Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 54-3 (2021) 257–262

Development of Non-cooperative Distributed Adaptive MPC Schemes based on ARX Models parameterized using Orthonormal Basis Filters

Sairengpuia Sailo * **Kunal Kumar** * **Sachin C. Patwardhan** ** **P.S.V. Nataraj** *

* *Systems and Control Engineering, Indian Institute of Technology Bombay, Mumbai-400076, India (e-mail: srpsailo7@gmail.com)*
** *Department of Chemical Engineering, Indian Institute of Technology Bombay, Mumbai-400076, India (e-mail: sachinp@iitb.ac.in)*

Abstract: Adaptive model predictive control (AMPC) schemes can prove beneficial in the control of non-linear time varying systems due to their ability to maintain closed-loop performance in the face of large transitions in operating conditions. In this work, two non-cooperative distributed adaptive MPC (dAMPC) schemes are proposed that are based on ARX models parameterized using generalized orthonormal basis filters (GOBF). Poles of the GOBF models estimated through offline estimation are kept fixed and only the Fourier coefficients are updated recursively online thereby avoiding drift in the pole locations. The system under consideration is decomposed into a number of subsystems coupled through the inputs. A separate local AMPC controller is implemented for each subsystem. Each local controller optimizes its input trajectories assuming the input trajectories of its neighbors fixed to the last computed values, and then shares the computed inputs with the neighbors. We consider both sequential and iterative distributed control strategies for the dAMPC development. The efficacy of the proposed dAMPC schemes is demonstrated using simulation studies on an octuple tank process. When the system is subjected to large transitions in operating points, the proposed dAMPC schemes generate closed loop performances comparable to a centralized adaptive MPC scheme while significantly reducing the average computation time.

Keywords: Model predictive control, distributed control, adaptive control, orthonormal basis filters, recursive least square estimation.

1. INTRODUCTION

Performance of a model predictive control (MPC) scheme based on linear black-box models can deteriorate if the operating condition changes significantly. A possible remedy to deal with this problem is to employ an adaptive MPC (AMPC) scheme wherein the linear model parameters are updated online. To reduce the computational burden on an online MPC optimization problem, it is usually transformed into a quadratic programming (QP) problem. AMPC scheme requires online parameter estimation and QP transformation using an updated model at every sampling instant. Thus, it demands more computational effort and time as compared to the non-adaptive MPC scheme wherein the matrices for QP formulation are initially constructed only once. This increment in computation time specially in case of large dimensional system can be a serious issue, although the AMPC scheme alleviates the limitation of the linear non-adaptive MPC scheme.

With the increasing complexity and dimension of industrial processes, a distributed MPC (dMPC) scheme has received a significant amount of attention in recent years. This is mainly due to its reduced computational complexity as a single large-scale optimization problem is decomposed into a number of distributed small-scale optimization problems. According to Christofides et al. (2013), dMPC algorithms are classified into two broad categories: (a) cooperative dMPC where each local controller optimizes a global cost function and (b) non-cooperative dMPC in which each local controller optimizes a local cost function.

In a non-cooperative dMPC scheme for two subsystems coupled through the inputs (Maxim et al. (2015)), each local controller solves its own local optimization problem based on the shifted previous optimal input trajectory of the neighbor. A non-cooperative dMPC scheme with iterative strategy is developed based on Nash optimality for control of large-scale systems in (Du et al. (2001)). Recently, Song et al. (2019) have developed a distributed adaptive MPC scheme for linear systems with unknown parameters. In this scheme, a cooperative distributed control algorithm is considered with a Lyapunov functional to solve the adaptive control problem. However, to the best of our knowledge, non-cooperative distributed adaptive MPC schemes based on black-box models have not been developed in the literature.

10.1016/j.ifacol.2021.08.251

In this work, we develop two non-cooperative distributed adaptive MPC (dAMPC) schemes based on ARX models parameterized using generalized orthonormal basis filters (GOBF). We decompose the overall system into a number of subsystems coupled through the inputs. Each subsystem is represented by GOBF-ARX local model. Fourier coefficients of each multi-input single-output (MISO) GOBF-ARX model in each subsystem are estimated online while keeping the GOBF pole locations fixed. Every subsystem is associated with a local adaptive MPC controller. Each local controller optimizes its own set of inputs while keeping its neighboring subsystem inputs fixed to the last computed values and then shares the computed inputs with the neighbors. We consider both sequential and iterative distributed control strategies for the dAMPC development. In a sequential strategy, each controller solves its own local optimization problem in sequence and only once per sampling period, whereas in an iterative strategy, all the controllers optimize their own local cost function simultaneously in parallel and iterative manner (Liu et al. (2010)). The efficacy of the proposed dAMPC schemes is demonstrated using simulation studies on an octuple tank system.

The paper is organized into five sections. The second section presents the control relevant model development and the parameter estimation schemes. Section 3 presents the details of the proposed dAMPC formulation. The results of the simulation study are presented in section 4. Finally, the main conclusions reached from the analysis of the simulation study are presented in the last section.

2. MODEL IDENTIFICATION

2.1 Development of GOBF-ARX model

In this text, superscript (i) is used to represent the i^{th} subsystem model, whereas superscript $[j]$ represents the j^{th} MISO model. Now, consider a $r \times m$ MIMO system which can be represented by r MISO models. The j^{th} MISO model can be expressed as follows

$$y_{j,k} = \sum_{i=1}^{m} \mathrm{G}_i^{[j]}(z,\theta)u_{i,k} + \mathrm{H}^{[j]}(z,\theta)e_{j,k} \tag{1}$$

where $j = 1, 2, ..., r$ and $\theta \in \mathbb{R}^p$ represents the model parameters. Both $\{\mathrm{G}_i^{[j]} : i = 1, 2, ..., m\}$ and $\mathrm{H}^{[j]}$ are strictly proper stable transfer functions, y represents the process output, u represents the process input and $\{e_{j,k}\}$ is a zero mean white noise sequence. For the purpose of parameter estimation, this model is rearranged in a one step predictor form as follows

$$\hat{y}_{j,k|k-1} = \sum_{i=1}^{m} \mathrm{W}_{u,i}^{[j]}(z,\theta)u_{i,k} + \mathrm{W}_y^{[j]}(z,\theta)y_{j,k} \tag{2}$$

$$y_{j,k} = \hat{y}_{j,k|k-1} + e_{j,k} \tag{3}$$

As suggested by Muddu et al. (2010), $\mathrm{W}_{u,i}^{[j]}(z,\theta)$ and $\mathrm{W}_y^{[j]}(z,\theta)$ can be parameterized using generalized orthonormal basis filters which are represented as

$$F_l(z) = \frac{\sqrt{\left(1 - \|\xi_l\|^2\right)}}{(z - \xi_l)} \prod_{i=1}^{l-1} \frac{(1 - \xi_i^* z)}{(z - \xi_i)} \tag{4}$$

where $\{\xi_l : l = 1, 2, ...\}$ represents an arbitrary sequence of poles inside the unit circle appearing in complex conjugate pairs. Thus, $\mathrm{W}_{u,i}^{[j]}(z,\theta)$ and $\mathrm{W}_y^{[j]}(z,\theta)$ are parameterized as follows

$$\mathrm{W}_{u,i}^{[j]}(z,\theta) = \sum_{l=1}^{n_{u,i}} \alpha_{i,l}^{[j]} F_{u,l}^{[j]}(z,\xi_{u,i}) \tag{5}$$

$$\mathrm{W}_y^{[j]}(z,\theta) = \sum_{l=1}^{n_y} \beta_l^{[j]} F_{y,l}^{[j]}(z,\xi_y) \tag{6}$$

where $n_{u,i}$ and n_y represent the number of OBF poles for $\mathrm{G}_i^{[j]}$ and $\mathrm{H}^{[j]}$ respectively. A state realization of the j^{th} MISO GOBF-ARX model is given by

$$x_{k+1}^{[j]} = \Psi^{[j]} x_k^{[j]} + \Gamma^{[j]} u_k + \mathrm{L}_p^{[j]} y_{j,k} \tag{7a}$$

$$y_{j,k} = \mathrm{C}^{[j]} x_k^{[j]} + e_{j,k} \tag{7b}$$

where $x_k^{[j]} \in \mathbb{R}^{n^{[j]}}$ represents the state vector, $u_k \in \mathbb{R}^m$ represents the input vector and $\{e_{j,k}\}$ represents the innovation sequence. $\mathrm{C}^{[j]}$ consists of the Fourier series expansion coefficients $\{\alpha_{i,l}^{[j]}\}$ and $\{\beta_l^{[j]}\}$ of (5) and (6) respectively. Defining the matrix $\Phi^{[j]}$ as

$$\Phi^{[j]} = \Psi^{[j]} + \mathrm{L}_p^{[j]} \mathrm{C}^{[j]} \tag{8}$$

the model (7) can be rearranged to get the standard innovation form of state space realization for the j^{th} MISO model as

$$x_{k+1}^{[j]} = \Phi^{[j]} x_k^{[j]} + \Gamma^{[j]} u_k + \mathrm{L}_p^{[j]} e_{j,k} \tag{9a}$$

$$y_{j,k} = \mathrm{C}^{[j]} x_k^{[j]} + e_{j,k} \tag{9b}$$

2.2 Development of subsystem-based model

To demonstrate the development of the subsystem-based local models with better intuitions, the overall system is decomposed into 2 subsystems $\{S_i : i = 1, 2\}$. Suppose the i^{th} subsystem consists of outputs $\{y_j, j = 1, 2, \dots r_i\}$. Then, r_i MISO models of the form (9) can be stacked appropriately to obtain the $r_i \times m$ MIMO model for the i^{th} subsystem as

$$x_{k+1}^{(i)} = \Phi^{(i)} x_k^{(i)} + \Gamma^{(i)} u_k + \mathrm{L}_p^{(i)} e_k^{(i)} \tag{10a}$$

$$y_k^{(i)} = \mathrm{C}^{(i)} x_k^{(i)} + e_k^{(i)} \tag{10b}$$

The stacked vectors are defined as

$$x_k^{(i)} = \left[\left(x_k^{[1]}\right)^T \left(x_k^{[2]}\right)^T \dots \left(x_k^{[r_i]}\right)^T \right]^T$$

$$y_k^{(i)} = \left[y_{1,k} \; y_{2,k} \; \dots \; y_{r_i,k} \right]^T$$

$$e_k^{(i)} = \left[e_{1,k} \; e_{2,k} \; \dots \; e_{r_i,k} \right]^T$$

Similarly, the model matrices are given by

$$\Phi^{(i)} = \text{block diagonal} \left[\Phi^{[1]} \; \Phi^{[2]} \; \dots \; \Phi^{[r_i]} \right]$$

$$\mathrm{L}_p^{(i)} = \text{block diagonal} \left[\mathrm{L}_p^{[1]} \; \mathrm{L}_p^{[2]} \; \dots \; \mathrm{L}_p^{[r_i]} \right]$$

$$\mathrm{C}^{(i)} = \text{block diagonal} \left[\mathrm{C}^{[1]} \; \mathrm{C}^{[2]} \; \dots \; \mathrm{C}^{[r_i]} \right]$$

$$\Gamma^{(i)} = \left[\left(\Gamma^{[1]}\right)^T \left(\Gamma^{[2]}\right)^T \dots \left(\Gamma^{[r_i]}\right)^T \right]^T$$

We can see from the model equations (10), that the subsystems are coupled through the inputs, but not through the states. Since, there are m inputs, the input matrix $\Gamma^{(i)}$ has m number of columns, and thus can be rewritten as

$$\Gamma^{(i)} = \begin{bmatrix} \Gamma_1^{(i)} & \Gamma_2^{(i)} & \dots & \Gamma_m^{(i)} \end{bmatrix} \tag{11}$$

Clearly, $\Gamma_t^{(i)} \in \mathbb{R}^{n^{(i)} \times 1}$, where $t = 1, 2, ..., m$ and $n^{(i)}$ is the number of states in the i^{th} subsystem model. Now, suppose out of the m inputs, the first m_1 inputs $\{u_l : l = 1, 2, ..., m_1\}$ are the manipulated inputs of S_1 and the remaining $(m - m_1)$ inputs are the manipulated inputs of S_2. By further decomposing the input matrix $\Gamma^{(i)}$ of the i^{th} subsystem in (10a) into two matrices $\Gamma_{um}^{(i)}$ and $\Gamma_{ud}^{(i)}$, the model for the i^{th} subsystem can be rewritten as

$$x_{k+1}^{(i)} = \Phi^{(i)} x_k^{(i)} + \Gamma_{um}^{(i)} u_{m,k}^{(i)} + \Gamma_{ud}^{(i)} u_{d,k}^{(i)} + \mathrm{L}_p^{(i)} e_k^{(i)} \tag{12a}$$

$$y_k^{(i)} = \mathrm{C}^{(i)} x_k^{(i)} + e_k^{(i)} \tag{12b}$$

where $u_{m,k}^{(i)}$ is the manipulated input vector of the i^{th} subsystem, and $u_{d,k}^{(i)}$ is the vector consisting of the inputs from the other subsystems which can be treated as a disturbance input of the i^{th} subsystem. Then, we have

$$\Gamma_{um}^{(1)} = \begin{bmatrix} \Gamma_1^{(1)} & \dots & \Gamma_{m_1}^{(1)} \end{bmatrix}; \quad \Gamma_{ud}^{(1)} = \begin{bmatrix} \Gamma_{m_1+1}^{(1)} & \dots & \Gamma_m^{(1)} \end{bmatrix}$$

$$\Gamma_{um}^{(2)} = \begin{bmatrix} \Gamma_{m_1+1}^{(2)} & \dots & \Gamma_m^{(2)} \end{bmatrix}; \quad \Gamma_{ud}^{(2)} = \begin{bmatrix} \Gamma_1^{(2)} & \dots & \Gamma_{m_1}^{(2)} \end{bmatrix}$$

2.3 Offline and Online Parameter Estimation

To begin with, multiple MISO GOBF-ARX models are developed using a batch of data generated by injecting simultaneous perturbations into the system. This offline estimation of model parameters is carried out using the nested optimization approach developed in (Muddu et al. (2010)). In this work, we consider a model with *fixed poles* for the development of the adaptive control scheme. Hence, once the OBF poles are estimated from offline identification exercise, the matrices Φ, Γ and L_p in (9) are treated as fixed. To adapt to the changing operating conditions, we modify the measurement equation of the MISO model in each of the subsystem as follows

$$y_{j,k} = \mu_{j,k} + \mathrm{C}_k^{[j]} x_k^{[j]} + e_{j,k} \tag{13}$$

The bias term $\mu_{j,k}$ is introduced to take care of the change in the operating points as well as the mean shift in any additive disturbance that may affect the plant outputs. The parameters $\mu_{j,k}$ and $\mathrm{C}_k^{[j]}$ in (13) are updated online using a Recursive Least Squares (RLS) estimation algorithm. Due to better numerical stability, we consider the square root form of the RLS algorithm (Söderström and Stoica (1988)).

3. DISTRIBUTED AMPC

3.1 dAMPC formulation

In this section, we formulate the dAMPC scheme based on the GOBF-ARX models identified online. At the k^{th} sampling instant, we have the model equations for the i^{th} subsystem as follows

$$x_{k+1}^{(i)} = \Phi^{(i)} x_k^{(i)} + \Gamma_{um}^{(i)} u_{m,k}^{(i)} + \Gamma_{ud}^{(i)} u_{d,k}^{(i)} + \mathrm{L}_p^{(i)} e_k^{(i)} \tag{14a}$$

$$y_k^{(i)} = \hat{\mu}_k^{(i)} + \hat{\mathrm{C}}_k^{(i)} x_k^{(i)} + e_k^{(i)} \tag{14b}$$

Note that, in the proposed adaptive formulation, the system matrices appearing in (14a) remain unchanged while $\hat{\mu}_k^{(i)}$ and $\hat{\mathrm{C}}_k^{(i)}$ in (14b) are time varying and are updated using the RLS algorithm. Thus, given a guess of the future inputs, the predictions over the future time window $[k+1, k+N_p]$, where N_p represents the prediction horizon, can be generated with the initial condition $\hat{x}_{k|k}^{(i)} = x_k^{(i)}$ as follows

$$\hat{x}_{k+l|k}^{(i)} = \Phi^{(i)} \hat{x}_{k+l-1|k}^{(i)} + \Gamma_{um}^{(i)} u_{m,k+l-1|k}^{(i)} + \Gamma_{ud}^{(i)} u_{d,k+l-1|k}^{(i)} + \mathrm{L}_p^{(i)} e_{f,k}^{(i)} \tag{15a}$$

$$\hat{y}_{k+l|k}^{(i)} = \hat{\mu}_k^{(i)} + \hat{\mathrm{C}}_k^{(i)} \hat{x}_{k+l|k}^{(i)} + e_{f,k}^{(i)} \tag{15b}$$

$$e_{f,k}^{(i)} = \alpha e_{f,k-1}^{(i)} + (1 - \alpha) e_k^{(i)} \tag{15c}$$

$$e_k^{(i)} = y_k^{(i)} - \left[\hat{\mu}_k^{(i)} + \hat{\mathrm{C}}_k^{(i)} x_k^{(i)} \right] \tag{15d}$$

where $l = 1, 2, ..., N_p$, $\alpha \in [0, 1)$ is a tuning parameter and $e_{f,k}^{(i)}$ represent the innovation filtered through a unity gain low pass filter to suppress the effect of the high frequency noise on the future predictions.

The local cost function for the i^{th} subsystem is given as follows

$$J_k^{(i)} = \sum_{l=1}^{N_p - 1} \left\| E_{k+l}^{(i)} \right\|_{W_E^{(i)}}^2 + \left\| x_{k+N_p|k}^{(i)} - x_{s,k}^{(i)} \right\|_{w_{\infty,k}^{(i)}}^2 + \sum_{l=0}^{N_c - 1} \left\| \delta u_{m,k+l|k}^{(i)} \right\|_{W_u^{(i)}}^2 + \sum_{l=0}^{N_c - 1} \left\| \Delta u_{m,k+l|k}^{(i)} \right\|_{W_{\Delta u}^{(i)}}^2 \tag{16}$$

where

$$E_{k+l}^{(i)} = s_k^{(i)} - \hat{y}_{k+l|k}^{(i)}$$

$$\delta u_{m,k+l|k} = u_{m,k+l|k}^{(i)} - u_{sm,k}^{(i)}$$

$$\Delta u_{m,k+l|k}^{(i)} = u_{m,k+l|k}^{(i)} - u_{m,k+l-1|k}^{(i)}$$

$$u_{m,k-1|k}^{(i)} = u_{m,k-1}^{(i)}$$

$W_E^{(i)} \succ 0$, $W_u^{(i)} \succcurlyeq 0$, $W_{\Delta u}^{(i)} \succcurlyeq 0$ are the weighting matrices and N_c represents the control horizon. The terminal state weighting matrix $w_{\infty,k}^{(i)}$ is evaluated at every sampling instant by solving a discrete Lyapunov equation given by

$$w_{\infty,k}^{(i)} = \left(\hat{\mathrm{C}}_k^{(i)} \right)^T W_E^{(i)} \hat{\mathrm{C}}_k^{(i)} + \left(\Phi^{(i)} \right)^T w_{\infty,k}^{(i)} \Phi^{(i)} \tag{17}$$

When the number of controlled outputs is equal to or more than the number of manipulated inputs, the target input $u_{sm,k}^{(i)}$ and the target state $x_{s,k}^{(i)}$ can be computed by solving the following optimization problem

$$\min_{u_{sm,k}^{(i)}} \left(s_k^{(i)} - y_{s,k}^{(i)} \right)^T W_E^{(i)} \left(s_k^{(i)} - y_{s,k}^{(i)} \right) \tag{18a}$$

subject to

$$\left(\mathrm{I} - \Phi^{(i)} \right) x_{s,k}^{(i)} = \Gamma_{um}^{(i)} u_{sm,k}^{(i)} + \Gamma_{ud}^{(i)} u_{sd,k}^{(i)} + \mathrm{L}_p^{(i)} e_{f,k}^{(i)} \tag{18b}$$

$$y_{s,k}^{(i)} = \hat{\mu}_k^{(i)} + \hat{\mathrm{C}}_k^{(i)} x_{s,k}^{(i)} + e_{f,k}^{(i)} \tag{18c}$$

$$u_{min}^{(i)} \le u_{sm,k}^{(i)} \le u_{max}^{(i)} \tag{18d}$$

where $s_k^{(i)}$ is the setpoint, $u_{sd,k}^{(i)}$ is the vector consisting of the target inputs of the neighboring subsystems. Then, the optimization problem (OP) for the i^{th} subsystem which is solved at every sampling instant is given as follows

$$\min_{u^{(i)}_{m,[k,k+N_c-1]}} \quad J^{(i)}_k \tag{19a}$$

subject to

$$\hat{x}^{(i)}_{k+l|k} = \Phi^{(i)}\hat{x}^{(i)}_{k+l-1|k} + \Gamma^{(i)}_{um}u^{(i)}_{m,k+l-1|k} + \Gamma^{(i)}_{ud}u^{(i)}_{d,k+l-1|k} + \mathrm{L}^{(i)}_p e^{(i)}_{f,k} \tag{19b}$$

$$\hat{y}^{(i)}_{k+l|k} = \hat{\mu}^{(i)}_k + \hat{\mathrm{C}}^{(i)}_k\hat{x}^{(i)}_{k+l|k} + e^{(i)}_{f,k} \tag{19c}$$

$$u^{(i)}_{min} \le u^{(i)}_{m,k+l|k} \le u^{(i)}_{max} \tag{19d}$$

$$\Delta u^{(i)}_{min} \le \Delta u^{(i)}_{m,k+l|k} \le \Delta u^{(i)}_{max} \tag{19e}$$

$$y^{(i)}_{min} \le \hat{y}^{(i)}_{k+l|k} \le y^{(i)}_{max} \tag{19f}$$

The optimization problem (19) is solved using QP solver. Then, only the first input move is applied to the plant i.e., $u^{(i)}_k = \left(u^{(i)}_{m,k|k}\right)^*$.

3.2 dAMPC algorithm

Based on the dMPC implementation strategies presented in (Christofides et al. (2013), Liu et al. (2010), Du et al. (2001)), the algorithms for implementation of the proposed non-cooperative dAMPC schemes in sequential and iterative configurations with 'N' numbers of subsystems are presented in Algorithms 1 and 2 respectively. The algorithms are employed at every sampling instant 'k'. It is to be noted that while solving the local optimization problem (19) by the i^{th} local controller, the inputs from the neighboring subsystems i.e., $u^{(i)}_{d,k}$ are assumed to be fixed to the last computed values. When the current sampling instant optimal input trajectories of the neighboring subsystems are not available, $u^{(i)}_{d,k}$ assumes the shifted optimal input trajectories of the previous sampling instant. In the algorithms, u_k denotes the input to the overall system, while $u^{(i)}_k$ and $u^{(i)}_{s,k}$ are the manipulated input and the target input of the i^{th} subsystem. In the iterative dAMPC algorithm, the maximum number of iterations (c_{max}) is considered as the termination condition.

Algorithm 1 Sequential dAMPC

1: **for** $i = 1, 2 \ldots, N$ **do**
2: acquire measurement $y^{(i)}_k$
3: using (14) update $x^{(i)}_k$ using u_{k-1} and $x^{(i)}_{k-1}$
4: update $\hat{\mu}^{(i)}_k$ and $\hat{\mathrm{C}}^{(i)}_k$ using $y^{(i)}_k$ and $x^{(i)}_k$
5: using (18) obtain $u^{(i)}_{s,k}, x^{(i)}_{s,k}$ with assumption that $\forall j = 1, 2, \ldots N, j \neq i, u^{(j)}_{s,k} = u^{(j)}_{s,k-1}$
6: $\left(u^{(i)}_k\right)^* = \text{argmin}\left(J^{(i)}_k\right)$ (OP given by (19))
7: send $\left(u^{(j)}_k\right)^* \forall j = 1, 2, \ldots i, i \neq N$ to $(i+1)^{th}$ subsystem
8: **end for**

4. SIMULATION STUDY

The simulation study for demonstrating the efficacy of the proposed dAMPC schemes is conducted on an octuple tank system. The octuple tank system used for the simulation study is the modified version of the system presented by Maxim et al. (2019). It is obtained by linking the two benchmark quadruple tanks presented by Johansson (2000) in a series as shown in Figure 1. As the name indicates the setup consists of eight interacting tanks and four pumps. The input voltages $v_1(= u_1)$, $v_2(= u_2)$, $v_3(= u_3)$ and $v_4(= u_4)$ supplied to the respective pumps are the manipulated inputs of the system. The controlled outputs are the four bottom tank levels $h_1(= y_1)$, $h_2(= y_2)$, $h_3(= y_3)$ and $h_4(= y_4)$ measured in term of volts using the level measurement devices.

Algorithm 2 Iterative dAMPC

1: **for** $c = 1, 2, \ldots, c_{max}$ **do**
2: **for** $i = 1, 2 \ldots, N$ **do**
3: **if** $c = 1$ **then**
4: acquire measurement $y^{(i)}_k$
5: using (14) update $x^{(i)}_k$ using u_{k-1} and $x^{(i)}_{k-1}$
6: update $\hat{\mu}^{(i)}_k$ and $\hat{\mathrm{C}}^{(i)}_k$ using $y^{(i)}_k$ and $x^{(i)}_k$
7: using (18) obtain $u^{(i)}_{s,k}, x^{(i)}_{s,k}$ with assumption that $\forall j \neq i, j = 1, 2 \ldots N, u^{(j)}_{s,k} = u^{(j)}_{s,k-1}$
8: **end if**
9: $\left(u^{(i)}_k\right)^* = \text{argmin}\left(J^{(i)}_k\right)$ (OP given by (19))
10: **end for**
11: **for** $i = 1, 2 \ldots, N$ **do**
12: send $\left(u^{(i)}_k\right)^*$ to each neighboring subsystem $j = 1, 2, \ldots, N, j \neq i$
13: **end for**
14: **end for**

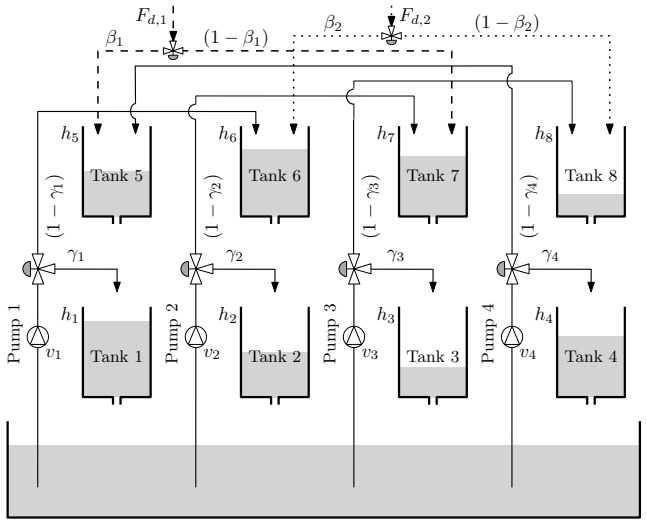

Fig. 1. Schematic diagram of octuple tank system

We introduce two unmeasured disturbance inlet flows $F_{d,1}$ and $F_{d,2}$ with valve settings β_1 and β_2 respectively. As per the redefined process, the system dynamics becomes

$$\frac{dh_1}{dt} = -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_5}{A_1}\sqrt{2gh_5} + \frac{\gamma_1 k_1}{A_1}v_1 \tag{20a}$$

$$\frac{dh_2}{dt} = -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_6}{A_2}\sqrt{2gh_6} + \frac{\gamma_2 k_2}{A_2}v_2 \tag{20b}$$

$$\frac{dh_3}{dt} = -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{a_7}{A_3}\sqrt{2gh_7} + \frac{\gamma_3 k_3}{A_3}v_3 \tag{20c}$$

$$\frac{dh_4}{dt} = -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{a_8}{A_4}\sqrt{2gh_8} + \frac{\gamma_4 k_4}{A_4}v_4 \quad (20d)$$

$$\frac{dh_5}{dt} = -\frac{a_5}{A_5}\sqrt{2gh_5} + \frac{(1-\gamma_4)k_4}{A_5}v_4 + \frac{\beta_1}{A_5}F_{d,1} \quad (20e)$$

$$\frac{dh_6}{dt} = -\frac{a_6}{A_6}\sqrt{2gh_6} + \frac{(1-\gamma_1)k_1}{A_6}v_1 + \frac{\beta_2}{A_6}F_{d,2} \quad (20f)$$

$$\frac{dh_7}{dt} = -\frac{a_7}{A_7}\sqrt{2gh_7} + \frac{(1-\gamma_2)k_2}{A_7}v_2 + \frac{(1-\beta_1)}{A_7}F_{d,1} \quad (20g)$$

$$\frac{dh_8}{dt} = -\frac{a_8}{A_8}\sqrt{2gh_8} + \frac{(1-\gamma_3)k_3}{A_8}v_3 + \frac{(1-\beta_2)}{A_8}F_{d,2} \quad (20h)$$

The system parameter values used in the simulation study and the chosen operating points are presented in Tables 1 and 2 respectively.

Table 1. Parameters

Parameters	Values
A_1, A_3, A_5, A_7	$28\,\mathrm{cm}^2$
A_2, A_4, A_6, A_8	$32\,\mathrm{cm}^2$
a_1, a_3, a_5, a_7	$0.071\,\mathrm{cm}^2$
a_2, a_4, a_6, a_8	$0.057\,\mathrm{cm}^2$
k_c	$0.50\,\mathrm{V/cm}$
g	$981\,\mathrm{cm/s}^2$

Table 2. Operating points

Parameters	Nominal values
$h_1^0, h_2^0, h_3^0, h_4^0$	$15.59, 16.09, 13.64, 15.45\,\mathrm{cm}$
$h_5^0, h_6^0, h_7^0, h_8^0$	$2.97, 2.63, 2.65, 3.03\,\mathrm{cm}$
$v_1^0, v_2^0, v_3^0, v_4^0$	$3.00, 3.00, 3.00, 3.00\,\mathrm{V}$
$F_{d,1}^0, F_{d,2}^0$	$2.00, 2.00\,\mathrm{cm}^3/\mathrm{s}$
$y_1^0, y_2^0, y_3^0, y_4^0$	$7.79, 8.04, 6.82, 7.72\,\mathrm{V}$
k_1, k_2, k_3, k_4	$3.33, 3.35, 3.33, 3.35\,\mathrm{cm}^2/\mathrm{V/s}$
$\gamma_1, \gamma_2, \gamma_3, \gamma_4$	$0.70, 0.60, 0.65, 0.55$
β_1, β_2	$0.45, 0.55$

The control objective is to control the water levels in the four bottom tanks under the influence of unmeasured disturbances. The considered system is decomposed into two subsystems, $S_1 : \{y_1, y_2, u_1, u_2\}$ and $S_2 : \{y_3, y_4, u_3, u_4\}$. Since each subsystem has 2 inputs and 2 outputs, each local controller is a 2×2 AMPC controller. The following performance indices are used to facilitate the comparison of the different control schemes.

- *Performance index (J)*

$$\mathrm{J} = \frac{1}{N_r}\sum_{i=1}^{N_r}\left(\frac{1}{N_s}\sum_{k=1}^{N_s}\left\{\|s_k - y_k\|^2_{W_E} + \|u_k - u_{k-1}\|^2_{W_{\Delta u}}\right\}\right)^{\{i\}} \quad (21)$$

  Here, N_s and N_r are the numbers of samples and noise realization respectively in the simulation study, s_k and y_k are the setpoint and output vectors of the overall system respectively and u_k is the manipulated input to the overall system at the sampling instant k. The superscript $\{i\}$ refers to the i^{th} noise realization.
- *Computation time (CT)*
  The average computation time per sampling instant in second.

The given system is perturbed in open loop by simultaneously introducing low frequency (between 0 to 0.5 rad/s) random binary sequence (RBS) in the inputs to the plant. The state and measurement noises are both assumed to be zero mean Gaussian with covariance matrix equal to $0.25^2 \times \mathbf{I}$. Then, the resulting input-output data is used to identify four MISO GOBF-ARX models with 2 poles each with respect to its inputs and 1 pole with respect to the associated output.

The closed-loop performance of the proposed dAMPC schemes under multiple setpoint changes is investigated. The same tuning parameters are used for both the proposed control schemes and are reported in Table 3. The simulation study has been carried out using MATLAB 2020a on a computer with Core i7 processor, 3.6 GHz and 16 GB RAM.

Table 3. Controller tuning parameters

Parameters	**Values**
Prediction Horizon	100
Control Horizon	5
Input blocking	[5 15 20 30 30]
Error weighting matrix	[1 1 1 1]
Input weighting matrix	[1 1 1 1]
Δu weighting matrix	[0.1 0.1 0.1 0.1]
Input constraints	0 to 20 V
Output constraints	0 to 20 V
Input rate constraints	-1 to 1
Filter coefficients	0.75
Sampling interval	5 s

Figures 2 and 3 present the comparison of the closed-loop performances of the different control schemes such as centralized (non-adaptive) MPC (cMPC), centralized adaptive MPC (cAMPC), distributed adaptive MPC in sequential configuration (dAMPCs) and distributed adaptive MPC in iterative configuration (dAMPCi) with the distributed controllers in non-cooperative strategy. Large setpoint changes in the order of about (60-150)% are introduced at the 50^{th} and 250^{th} sampling instances. It can be observed from these plots that the proposed dAMPC schemes in both sequential and iterative configurations produce similar performance when compared with the cAMPC scheme.

Variations in the model parameters are captured by the sensitivity matrix defined as

$$\mathbf{G}_k = \hat{\mathbf{C}}_k \left[\mathbf{I} - \Phi\right]^{-1} \Gamma \quad (22)$$

Parameter variations are shown in Figure 5. In this plot, G_{jj} represents the gain between *output* j and *input* j. We can observe significant changes in the model parameters in response to the changes in the operating points.

Table 4 compares the performances of the different control schemes with respect to the performance indices based on 20 stochastic simulation trials. Multiple noise realizations are considered in order to make our analysis independent of a specific noise realization. Mean value and standard deviation of each performance index are reported in the table. Note that, for dAMPCi scheme, the indices are reported after the first two iterations as relative change in the performance index after the first two iterations were not significant and waiting till the convergence was

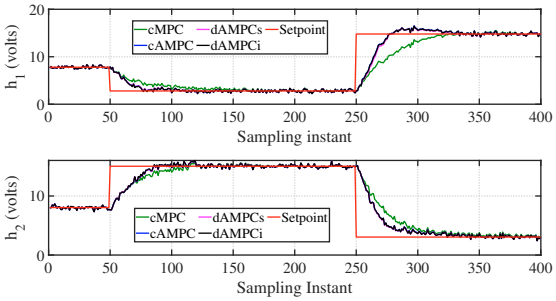

Fig. 2. Comparison of controlled outputs: levels 1 and 2

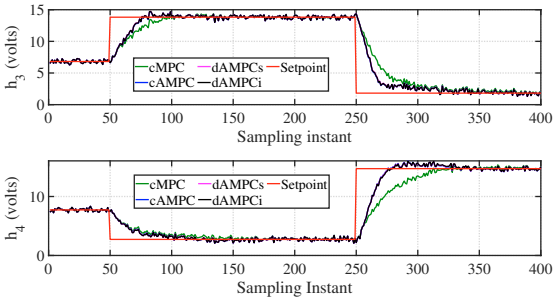

Fig. 3. Comparison of controlled outputs: levels 3 and 4

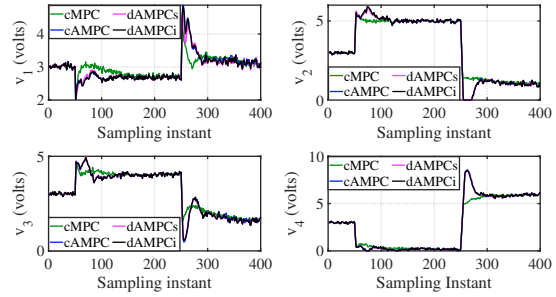

Fig. 4. Comparison of manipulated inputs

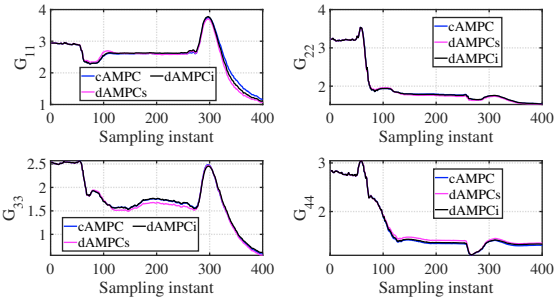

Fig. 5. Comparison of sensitivity variations

achieved only increased the computation time. With respect to J, the proposed dAMPC schemes along with the cAMPC scheme result in a significantly improved performance in response to the setpoint changes when compared with the performance of the cMPC scheme. As compared to the cAMPC scheme, the proposed dAMPC schemes result in a comparable performance with respect to J. The cMPC scheme requires lesser computation time than the cAMPC scheme. However, it is evident that the proposed dAMPC schemes especially the iterative dAMPC scheme are computationally more efficient than the cAMPC scheme.

Table 4. Comparison of performance indices

Parameters	J	CT(s)
cMPC	25.7690 (0.1184)	0.0783 (0.59×10^{-3})
cAMPC	18.8819 (0.1640)	0.1048 (0.93×10^{-3})
dAMPCs	18.8898 (0.1665)	0.0753 (0.74×10^{-3})
dAMPCi	18.8341 (0.1877)	0.0530* (0.76×10^{-3})

*The mean time spent by each AMPC controller is reported.

5. CONCLUSION

In this work, two non-cooperative distributed adaptive MPC (dAMPC) schemes are proposed that are based on ARX models parameterized using generalized orthonormal basis filters (GOBF). We have considered both sequential and iterative distributed control strategies for the dAMPC development. The efficacy of the proposed dAMPC schemes is demonstrated using simulation studies on an octuple tank process. An analysis of the simulation results reveals that the proposed dAMPC schemes outperform the centralized adaptive MPC scheme in terms of average computation time with comparable closed-loop performance.

REFERENCES

Christofides, P.D., Scattolini, R., Muñoz de la Peña, D., and Liu, J. (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering*, 51, 21 – 41. CPC VIII.

Du, X., Xi, Y., and Li, S. (2001). Distributed model predictive control for large-scale systems. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, volume 4, 3142–3143.

Johansson, K.H. (2000). The quadruple-tank process: A multivariable laboratory process with an adjustable zero. *IEEE Transactions on Control Systems Technology*, 8(3), 456–465.

Liu, J., Chen, X., Muñoz de la Peña, D., and Christofides, P.D. (2010). Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal*, 56(8), 2137–2149.

Maxim, A., Caruntu, C.F., Lazar, C., De Keyser, R., and Ionescu, C.M. (2019). Comparative analysis of distributed model predictive control strategies. In *2019 23rd International Conference on System Theory, Control and Computing (ICSTCC)*, 468–473.

Maxim, A., Lazar, C., and Caruntu, C.F. (2015). A computationally efficient non-cooperative distributed model predictive control algorithm for two agent systems. In *2015 20th International Conference on Control Systems and Computer Science*, 673–678.

Muddu, M., Narang, A., and Patwardhan, S.C. (2010). Reparametrized ARX models for predictive control of staged and packed bed distillation columns. *Control Engineering Practice*, 18(2), 114–130.

Söderström, T. and Stoica, P. (1988). *System Identification*. Prentice-Hall, Inc., USA.

Song, Y., Zhu, K., Wei, G., and Wang, J. (2019). Distributed MPC-based adaptive control for linear systems with unknown parameters. *Journal of the Franklin Institute*, 356(5), 2606 – 2624.