Coordination of Distributed Model Predictive Controllers for Constrained Dynamic Processes ⋆

Natalia I. Marcos *, **J. Fraser Forbes** * and **Martin Guay** **

* *Department of Chemical and Materials Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2V4*
** *Department of Chemical Engineering, Queen's University, Kingston, Ontario, Canada K7L 3N6*

Abstract: In this paper, a coordinated-distributed model predictive control (MPC) scheme is presented for large-scale discrete-time linear process systems. Coordinated-distributed MPC control aims at enhancing the performance of fully decentralized MPC controllers by achieving the plant-wide optimal operations. The 'price-driven' decomposition-coordination method is used to adjust the operations of the individual processing units in order to satisfy an overall plant performance objective. Newton's method, together with a sensitivity analysis technique, are used to efficiently update the price in the price-driven decomposition-coordination method. The efficiency of the proposed control scheme is evaluated using a model of a fluid catalytic cracking process.

Keywords: Decomposition-coordination methods; Large-scale optimization; Optimal control theory.

1. INTRODUCTION

Since the late seventies, the design of chemical processes has evolved towards integrated operations that have increased plant's efficiency. The improvement in the design of chemical processes included, among other things, energy and mass integration, and the use of recycle streams. As a result, processes became more complex and processing units became more tightly interconnected. Control of such integrated large-scale processes has been typically performed with *decentralized* schemes because of the difficulties in implementation and maintenance of *centralized* control frameworks.

Centralized and decentralized control are two distinct control strategies. In centralized control, no real distinction is made among processing units. The centralized control framework is formulated as a monolithic control problem that incorporates all process variables with no unit-level decomposition. While a centralized strategy can lead to optimal plant-wide performance, it presents some disadvantages (e.g., the large-dimensionality of the control problem and lack of flexibility in terms of operation and maintenance), which make centralized control unsuitable for industrial processes. In decentralized control, each engineering unit is optimized separately by neglecting the interactions with the other units. The decentralized approach is the most commonly used in the industry because of its robustness and its resiliency to systems failures. Nevertheless, decentralized control does not generally lead to the desired plant-wide optimal operations (Lu (2003); Sun and El-Farra (2008)).

⋆ This work is supported by Natural Sciences and Engineering Research Council of Canada (NSERC) and Alberta Ingenuity.

A compromise between centralized and decentralized control is desired in order to improve plant operations. *Distributed* control has emerged as a promising control strategy that can lead to the plant-wide optimal operations, while keeping manageable controllers for each subunit in the plant. In the distributed control framework, it is assumed that each subsystem computes its own optimal solution while considering all or certain degree of interactions with the other subsystems. To attain the desired control performance, information related to each subsystems' optimal solutions is generally exchanged among the subsystems. In this work, we present a coordinated-distributed model predictive control (CDMPC) framework for constrained dynamic processes. In CDMPC control, data is exchanged with each individual MPC controller via a 'coordinator' or 'master'.

1.1 Distributed MPC Control

Distributed MPC control has attracted the attention of many researchers in recent years. Dunbar and Murray (2004) formulated MPC platforms for nonlinear interacting subsystems (multi-vehicle formations) whose state variables are coupled in a single objective function. For linear interconnected systems, Venkat et al. (2005) proposed a communication-based MPC that can converge to a Nash equilibrium. The communication-based MPC was further improved by a cooperation-based MPC that leads to the Pareto optimal feasible solution. Cheng et al. (2008, 2007) proposed a coordinated scheme for MPC steady state target calculation based on Dantzig-Wolfe decomposition and price-driven coordination methods, respectively.

The main contribution of this work is to propose the *price-driven* decomposition-coordination algorithm, as described in Cheng et al. (2007), for the control of constrained process systems whose dynamics are represented by discrete-time models. The CDMPC control scheme presented in this paper achieves the centralized optimal operations and can be implemented when step-response models are available for the process. Since our control formulation uses models obtained from step-test data, it does not need estimation of unavailable process variables (as it might be required when formulating MPC controllers based on state-space models). Furthermore, the proposed CDMPC control scheme allows for bias correction in the predicted outputs through feedback.

An illustration of CDMPC is shown in Fig. 1. The price-driven decomposition-coordination method is used in the formulation of the CDMPC controllers. In the price-driven decomposition-coordination method, the coordinator sets up a price, 'p', for the subsystems' interacting variables (Fig. 1). The price provided by the coordinator is then

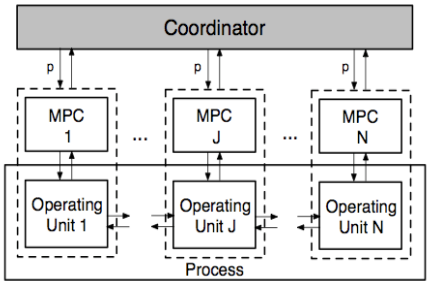

Fig. 1. Illustration of CDMPC Control

adjusted to alter the subunits' calculated control actions towards the overall plant optimum. In this work, the price, p, is updated based on Newton's method. An iterative procedure is established between the coordinator and the subunits until the desired plant-wide optimal solution is achieved.

2. CDMPC CONTROL FOR DYNAMIC PROCESS SYSTEMS

In this section, the CDMPC control scheme is presented. Since we consider the centralized performance as the ideal benchmark, we begin the CDMPC control formulation by decomposing the centralized control problem into N smaller subproblems that are easier to solve. Then, an efficient mechanism is used to achieve the same solution as the one obtained in the centralized control problem.

2.1 Process Model

Consider the overall plant process, modelled by step-response coefficients:

$$\begin{aligned} y_z(k+l) = & \sum_{h=1}^{T-1} \sum_{w=1}^{r} S_{zw,h} \Delta u_w(k+l-h) + \\ & \sum_{w=1}^{r} S_{zw,T}\ u_w(k+l-T), \\ & \forall z = 1, \ldots, m, \end{aligned} \tag{1}$$

where y_z $(\forall z = 1, ..., m) \in \Re^m$ denote the process outputs; $u_w \in \Re^r$ and $\Delta u_w \in \Re^r$ $(\forall w = 1, ..., r)$ denote the manipulated variables and the change in the manipulated variables, respectively. The coefficients $S_{11,h}, ..., S_{mr,h}$ represent the step-response coefficients for h^{th} time step. The step-response weight $S_{11,h}$ is the coefficient between Δu_1 and output y_1 for the h^{th} time step. In a similar manner, $S_{mr,h}$ is the coefficient between Δu_r and output y_m for the h^{th} time step.

2.2 Centralized MPC Formulation

For the centralized MPC implementation, it is convenient to arrange process model (1) in a matrix form as following:

$$\hat{Y}(k+1) = \mathcal{S}\Delta\hat{U}(k) + Y^0(k+1) + \hat{D}(k+1), \tag{2}$$

where the output variables, input variables and change in input variables predicted along the prediction horizon H_p and control horizon H_u are defined as:

$$\begin{cases} \hat{Y}(k+1) = [\hat{y}(k+1|k)^\top, ..., \hat{y}(k+H_p|k)^\top]^\top, \\ \hat{y}(.) = [\hat{y}_1(.), ..., \hat{y}_m(.)]^\top, \\ \Delta\hat{U}(k) = [\Delta\hat{u}(k|k)^\top, ..., \Delta\hat{u}(k+H_u-1|k)^\top]^\top, \\ \Delta\hat{u}(.) = [\Delta\hat{u}_1(.), ..., \Delta\hat{u}_r(.)]^\top, \hat{u}(.) = [\hat{u}_1(.), ..., \hat{u}_r(.)]^\top. \end{cases} \tag{3}$$

The $m \times H_p$ vector of unforced responses $Y^0(k+1)$ is:

$$\begin{cases} Y^0(k+1) = [y^0(k+1)^\top, ..., y^0(k+H_p)^\top]^\top, \\ y^0(.) = [y_1^0(.), ..., y_m^0(.)]^\top. \end{cases} \tag{4}$$

The vector $\hat{D}(k+1)$ has been incorporated in (2) to correct through feedback the discrepancies between the measured and predicted outputs. The vector $\hat{D}(k+1)$ is defined as:

$$\hat{D}(k+1) = \underbrace{[I_m, ..., I_m]^\top}_{H_p\ times}[y(k) - \hat{y}(k|k-1)],$$

where I_m is the $m \times m$ identity matrix. It is assumed that the difference between the measured and predicted outputs at time k remains constant throughout the prediction horizon.

In (2), the matrix of step-response coefficients $\mathcal{S}$ is defined as:

$$\mathcal{S} = \begin{bmatrix} S_1 & 0 & \cdots & 0 \\ S_2 & S_1 & 0 & 0 \\ \vdots & \vdots & \ddots & 0 \\ S_{H_u} & S_{H_u-1} & \cdots & S_1 \\ \vdots & \vdots & \ddots & \vdots \\ S_{H_p} & S_{H_p-1} & \cdots & S_{H_p-H_u+1} \end{bmatrix}, \tag{5}$$

where S_h is the $m \times r$ matrix of step-response coefficients for the h^{th} time step $(\forall h = 1, ..., H_p)$:

$$S_h = \begin{bmatrix} S_{11,h} & S_{12,h} & \cdots & S_{1r,h} \\ \vdots & \cdots & \cdots & \vdots \\ S_{m1,h} & \cdots & \cdots & S_{mr,h} \end{bmatrix}. \tag{6}$$

The centralized MPC controller is formulated to minimized the following objective function:

$$\min_{\hat{Y},\Delta\hat{U}} \quad \mathcal{J} = \frac{1}{2}\Big((Y_{sp} - \hat{Y}(k+1))^\top Q(Y_{sp} - \hat{Y}(k+1)) + \Delta\hat{U}(k)^\top R\Delta\hat{U}(k)\Big) \tag{7}$$

subject to:

$$\begin{cases} \text{Process model (2)-(6), and} \\ \hat{u}(k+l|k) = \sum_{h=0}^{l} \Delta\hat{u}(k+h|k) + u(k-1), \\ \Delta\hat{u}(k+h|k) = 0, \qquad H_u \le h \le H_p - 1, \\ y_{min} \le \hat{y}(k+l+1|k) \le y_{max}, \\ u_{min} \le \hat{u}(k+l|k) \le u_{max}, \\ \Delta u_{min} \le \Delta\hat{u}(k+n|k) \le \Delta u_{max}, \\ \forall l = 0,\ldots,H_p-1, \quad \text{and} \quad \forall n = 0,\ldots,H_u-1, \end{cases} \tag{8}$$

where Y_{sp} is the vector of desired set-points, $Q = diag\{Q(l+1)\}$ and $R = diag\{R(n)\}$ are positive definite matrices of appropriate dimensions.

2.3 Decomposition of Centralized MPC Formulation

We propose a decomposition of the overall optimization problem (7)-(8) into N subproblems 'i'. We consider that the plant dynamics and constraints can be decomposed into N subunits, followed by a block decomposition of the tuning matrices Q and R. As a result of the centralized problem decomposition, each subunit i ($\forall i : 1,\ldots,N$) solves its own optimization problem given by:

$$\min_{\hat{Y}_i,\Delta\hat{U}_i,\hat{V}_i} \mathcal{J}_i = \frac{1}{2}\Big((Y_{i\_sp} - \hat{Y}_i(k+1))^\top Q_{ii}(Y_{i\_sp} - \hat{Y}_i(k+1)) + \Delta\hat{U}_i(k)^\top R_{ii}\Delta\hat{U}_i(k)\Big) + p^\top\Theta_i Z_i(k) \tag{9}$$

subject to:

$$\hat{Y}_i(k+1) = \mathcal{S}_{ii}\Delta\hat{U}_i(k) + \hat{V}_i(k) + Y_i^0(k+1) + \hat{D}_i(k+1), \tag{10}$$

$$\hat{V}_i(k) = \mathcal{S}_{ij}\Delta\hat{U}_j(k), \quad \forall j \ne i, \tag{11}$$

and

$$\begin{cases} \hat{u}_i^s(k+l|k) = \sum_{n=0}^{l} \Delta\hat{u}_i^s(k+h|k) + u_i^s(k-1), \\ \Delta\hat{u}_i^s(k+h|k) = 0, \qquad H_u \le h \le H_p - 1, \\ y_{i\_min}^s \le \hat{y}_i^s(k+l+1|k) \le y_{i\_max}^s, \\ u_{i\_min}^s \le \hat{u}_i^s(k+l|k) \le u_{i\_max}^s, \\ \Delta u_{i\_min}^s \le \Delta\hat{u}_i^s(k+n|k) \le \Delta u_{i\_max}^s, \\ \forall l = 0,\ldots,H_p-1, \quad \text{and} \quad \forall n = 0,\ldots,H_u-1. \end{cases} \tag{12}$$

The vector $\hat{y}_i^s$ is a subset of the plant predicted outputs ($\hat{y}_i^s \subset [\hat{y}_1,...,\hat{y}_m]$) and represents the predicted output variables of subsystem i, $\forall i = 1,...,N$. Similarly, the vector $\hat{u}_i^s$ is a subset of the plant predicted inputs ($\hat{u}_i^s \subset [\hat{u}_1,...,\hat{u}_r]$) and represents the predicted input variables of subsystem i, $\forall i = 1,...,N$. According to the proposed decomposition, the predicted change in input variables and predicted outputs can be arranged in vector form as $\Delta\hat{U}_i(k) = [\Delta\hat{u}_i^s(k|k)^\top,...,\Delta\hat{u}_i^s(k+H_u-1|k)^\top]^\top$ and $\hat{Y}_i(k+1) = [\hat{y}_i^s(k+1|k)^\top,...,\hat{y}_i^s(k+H_p|k)^\top]^\top$, respectively.

In (10) and (11), the matrix $\mathcal{S}_{ii}$ corresponds to the step-response coefficient matrix between $\Delta\hat{U}_i(k)$ and predicted output variables $\hat{Y}_i(k+1)$; while the matrix $\mathcal{S}_{ij}$ corresponds to the step-response coefficient matrix between the interacting variables $\Delta\hat{U}_j(k)$ and predicted output variables $\hat{Y}_i(k+1)$. The matrices $\mathcal{S}_{ii}$ and $\mathcal{S}_{ij}$ can be obtained by decomposing matrices (5) and (6) into N subsystems.

Finally, the variables $\hat{V}_i(k)$ represent the interacting or linking variables among the different subunits in the plant. The interacting variables account for the effect that inputs from unit j have on unit i, with $i \ne j$. In the objective function (9), $Z_i(k) = [\hat{Y}_i(k+1)^\top, \Delta\hat{U}_i(k)^\top, \hat{V}_i(k)^\top]^\top$ represents the vector of decision variables for subunit i; the matrix Θ_i is the coefficient matrix for the linking variables, which is constructed according to (10) and (11), and p is a price vector provided by the coordinator.

For simplicity, we re-write problem (9)-(12) as:

$$\min_{Z_i} \quad \mathcal{J}_i = \frac{1}{2}\Big(Z_i(k)^\top \Upsilon_i Z_i(k)\Big) + \Phi_i^\top Z_i(k) + p^\top\Theta_i Z_i(k) \tag{13}$$

subject to:

$$\begin{cases} B_i^{eq} Z_i(k) = b_i^{eq}, \\ B_i^{ineq} Z_i(k) \le b_i^{ineq} \qquad \forall i = 1,...,N. \end{cases} \tag{14}$$

The optimization problem (13)-(14) can be straightforwardly obtained by arranging (9)-(12) in a matrix form for the entire prediction and control horizons. The optimization problem described by (13)-(14) forms part of the price-driven decomposition coordination method. The price-driven decomposition-coordination method was discussed in Jose and Ungar (2000, 1998) to solve algebraic optimization problems such as resource allocation or auction problems. This method was successfully adapted and implemented in Cheng et al. (2007) to solve the MPC steady-state target calculation problem. In this work, we use the price-driven method to solve MPC dynamic calculation problems.

2.4 Coordination of CDMPC Controllers

In the previous section, a decomposition of the overall problem into N smaller subproblems was presented. A key step in the formulation of CDMPC controllers is to design an efficient coordination mechanism that ensures convergence of the distributed optimal solutions to the centralized optimum. In this section, we extend the results obtained in Cheng et al. (2007) for the MPC steady-state target calculation to the MPC dynamic calculation.

As discussed in Jose and Ungar (2000), a large-scale problem:

$$\max_{z_1,\ldots,z_n} \quad \sum_{i}^{n} f_i(z_i)$$

$$\text{s.t.} \quad \sum_{i}^{n} R_i(z_i) \le \bar{R}, \qquad z_i \in \Omega_i,$$

with $z_i \in \Re^{n_i}$ decision variables, R_i vector of shared resources, and vector $\bar{R}$ representing the availability of shared resources, can be decomposed into N subproblems:

$$\max_{z_i \in \Omega_i} f_i(z_i) - (p + qR_i(z_i))^\top R_i(z_i). \quad (15)$$

In (15), 'p' represents the price vector, and the variable q is a nonnegative scalar that could be assumed to be zero for quadratic programming problems. In this work, we assume $q = 0$.

It was shown in Jose and Ungar (2000) that, when the subproblems present concave objective functions and compact convex feasible sets, they can be successfully coordinated. Moreover, at equilibrium, the following condition is satisfied:

$$\Delta R(p,q) = \sum_i R_i(p,q) - \bar{R} \leq 0,$$

with $p^\top(\Delta R(p,q)) = 0$, and $p \geq 0$.

Coordination of subproblem (13)-(14) for $i : 1, ..., N$ to achieve the plant-wide optimal solution can be performed by using an efficient price-update technique, such as Newton's method. Based on Newton's method, the price vector can be updated as follows (Cheng et al. (2007)):

$$p^{[\kappa+1]} = p^{[\kappa]} - \alpha J^{-1} \Delta R^{[\kappa]}, \quad (16)$$

provided that the matrix J is invertible. In the price update mechanism (16), the superscripts '$[\kappa]$' and '$[\kappa+1]$' denote the iteration steps; α is the step size in Newton's method, $\Delta R^{[\kappa]} = \Delta R(p,q)$, and J can be calculated as:

$$J = \frac{d\Delta R^{[\kappa]}}{dp^{[\kappa]}} = \sum_i \frac{dR_i^{[\kappa]}}{dp^{[\kappa]}}. \quad (17)$$

For the problem formulation described by (13)-(14), the shared resources or linking constraints are defined as $R_i^{[\kappa]} = \Theta_i Z_i^{[\kappa]}$, with $Z_i^{[\kappa]}$ representing the decision variables at each iteration step 'κ'. Therefore, the Jacobian matrix J in (17) becomes:

$$J = \sum_i \frac{dR_i^{[\kappa]}}{dp^{[\kappa]}} = \sum_i \Theta_i \frac{dZ_i^{[\kappa]}}{dp^{[\kappa]}}. \quad (18)$$

The Jacobian matrix (18) requires information of the sensitivity matrix $dZ_i^{[\kappa]}/dp^{[\kappa]}$; that is, in order to efficiently adjust the price vector, the coordinator should be aware of how the price affects the decision variables $Z_i^{[\kappa]}$ at each iteration. A sensitivity analysis was proposed in Wolbert et al. (1994) for an algebraic optimization of a process flowsheet, and it was extended in Cheng et al. (2007) for the MPC steady-state target calculation. This approach can be followed to solve problem (18). By performing a sensitivity analysis, the matrix $dZ_i^{[\kappa]}/dp^{[\kappa]}$ can be calculated. This requires solving the following system of equations:

$$\Gamma_i \begin{bmatrix} \nabla_p Z_i(k) \\ \nabla_p \lambda_i \\ \nabla_{pA} \mu_i \\ \nabla_{pI} \sigma_i \end{bmatrix} = - \begin{bmatrix} \Theta_i^\top \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (19)$$

where

$$\Gamma_i = \begin{bmatrix} \Upsilon_i & B_i^{eq^\top} & {}_A B_i^{ineq^\top} & 0 \\ B_i^{eq} & 0 & 0 & 0 \\ {}_A B_i^{ineq} & 0 & 0 & 0 \\ {}_I B_i^{ineq} & 0 & 0 & I \end{bmatrix}, \quad (20)$$

assuming that Γ_i is full-rank. We refer the reader to Cheng et al. (2007) for a detailed derivation of equations (19) and (20).

2.5 Implementation of CDMPC Control Scheme

In the traditional MPC implementation, a control action sequence is determined at each sampling interval by optimizing an objective criterion over a finite-time horizon. Only the first control signal is applied to the process, while the rest of the calculated control inputs are discarded (Camacho and Bordons (1999); Maciejowski (2002)). At the next sampling interval, new process measurements are available and the optimization is repeated to calculate a new control action sequence.

In a CDMPC control platform, the coordinator imposes an extra step to the traditional MPC implementation. Before the control input is applied to the process, the control action calculated by each distributed MPC controller needs to converge to the optimal centralized control action. Convergence of the CDMPC solutions to the centralized performance can be achieved by allowing the coordinator to iteratively adjust the price vector, and therefore the optimal solution of each subsystem, according to the plant-wide objective.

Implementation of the CDMPC controllers is carried out according to the following steps:

(1) **Initialization:** The coordinator sets up an initial price vector $p^{[\kappa]}$ for the interacting variables ($\Theta_i Z_i$, $\forall i = 1, ..., N$) and sends that information to every subsystem.
(2) **Optimization performed by each subsystem:** Based on the price provided by the coordinator, each subsystem solves its own optimization problem (13)-(14) and calculates the resources $R_i^{[\kappa]} = \Theta_i Z_i^{[\kappa]}$; as well as $dZ_i^{[\kappa]}/dp^{[\kappa]}$, according to (19)-(20). This information is communicated back to the coordinator.
(3) **Price update:** The coordinator gathers the information from each subsystem; it calculates $\Delta R^{[\kappa]}$, and J given by (18). Then, the coordinator determines the step size α (with $0 < \alpha \leq 1$) and updates the price vector $p^{[\kappa]}$ as per (16). The new price vector is informed to each subsystem.
(4) **Iteration until convergence:** Steps (2)-(3) are repeated until convergence of the price-driven decomposition coordination algorithm. Convergence of the algorithm is achieved when $||\Delta R^{[\kappa]}|| \leq \epsilon$, where ϵ is a tolerance error.
(5) **Implementation of control action:** Once the decomposition-coordination algorithm converges, the control actions calculated for the first sampling interval are implemented in each subsystem and the optimization problem (steps (1)-(4)) is initiated again for the next receding horizon.

3. SIMULATION EXAMPLE

In this section, a case study is performed to illustrate the effectiveness of the proposed algorithm. We consider a fluid

catalytic cracking (FCC) process given in Grosdidier et al. (1993). A diagram of the FCC system is shown in Fig. 2.

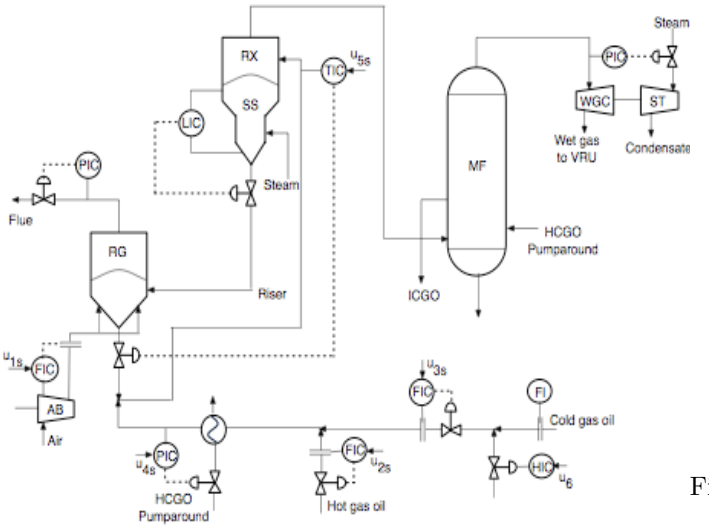

Fig. 2. FCC process (Grosdidier et al. (1993))

In the FCC unit, gas oil is converted into hydrocarbons of shorter chains. A description of the FCC process, together with the limit values for the controlled and manipulated variables are given in Grosdidier et al. (1993). The model of the FCC process, as well as, the models of the regulatory controllers are shown in tables 1 and 2, respectively. The continuous-time transfer function models were obtained through identification analysis of step-test data and include seven outputs and six inputs. The transfer function matrix for the overall process, including the models for the regulatory controllers, can be obtained by multiplying each transfer function model in table 1 by the corresponding input model in table 2, except for transfer functions between $y5-u5$ and $y6-u5$, which do not require that multiplication (Grosdidier et al. (1993)). To implement the CDMPC controllers, step-response models were obtained based on the process dynamics given in tables 1 and 2. The sampling interval used for simulations was 1 $[min]$.

3.1 Simulation Results

We begin by decomposing the centralized problem into two subsystems. The first subsystem includes outputs y_1 to y_3 and inputs u_1 to u_3, while the second subsystem includes outputs y_4 to y_7 and inputs u_4 to u_6. The following parameters were used in the simulation study: weighting matrices $Q(l+1) = diag\{5;10;0.001;5;5;5;0.001\}$, $R(n) = diag\{100;100;100;100;100;100\}$, for $l = 0,...,H_p-1$ and $n = 0,...,H_u-1$. The weighting matrices $Q(.)$ and $R(.)$ are decomposed as $Q_{11}(.) = diag\{5;10;0.001\}$ and $R_{11}(.) = diag\{100;100;100\}$ for the first subsystem, and $Q_{22}(.) = diag\{5;5;5;0.001\}$ and $R_{22}(.) = diag\{100;100;100\}$ for the second subsystem. The prediction horizon H_p and the control horizon H_u considered for the simulation are 50 and 5, respectively.

A set-point change of 0.5 was performed in output y_1 at initial time, while the targets for the remaining outputs were kept at the origin. The results of the simulation are

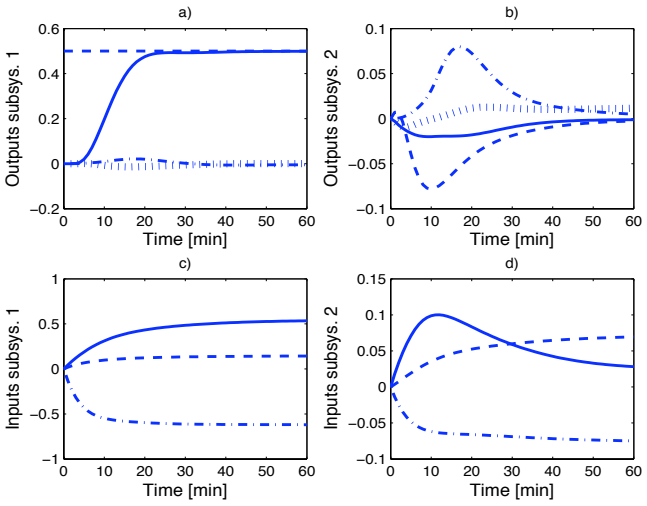

Fig. 3. a) Output variables for subsystem 1: set-point for y_1 (dashed line), y_1 (solid line), y_2 (dotted line), y_3 (dash-dot line); b) Output variables for subsystem 2: y_4(dash-dot line), y_5 (dashed line), y_6 (solid line), y_7 (dotted line); c) Input variables for subsystem 1:u_1 (dashed line), u_2 (dashed-dot line), u_3 (solid line); d) Input variables for subsystem 2: u_4 (dashed-dot line), u_5 (solid line), u_6 (dashed line)

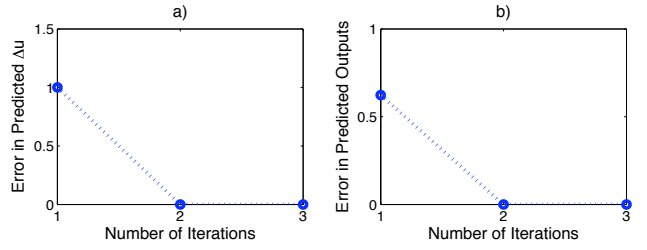

Fig. 4. a) Error in predicted change in input variables $\Delta\hat{U}(k)$; b) Error in predicted output variables $\hat{Y}(k+1)$

presented in Fig. (3)-(4). The closed-loop performance of the CDMPC controllers for subsystems 1 and 2 is shown in Fig. 3, where the trajectories are plotted in deviation variables. It can be seen in Fig. (3a)-(3d) that the CDMPC controllers provide a good performance since output y_1 achieves the new set-point and outputs y_2 to y_7 are stabilized at their new steady-state optimal values. Fig. (4a)-(4b) show the normalized errors of the predicted inputs ($||\Delta\hat{U}_{CDMPC} - \Delta\hat{U}_{cen}||/||\Delta\hat{U}_{cen}||$) and predicted outputs ($||\Delta\hat{Y}_{CDMPC} - \Delta\hat{Y}_{cen}||/||\Delta\hat{Y}_{cen}||$) for the optimization performed at the first sampling time. These prediction errors are calculated as the difference between the CDMPC optimal solutions and the optimal solutions calculated with a centralized MPC controller. It can be observed in Fig. (4a)-(4b) that the solutions achieved with the CDMPC controllers converge to the centralized performance within 2 iterations. The fast convergence observed in this simulation study confirms the results reported in Cheng et al. (2007) when using Newton's method as price-adjustment algorithm for the MPC steady-state target calculation. In the numerical simulations performed for the FCC unit, the same fast solution convergence (2 iterations) was observed within each control execution.

Table 1. FCC process models

	u_1	u_2	u_3	u_4	u_5	u_6
y_1	$\frac{0.097(1.7s+1)e^{-2s}}{19s^2+6.5s+1}$	$\frac{-0.87e^{-2s}}{13s^2+4.9s+1}$	$\frac{-0.092(0.25s+1)e^{-3s}}{3.7s^2+4.7s+1}$	$\frac{0.026e^{-7s}}{12s+1}$	$\frac{-0.074(4.8s+1)}{9.3s^2+3.4s+1}$	$\frac{-(0.48s)e^{-12s}}{(6s+1)(8s+1)}$
y_2	0	$\frac{0.55e^{-4s}}{27s^2+8.7s+1}$	$\frac{0.55e^{-4s}}{10s^2+4.9s+1}$	0	$\frac{0.74(1.7s+1)e^{-2s}}{11s^2+7.3s+1}$	$\frac{0.36e^{-11s}}{33s^2+6.5s+1}$
y_3	0	$\frac{0.14e^{-11s}}{46s^2+8.5s+1}$	$\frac{0.14e^{-6s}}{46s^2+8.5s+1}$	0	$\frac{0.27(16s+1)}{53s^2+23s+1}$	$\frac{0.015(12s+1)e^{-9s}}{66s^2+27s+1}$
y_4	0	$\frac{0.25e^{-11s}}{17s^2+7s+1}$	$\frac{0.25e^{-7s}}{3s+1}$	0	$\frac{0.70}{3s+1}$	$\frac{0.079(6.3s+1)e^{-10s}}{24s^2+12s+1}$
y_5	0	$\frac{0.66e^{-s}}{2.5s+1}$	$\frac{0.66e^{-s}}{2.5s+1}$	$\frac{-0.9e^{-10s}}{6s+1}$	$\frac{1}{2s+1}$	$\frac{-0.54e^{-11s}}{9s+1}$
y_6	0	$\frac{-0.84e^{-s}}{6.1s+1}$	$\frac{-0.90}{1.5s+1}$	$\frac{0.35e^{-10s}}{5s+1}$	$\frac{-(0.64s+1)}{13s^2+7s+1}$	$\frac{0.23(0.5s+1)e^{-14s}}{3.6s^2+11s+1}$
y_7	0	$\frac{0.81}{6s+1}$	$\frac{0.90}{s+1}$	$\frac{-0.35e^{-10s}}{5s+1}$	0.80	$\frac{-0.26e^{-18s}}{7.1s+1}$

Table 2. Models between regulatory controller set-points us_i and process inputs u_i, for $i = 1, ..., 6$

(us_1, u_1)	(us_2, u_2)	(us_3, u_3)	(us_4, u_4)	(us_5, u_5)	(us_6, u_6)
$\frac{1}{(0.75s+1)(4.5s+1)}$	$\frac{1}{(s+1)}$	$\frac{1}{1.7s^2+2.1s+1}$	$\frac{(3.3s+1)e^{-s}}{40s^2+13s+1}$	$\frac{(0.64s+1)}{13s^2+7s+1}$	1

Remark: For the ease of presentation, we decomposed the overall FCC process into two subsystems of similar dimensions. Nevertheless, the CDMPC control scheme can be applied to N number of subsystems of different dimensions. As future work, we will evaluate the efficiency of the CDMPC control scheme on process systems that include more subunits and there is a significant mismatch in the size of the subunits.

4. CONCLUSION

In this paper, we presented a coordinated-distributed model predicted control scheme for constrained dynamic processes. The CDMPC control framework improves the performance of decentralized controllers by achieving the overall plant-wide optimal operations.

An important advantage of CDMPC controllers is the simplicity in the control scheme, which does not require a radical new configuration of the decentralized MPC controllers operating in the plant. The upgrade from the existing decentralized MPC controllers to CDMPC controllers only involves small modifications in the control formulation of each subsystem and the addition of a coordinator.

The price-driven decomposition-coordination algorithm was used to efficiently coordinate the dynamic behavior of the CDMPC controllers. Newton's method was selected to update the price vector during the coordination process. It was shown with a benchmark process system that Newton's method provides a rapid convergence of the unit operations towards the plant-wide optimal performance.

REFERENCES

Camacho, E.F. and Bordons, C. (1999). *Model Predictive Control*. Springer, Berlin.

Cheng, R., Forbes, J.F., and Yip, W.S. (2007). Price-driven coordination method for solving plant-wide mpc problems. *Journal of Process Control*, 17, 429–438.

Cheng, R., Forbes, J.F., and Yip, W.S. (2008). Dantzig-wolfe decomposition and plant-wide mpc coordination. *Computers and Chemical Engineering*, 32, 1507–1522.

Dunbar, W.B. and Murray, R.M. (2004). Receding horizon control of multi-vehicle formations: A distributed implementation. In *43rd IEEE Conference on Decision and Control*, 1995–2002.

Grosdidier, P., Mason, A., Aitolahti, A., Heinonen, P., and Vanhamaki, V. (1993). Fcc unit reactor-regenerator control. *Computers and Chemical Engineering*, 17(2), 165–179.

Jose, R.A. and Ungar, L.H. (1998). Auction-driven coordination for plantwide optimization. In *Proceedings of Foundations of Computer-Aided Process Operation (FOCAPO)*.

Jose, R.A. and Ungar, L.H. (2000). Pricing interprocess streams using slack auctions. *AIChE Journal*, 46(3), 575–587.

Lu, J.Z. (2003). Challenging control problems and emerging technologies in enterprise optimization. *Control Engineering Practice*, 11, 847–858.

Maciejowski, J.M. (2002). *Predictive Control with Constraints*. Prentice Hall, Essex, England.

Sun, Y. and El-Farra, N.H. (2008). Quasi-decentralized model-based networked control of process systems. *Computers and Chemical Engineering*, 32, 2016–2029.

Venkat, A.N., Rawlings, J.B., and Wright, S.J. (2005). Stability and optimality of distributed model predictive control. In *Proceedings of the 44th IEEE Conference on Decision and Control*, 6680–6685.

Wolbert, D., Joulia, X., Koehret, B., and Biegler, L.T. (1994). Flowsheet optimization and optimal sensitivity analysis using analytical derivatives. *Computers and Chemical Engineering*, 18(11/12), 1083–1095.