiently fast to reach a nearly-optimal solution within the sampling time. Given this characteristic of fast convergence, it was possible to consider that the computational delay is not an issue in this case (Wang 2009). Thereby, the MPC was implemented using an ideal formulation, without taking into account this kind of delay in the system model.

For the simulations, the following parameters were adopted: $N_u = 10$, $N_x = 30$, $Q_m = I$ and $W_m = 5I$, where I is an identity matrix of suitable dimension. To simplify the problem and facilitate the visualization of the results, the references in all the subsystems will be 18 °C. However, any other configuration is admissible. With respect to the simulation environment, a simulator of the CIESOL plant that adheres to the aforementioned nonlinear dynamics was employed. Implemented in Matlab with Simulink, this simulator receives real data from the operation of the solar plant and generates the signals at any point of interest.

For the case study, the temperature and flow signals are obtained from the simulator. The control algorithm is executed and the current control value is fed back into the simulator to be implemented in each subsystem. No control signal was applied during the first 3 min of the simulation, which explains the initial behavior observed in the simulation analysis that follows. This policy was followed because the variables were not in a steady mode, which could generate distorted values for the control algorithm and the state observers.

4.3 Simulation and comparison of results

For the purpose of comparison, simulations were performed in three scenarios:

- The first one uses a PID controller in each HVAC system,
- the second uses isolated MPC controllers in each environment, not taking into account aspects of cooperation between them, and
- the last scenario uses the proposed hierarchical decomposition to solve the MPC problem given by Eq. (1).

The simulation results illustrated in Figs. 6, 7, and 8 provide three subplots, presenting the plant behavior obtained for a 30-min simulation run. Even for a small time window, remarkable characteristics of the operations can be noticed.

The top most plot depicts the output temperatures in each subsystem, along with the output temperature of the water coming from the solar plant. The middle plot shows the control actions (water flows) which begin after 3 min of operation. The

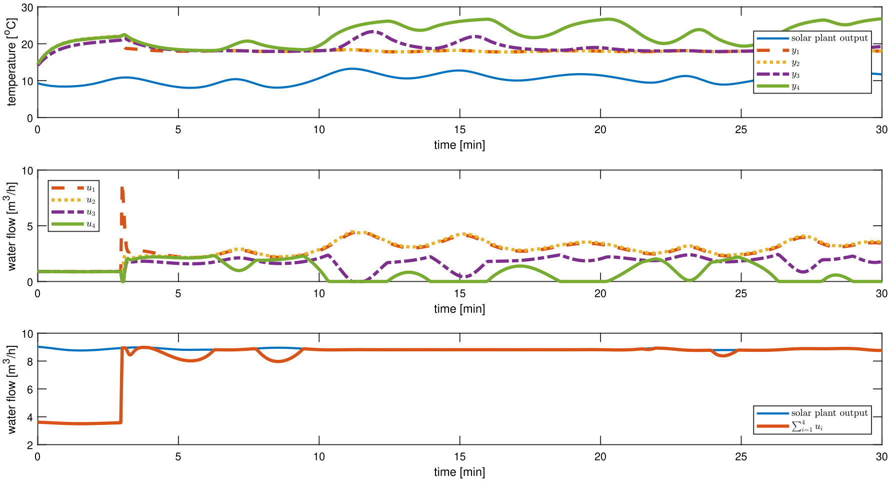

Fig. 6 System behavior under PID control

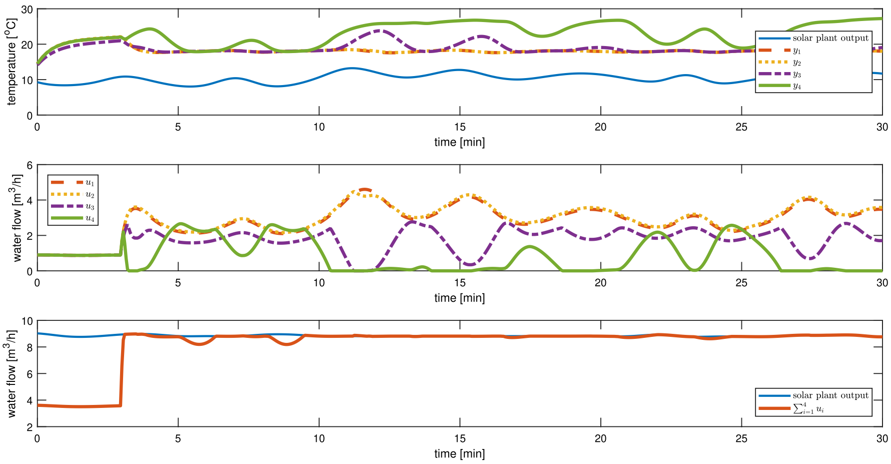

Fig. 7 System behavior under non-cooperative distributed MPC

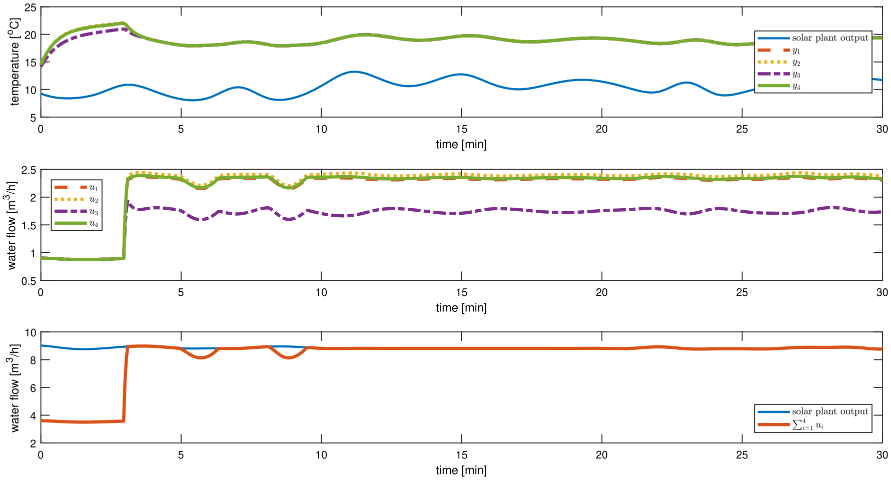

Fig. 8 System behavior induced by the hierarchical decomposition

bottom most plot presents the available water flow over time ($s^{\max}$) and the sum of the control actions. A discussion on the results of the three scenarios follows:

1. In the first test, the system behavior was obtained by PID controllers. Their use is justified by the simplicity of implementation, however the analyses demonstrate that the problem is not so simple, given the effects of the coupling constraint. A digital PID was implemented and tuned with the IMC method (Åström and Wittenmark 1997; Skogestad 2003), with post manual adjustment to obtain a good balance between reference tracking and input disturbance rejection. The parameters used were $K_c = -2.50$, $T_i = 50.0$, and $T_d = -0.3125$.

Figure 6 shows the behavior for the tested settings under PID control. Notice that when enough resources are available, like at minute 6, all environments reach their reference. However, when the resources are not sufficient, the obtained results are not satisfactory due to the non-cooperative control, which causes the environments at the end of the water pipeline to suffer from the depletion of resources. At some points in time it is possible to observe that even output y_3 is affected by the lack of resources. As shown in Fig. 6, the temperature at HVAC environment 4 deviates significantly from the 18 °C reference, indicating that the PID approach is not satisfactory.

2. For the second test case, predictive controllers were used in each environment, but without taking into account the behavior of their neighbors, a strategy regarded as non-cooperative distributed MPC (Scherer et al. 2015). Such a control structure is relatively simple to implement, which would be ideal if there was no shortage of resources. The top plot of Fig. 7 gives the output temperatures of each subsystem and the solar plant. The middle plot presents the water flows (control actions) for the environments. The bottom plot depicts the total water rate output by the solar plant, and the total rate consumed by the HVAC units.

   As in the case of PID controllers, the strategy proved to be inadequate for this type of problem. The solution presented is far from an optimal solution to the problem. One can easily notice that subsystem 4 can not keep track for the reference at 18 °C, which causes even subsystem 3 to raise its temperature. This behavior is verified in the middle plot of the figure, where control actions are reduced to zero because there is no available resource for HVAC units 3 and 4 during certain periods of time.

3. Finally, the bilevel decomposition was applied for the solution of the MPC problem, whose results are shown in Fig. 8.

   The simulation results demonstrate that a significant improvement in performance is obtained by the hierarchical decomposition, which managed to track the room temperatures equally around the reference at 18 °C.

The simulation analysis showed that hierarchical decompositions can be effective for MPC of resource-constrained dynamic systems. Specifically, the hierarchical decomposition implemented with bilevel optimization consistently converged to the solution of the MPC problem (1), inducing thermal comfort for all environments of the building. Besides that, the examples showed that the problem in question is not trivial, requiring a controller that oversees the whole plant, or which uses problem decomposition strategies to achieve an optimal result.

5 Final remarks

The thermal control system in a building is a dynamic system that arises from the interconnection of dynamic HVAC subsystems, one for each room, which share limited energy resources. In this work, such a distributed structure was harnessed by hierarchical approaches that split the computations between a master problem, responsible for coordination, and a set of subproblems that can be solved concurrently or in parallel. To that end, hierarchical approaches were derived from bilevel optimization, Benders and Lagrangean decompositions for the energy management and thermal regulation in buildings. The performance of these decompositions were assessed in numerical experiments, which consisted in solving the MPC problem for a host of dynamic systems. Among these approaches, bilevel decompositions was applied for the energy management and temperature regulation of the simulation model of the CIESOL building in Almeria, Spain.

A number of extensions and applications can be considered for future work:

- The system model may have the energy resource as a variable, rather than a given constant, in which case the objective would account for the energy cost. In such a situation, the objective of the master may be in conflict with the goals of the subsystems, a situation that would be suitable for bilevel decomposition.
- An extension arises from the application to continuous time systems, with the dynamics modeled by algebraic differential equations.
- The performance of Benders decomposition could be improved with regularization techniques, such as level regularization and local branching (Rei et al. 2009).

Acknowledgements The leading author acknowledges funding in part from CAPES/Brazil, under Grant Number 88881.119526/2016-01.

References

Álvarez JD, Yebra LJ, Berenguel M (2007) Repetitive control of tubular heat exchangers. J Process Control 17:689–701. https://doi.org/10.1016/j.jprocont.2007.02.003

Álvarez JD, Redondo JL, Camponogara E, Normey-Rico J, Berenguel M, Ortigosa PM (2013) Optimizing building comfort temperature regulation via model predictive control. Energy Build 57:361–372. https://doi.org/10.1016/j.enbuild.2012.10.044

Åström KJ, Wittenmark B (1997) Computer-controlled systems, 3rd edn. Prentice-Hall Inc, Upper Saddle River

Benders JF (1962) Partitioning procedures for solving mixed-variables programming problems. Numer Math 4(3):238–252

Bertsekas D (1995) Nonlinear programming. Athena Scientific, Belmont

Bertsekas DP, Tsitsiklis JN (1997) Parallel and distributed computation: numerical methods. Athena Scientific, Belmont

Bezanson J, Edelman A, Karpinski S, Shah VB (2017) Julia: a fresh approach to numerical computing. SIAM Rev 59(1):65–98. https://doi.org/10.1137/141000671

Boyd S, Vandenberghe L (2004) Convex optimization. Cambridge University Press, Cambridge

Boyd S, Parikh N, Chu E, Peleato B, Eckstein J (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Found Trends Mach Learn 3(1):1–122. https://doi.org/10.1561/2200000016

Camacho EF, Bordons C (2004) Model predictive control. Springer, Berlin

Camponogara E, Scherer HF (2011) Distributed optimization for model predictive control of linear dynamic networks with control-input and output constraints. IEEE Trans Autom Sci Eng 8:233–242. https://doi.org/10.1109/TASE.2010.2061842

Castilla M, Álvarez JD, Berenguel M, Rodríguez F, Guzmán JL, Pérez M (2011) A comparison of thermal comfort predictive control strategies. Energy Build 43(10):2737–2746. https://doi.org/10.1016/j.enbuild.2011.06.030

Castilla M, Álvarez JD, Normey-Rico JE, Rodríguez F (2014) Thermal comfort control using a non-linear MPC strategy: a real case of study in a bioclimatic building. J Process Control 24(6):703–713. https://doi.org/10.1016/j.jprocont.2013.08.009

Chen W, Shao Z, Biegler LT (2014) A bilevel NLP sensitivity-based decomposition for dynamic optimization with moving finite elements. AIChE J 60(3):966–979. https://doi.org/10.1002/aic.14339

Colson B, Marcotte P, Savard G (2007) An overview of bilevel optimization. Ann Oper Res 153(1):235–256. https://doi.org/10.1007/s10479-007-0176-2

D&R International Ltd (2009) Building energy data book. Technical report, U.S. Department of Energy

Escrivá-Escrivá G, Segura-Heras I, Alcázar-Ortega M (2010) Application of an energy management and control system to assess the potential of different control strategies in HVAC systems. Energy Build 42(11):2258–2267. https://doi.org/10.1016/j.enbuild.2010.07.023

Geoffrion AM (1972) Generalized benders decomposition. J Optim Theory Appl 10(4):822–844

Guignard M, Kim S (1987) Lagrangean decomposition: a model yielding stronger Lagrangean bounds. Math Program 39(2):215–228

Löfberg J (2004) YALMIP: a toolbox for modeling and optimization in MATLAB. In: Proceedings of the IEEE international symposium on computer-aided control system design. Taipei, Taiwan

Maasoumy M, Razmara M, Shahbakhti M, Vincentelli AS (2014) Handling model uncertainty in model predictive control for energy efficient buildings. Energy Build 77:377–392. https://doi.org/10.1016/j.enbuild.2014.03.057

Moroşan PD, Bourdais R, Dumur D, Buisson J (2010) Building temperature regulation using a distributed model predictive control. Energy Build 42:1445–1452. https://doi.org/10.1016/j.enbuild.2010.03.014

Pasamontes M, Álvarez JD, Guzmán JL, Berenguel M (2009) Hybrid modeling of a solar cooling system. In: Proceedings of the IFAC international conference on analysis and design of hybrid systems, ADHS09. Zaragoza, Spain

Pérez-Lombard L, Ortiz J, Pout C (2008) A review on buildings energy consumption information. Energy Build 40:394–398. https://doi.org/10.1016/j.enbuild.2007.03.007

Rei W, Cordeau JF, Gendreau M, Soriano P (2009) Accelerating benders decomposition by local branching. INFORMS J Comput 21:333–345. https://doi.org/10.1287/ijoc.1080.0296

Ruscio D (2013) Model predictive control with integral action: a simple MPC algorithm. Model Identif Control Nor Res Bull 34:119–129

Salakij S, Yu N, Paolucci S, Antsaklis P (2016) Model-based predictive control for building energy management. I: energy modeling and optimal control. Energy Build 133:345–358. https://doi.org/10.1016/j.enbuild.2016.09.044

Scattolini R (2009) Architectures for distributed and hierarchical model predictive control—a review. J Process Control 19(5):723–731. https://doi.org/10.1016/j.jprocont.2009.02.003

Scherer H, Álvarez JD, Guzmán JL, Camponogara E, Normey-Rico J (2013) Efficient building energy management using distributed model predictive control. J Process Control 24:740–749. https://doi.org/10.1016/j.jprocont.2013.09.024

Scherer H, Camponogara E, Normey-Rico J, Álvarez JD, Guzmán JL (2015) Distributed MPC for resource-constrained control systems. Optim Control Appl Methods 36:272–391. https://doi.org/10.1002/oca.2151

Skogestad S (2003) Simple analytic rules for model reduction and PID controller tuning. J Process Control 13(4):291–309. https://doi.org/10.1016/S0959-1524(02)00062-8

Terrazas-Moreno S, Trotter PA, Grossmann IE (2011) Temporal and spatial Lagrangean decompositions in multi-site, multi-period production planning problems with sequence-dependent changeovers. Comput Chem Eng 35:2913–2928. https://doi.org/10.1016/j.compchemeng.2011.01.004

Wächter A, Biegler LT (2006) On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math Program 106(1):25–57. https://doi.org/10.1007/s10107-004-0559-y

Wang L (2009) Model predictive control system design and implementation using MATLAB, 1st edn. Springer, Berlin

You F, Grossmann IE (2013) Multicut benders decomposition algorithm for process supply chain planning under uncertainty. Ann Oper Res 210(1):191–211. https://doi.org/10.1007/s10479-011-0974-4

Yu N, Salakij S, Chavez R, Paolucci S, Sen M, Antsaklis P (2017) Model-based predictive control for building energy management: part II—experimental validations. Energy Build 146:19–26. https://doi.org/10.1016/j.enbuild.2017.04.027

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Eduardo Camponogara[1] · **Helton Scherer**[2] · **Lorenz Biegler**[3] · **Ignacio Grossmann**[3]

Helton Scherer
hfscherer@yahoo.com.br

Lorenz Biegler
biegler@cmu.edu

Ignacio Grossmann
grossmann@cmu.edu

[1] Department of Automation and Systems Engineering, Federal University of Santa Catarina, Florianópolis, SC 88040-900, Brazil

[2] Itaipu Technological Park, Foz do Iguaçu, PR 85867-900, Brazil

[3] Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA