Distributed model predictive control and estimation of large-scale multi-rate systems

S. Roshany-Yamchi, R.R. Negenborn, M. Cychowski, B. De Schutter, J. Connell, K. Delaney

Delft University of Technology, Delft, The Netherlands

Distributed Model Predictive Control and Estimation of Large-Scale Multi-Rate Systems ⋆

Samira Roshany-Yamchi * **Marcin Cychowski** *
Kieran Delaney * **Rudy R. Negenborn** ** **Bart De Schutter** **

* *NIMBUS Centre, Cork Institute of Technology, Cork, Ireland,*
(e-mail: {samira.roshany,marcin.cychowski,kieran.delaney}@cit.ie).
** *Delft Center for Systems and Control,*
Delft University of Technology, Delft, The Netherlands,
(e-mail: {r.r.negenborn,b.deschutter}@tudelft.nl).

Abstract: In this paper, we propose a new method for control of large-scale multi-rate systems with linear dynamics that are coupled via inputs. These systems are multi-rate systems in the sense that either output measurements or input updates are not available at certain sampling times. Such systems can arise, e.g., when the number of sensors is less than the number of variables to be controlled, or when measurements of outputs cannot be completed simultaneously because of applicational limitations. The multi-rate nature gives rise to lack of information, which will cause uncertainty in the system's performance. We propose to control such systems with a distributed model predictive control (MPC) approach based on Nash game theory, in which multiple control agents each determine actions for their own parts of the system. Via communication, the agents can in a cooperative way take one another's actions into account. To compensate for the information loss due to the multi-rate nature of the systems under study, a distributed Kalman Filter is proposed to provide the optimal estimation of the missing information. Using simulation studies on a distillation column the added value of the proposed distributed MPC and Kalman Filter method is illustrated in comparison with a centralized method, and a distributed MPC with fully decentralized (i.e., no communication) Kalman Filter.

1. INTRODUCTION

Economical and technological demands motivate the development of large-scale plants such as process plants, manufacturing systems, and satellite orbit formations with low complexity and high performance accuracy. Large-scale systems consist of many subsystems that interact. Control of such systems in a centralized setting needs high computational effort. Model Predictive Control (MPC) is a well-established method that can handle constraints and is relatively easy to tune. However, centralized MPC is impractical for controlling large-scale systems including several interacting subsystems and requires significant computation. Many distributed MPC methods have been developed by researchers to cope with large-scale control problem. Examples include the work by Al-Gherwi et al. (2010); Li et al. (2005); Mercangoz and Doyle (2007); Negenborn et al. (2008). Architectures for distributed MPC have been extensively reviewed by Scattolini (2009).

In this paper, we propose a novel control method for multi-rate sampled linear systems that uses distributed MPC. In multi-rate plants, either the measurements are available less frequently or the control moves are made at a lower rate. This kind of systems can be seen in many industrial applications. Scattolini and Schiavoni (1995); Embiruu and Fontes (2006); Gopinath et al. (1994); Ohshima et al. (1994) illustrate practical situations in which measurements of process variables and input updates occur at different rates. E.g., they illustrate that in the process industry even when the quality variables, such as, e.g., a product concentration or the average molecular weight distribution of a polymerization process, are measured online, time delays involved in the measurements are significantly large when compared to other process measurements. On the other hand in some biomedical applications (Gopinath et al., 1994) the input injection rate is slower than the output measurement, e.g., in some drug infusion systems, the drug injection to the patient happens less frequently than the body symptom's measurements such as blood pressure, body temperature, etc.

Our focus is to develop a generalized scheme that covers both of these aspects for large-scale systems. Lee et al. (1992); Scattolini and Schiavoni (1995) developed a state space based multi-rate MPC scheme in a centralized case. In a distributed MPC architecture, the whole system is decomposed into a number of small subsystems. Each subsystem is controlled by a so-called *agent*, which solves

⋆ This research is supported by the Irish Programme for Research in Third Level Institutions (Cycle 4) (funded under the National Development Plan 2007-2013 with assistance from the European Regional Development Fund), the BSIK project "Next Generation Infrastructures (NGI)", the Delft Research Center Next Generation Infrastructures, the European STREP project "Hierarchical and distributed model predictive control (HD-MPC)", and the VENI project "Intelligent multi-agent control for flexible operation of water infrastructures" (project 11210) of the Dutch Technology Foundation STW.

its own, local optimization problem. We propose a new MPC control strategy for large-scale systems with *multi-rateness* in its subsystems. This means that each of the subsystems is multi-rate in inputs and/or outputs. The multi-rate control method that we have developed allows control moves to be made using state estimates from a distributed Kalman Filter. The efficiency of this method is illustrated by comparing it to other MPC schemes for the high purity distillation column model developed by Skogestad et al. (1988).

2. STATE-SPACE REPRESENTATION

Consider the distributed MPC of plants with linear dynamics whose centralized nominal model is decomposed into m subsystems. Let t be the discrete-time index for the system under control. The following augmented model can be derived for a distributed system with input coupling:

$$\mathbf{x}_i(t+1) = \mathbf{A}_i\mathbf{x}_i(t) + \mathbf{B}_{ii}\Delta\mathbf{u}_i(t) + \mathbf{D}_i\mathbf{v}_i(t) + \sum_{\substack{j=1\\ j\neq i}}^{m}\mathbf{B}_{ij}\Delta\mathbf{u}_j(t) \qquad (1)$$

$$\mathbf{y}_i(t) = \mathbf{C}_i\mathbf{x}_i(t) + \mathbf{z}_i(t), \qquad (2)$$

where for each subsystem i, $\mathbf{x}_i$, $\mathbf{u}_i$, $\mathbf{y}_i$ denote the state, input and output variables, respectively, $\mathbf{v}_i$, $\mathbf{z}_i$ are the state and measurement noise sequences, respectively and $\mathbf{A}_i$, $\mathbf{B}_{ii}$, $\mathbf{B}_{ij}$, $\mathbf{D}_i$ and $\mathbf{C}_i$ are matrices of appropriate dimensions. Assume that the sampling instants for the system vary as $t = 1, 2, \ldots, T_f$, where T_f is the final sampling time. We follow a strategy similar to Scattolini and Schiavoni (1995) to implement multi-rate measurement or input updating mechanisms for subsystem i. In a multi-rate output setting, the output vector $\mathbf{y}_i(t)$ of subsystem i can be measured every T_{y_i} units, where $T_{y_i} > 0$. Define the switching function γ_{i_j}, for $j = 1, 2, \ldots, q$ with q being the number of outputs, as follows:

$$\gamma_{i_j}(t) = \begin{cases} 1 & \text{if } t = \tau_{y_j}T_{y_j} \\ 0 & \text{otherwise,} \end{cases} \qquad (3)$$

where τ_{y_i} is an integer introduced to describe the output sampling mechanism for subsystem i and $\gamma_{i_j}(t)$ represents output sampling parameter of subsystem i for j number of outputs $j = 1, 2, \ldots, q_i$. The following measured output vector $\boldsymbol{\varphi}_i(t)$ can now be defined:

$$\boldsymbol{\varphi}_i(t) = \boldsymbol{\Upsilon}_i(t)\mathbf{y}_i(t), \qquad (4)$$

where

$$\boldsymbol{\Upsilon}_i(t) = \mathrm{diag}[\gamma_{i_1}(t)\ \ \gamma_{i_2}(t)\ \ldots\ \gamma_{i_q}(t)]. \qquad (5)$$

In a multi-rate input setting, the input vector $\mathbf{u}_i(t)$ of subsystem i is updated every T_{u_i} units, where $T_{u_i} > 0$. Introduce a switching function μ_{i_j} for $j = 1, 2, \ldots, l$ with l being the number of inputs. Define the inputs holding mechanism as:

$$\mu_{i_j}(t) = \begin{cases} 1 & \text{if } t = \tau_{u_j}T_{u_j} \\ 0 & \text{otherwise,} \end{cases} \qquad (6)$$

where τ_{u_i} is an integer introduced to describe the input updating mechanism and $\mu_{i_j}(t)$ represents input sampling parameter of subsystem i for j number of inputs $j = 1, 2, \ldots, l_i$. The following input matrix $\boldsymbol{\Psi}_i(t)$ for subsystem i can be defined.

$$\boldsymbol{\Psi}_i(t) = \mathrm{diag}\left[\mu_{i_1}(t)\ \ \mu_{i_2}(t)\ \ldots\ \mu_{i_l}(t)\right]. \qquad (7)$$

Now a new control variable $\boldsymbol{\vartheta}_i(t)$ is introduced to implement the input administering mechanism:

$$\Delta\mathbf{u}_i(t) = \boldsymbol{\Psi}_i(t)\boldsymbol{\vartheta}_i(t). \qquad (8)$$

In fact the control input computed by the local MPC is $\boldsymbol{\vartheta}_i(t)$ and not $\Delta u_i(t)$, however in the multi-rate system the manipulated variables are defined as (8) which includes both the computed inputs and the input updating variable $\boldsymbol{\Psi}_i(t)$.

After substituting (8) into (1) we get:

$$\mathbf{x}_i(t+1) = \mathbf{A}_i\mathbf{x}_i(t) + \mathbf{B}_{ii}\boldsymbol{\Psi}_i(t)\boldsymbol{\vartheta}_i(t) + \mathbf{D}_i\mathbf{v}_i(t) + \sum_{\substack{j=1\\ j\neq i}}^{m}\mathbf{B}_{ij}\boldsymbol{\Psi}_j(t)\boldsymbol{\vartheta}_j(t) \qquad (9)$$

$$(10)$$

As in multi-rate systems output measurements are done at specific sampling times, therefore the output sampling mechanism needs to be included in the system's model. To do that both sides of (2) are multiplied by the output sampling parameter $\boldsymbol{\Upsilon}_i(t)$, we get:

$$\boldsymbol{\Upsilon}_i(t)\mathbf{y}_i(t) = \boldsymbol{\Upsilon}_i(t)\mathbf{C}_i\mathbf{x}_i(t) + \boldsymbol{\Upsilon}_i(t)\mathbf{z}_i(t). \qquad (11)$$

The left hand side of (11) can be replaced by (4), therefore:

$$\boldsymbol{\varphi}_i(t) = \boldsymbol{\Upsilon}_i(t)\mathbf{C}_i\mathbf{x}_i(t) + \boldsymbol{\Upsilon}_i(t)\mathbf{z}_i(t). \qquad (12)$$

Equations (9) and (12) give the linear state-space representation of the distributed multi-rate system for $i = 1, 2, \ldots, m$ with m the number of subsystems. Next, the distributed MPC problem will be formulated for such a system.

3. CONTROL PROBLEM FORMULATION

3.1 Nash-based Distributed MPC

In the distributed control structure, input coupling among subsystems is considered as given by (9). These subsystems communicate with one another to accomplish a global objective (see Fig. 1). One type of Distributed MPC based on Nash optimality has been investigated by Li et al. (2005); Al-Gherwi et al. (2010). In this approach, the agents communicate but they do not take a cooperative decision. The agents iterate to resolve their local optimization problem simultaneously and obtain their optimal solution (Li et al., 2005; Giovanini and Balderud, 2006). An initial guess for each agent is first given based on the solution found at the last sampling time. Then each agent checks if its terminal iteration condition satisfies a user-defined threshold. This implies that the agents do not share information about the utility of each decision; the agreement (Nash equilibrium) between the agents is reached when neither of their solutions can be improved. The main advantage of this scheme is that the on-line optimization of a large-scale problem can be converted into several small-scale subproblems, thus reducing the computational complexity significantly while keeping satisfactory performance.

3.2 Computation

Consider a linear system consisting of m subsystems and m control agents. In Nash-based distributed MPC each

Fig. 1. Distributed control and estimation architecture.

control agent calculates the $\boldsymbol{\vartheta}_i(t)$ manipulated variable action by minimizing its local cost function as follows:

$$\min_{\boldsymbol{\vartheta}_i(t),\ldots,\boldsymbol{\vartheta}_i(t+N_c-1)} J_i(t) = \sum_{k=1}^{N_p} \|\mathbf{y}_i(t+k) - \mathbf{y}_i^0(t+k)\|^2_{\mathbf{Q}_i} + \sum_{k=0}^{N_c-1} \|\boldsymbol{\vartheta}_i(t+k)\|^2_{\mathbf{R}_i}, \quad (13)$$

$$\text{subject to } \boldsymbol{\vartheta}_{i,\min} \leq \boldsymbol{\vartheta}_i(t) \leq \boldsymbol{\vartheta}_{i,\max}, \quad (14)$$

where $\boldsymbol{\vartheta}_{i,\min}$ and $\boldsymbol{\vartheta}_{i,\max}$ are the lower and upper limits for the inputs, respectively.$\mathbf{Q}_i \geq 0$ and $\mathbf{R}_i > 0$ denote the weighting matrices and N_p and N_c are the prediction and control horizons, respectively. In order to solve the problem in (13)-(14), first we substitute (9) into (12). Based on the augmented state-space model (9)-(12), the future state variables are calculated sequentially using the set of future control parameters. The matrices thus obtained can be written in a compact form as

$$\mathbf{Y}_i(t) = \mathbf{F}_i\mathbf{x}_i(t) + \boldsymbol{\phi}_{ii}(t)\boldsymbol{\theta}_i(t) + \boldsymbol{\Gamma}_i\boldsymbol{\zeta}_i(t) + \sum_{\substack{j=1\\ j\neq i}}^{m} \boldsymbol{\phi}_{ij}(t)\boldsymbol{\theta}_j(t), \quad (15)$$

with

$$\begin{aligned} \mathbf{Y}_i(t) &= [\mathbf{y}_i(t+1)\ \ \mathbf{y}_i(t+2),\ldots,\mathbf{y}_i(t+N_p)]^T, \\ \boldsymbol{\theta}_i(t) &= [\boldsymbol{\vartheta}_i(t)\ \ \boldsymbol{\vartheta}_i(t+1),\ldots,\boldsymbol{\vartheta}_i(t+N_c-1)]^T, \\ \boldsymbol{\zeta}_i(t) &= [\mathbf{v}_i(t)\ \ \mathbf{v}_i(t+1),\ldots,\mathbf{v}_i(t+N_c-1)]^T, \\ \mathbf{F}_i &= [\mathbf{C}_i\mathbf{A}_i\ \ \mathbf{C}_i\mathbf{A}_i^2,\ldots,\mathbf{C}_i\mathbf{A}_i^{N_p}]^T, \end{aligned} \quad (16)$$

$$\boldsymbol{\phi}_{ij}(t) = \begin{bmatrix} \mathbf{C}_i\mathbf{B}_{ij}\boldsymbol{\Psi}_j(t) & 0 & \ldots & 0 \\ \mathbf{C}_i\mathbf{A}_i\mathbf{B}_{ij}\boldsymbol{\Psi}_j(t) & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & 0 \\ \mathbf{C}_i\mathbf{A}_i^{N_p-1}\mathbf{B}_{ij}\boldsymbol{\Psi}_j(t) & \ldots & \ldots & \mathbf{C}_i\mathbf{A}_i^{N_p-N_c}\mathbf{B}_{ij}\boldsymbol{\Psi}_j(t+N_c-1) \end{bmatrix} \quad (17)$$

$$\boldsymbol{\Gamma}_i = \begin{bmatrix} \mathbf{C}_i\mathbf{D}_i & 0 & \ldots & 0 \\ \mathbf{C}_i\mathbf{A}_i\mathbf{D}_i & \mathbf{C}_i\mathbf{D}_i & \ddots & \vdots \\ \vdots & \vdots & \vdots & 0 \\ \mathbf{C}_i\mathbf{A}_i^{N_p-1}\mathbf{D}_i & \mathbf{C}_i\mathbf{A}_i^{N_p-2}\mathbf{D}_i & \ldots & \mathbf{C}_i\mathbf{A}_i^{N_p-N_c}\mathbf{D}_i \end{bmatrix}. \quad (18)$$

Note that in (15), $\boldsymbol{\theta}_i(t)$ is used as the control input and the input sampling variable $\boldsymbol{\Psi}_j(t)$ is embedded in (17). In practice, the current state $\mathbf{x}_i(t)$ is usually not available from measurements and a state observer needs to be used to reconstruct the full state vector. In this case, we replace $\mathbf{x}_i(t)$ by its estimate $\widehat{\mathbf{x}}_i(t)$, hence

$$\widehat{\mathbf{Y}}_i(t) = \mathbf{F}_i\widehat{\mathbf{x}}_i(t) + \boldsymbol{\phi}_{ii}(t)\boldsymbol{\theta}_i(t) + \boldsymbol{\Gamma}_i\boldsymbol{\zeta}_i(t) + \sum_{\substack{j=1\\ j\neq i}}^{m} \boldsymbol{\phi}_{ij}(t)\boldsymbol{\theta}_j(t). \quad (19)$$

If $\mathbf{Y}_i^0(t) = [\mathbf{y}_i^0(t+1)\ \mathbf{y}_i^0(t+2),\ldots,\mathbf{y}_i^0(t+N_p)]^T$, the local optimization problem (13) for agent i can be reformulated as:

$$\min_{\boldsymbol{\theta}_i(t)} J_i(t) = \|\widehat{\mathbf{Y}}_i(t) - \mathbf{Y}_i^0(t)\|^2_{\mathbf{Q}_i} + \|\boldsymbol{\theta}_i(t)\|^2_{\mathbf{R}_i}, \quad (20)$$

$$\text{subject to } \boldsymbol{\theta}_{i,\min} \leq \boldsymbol{\theta}_i(t) \leq \boldsymbol{\theta}_{i,\max}, \quad (21)$$

where $\boldsymbol{\theta}_{i,\min}$ and $\boldsymbol{\theta}_{i,\max}$ are the lower and upper bounds for the inputs, respectively. It can be shown that problem (20)–(21) is equivalent to a quadratic programming problem which can be solved efficiently and reliably using standard off-the-shelf solvers. Notice that although the global objective function can be split into several local objective functions, the output of each subsystem is still related to all the input variables due to the input coupling.

The Nash-based MPC algorithm proceeds by allowing each subsystem/agent to optimise its objective function using its own control decision $\boldsymbol{\vartheta}_i(t)$ assuming that other subsystem's solutions $\boldsymbol{\vartheta}_j(t)$ are known. Let $\boldsymbol{\vartheta}_i^n(t)$ define the computed control input for subsystem i at iteration n, $(n \geq 0)$. At sampling time t, in the first iteration $(n = 0)$, each agent makes initial estimation of the input variables $\boldsymbol{\vartheta}_i^n(t)$ and announces it to the other agents. Then each agent solves its optimization problem (20)–(21) simultaneously and gets its optimal solution $\boldsymbol{\vartheta}_i^{n+1}(t)$. Next, all the agent compares the new solution with that obtained in the previous iteration and checks the convergence condition:

$$\|\boldsymbol{\vartheta}_i^{n+1}(t) - \boldsymbol{\vartheta}_i^n(t)\| \leq \boldsymbol{\epsilon}_i, \quad (22)$$

in which $\boldsymbol{\epsilon}_i$, (for $i = 1, 2, \ldots, m$) is the error accuracy If the Nash optimal solution is achieved, each subsystem does not change its decision $\boldsymbol{\vartheta}_i^n(t)$ because it has achieved an equilibrium point of the coupling decision process (Giovanini and Balderud, 2006); otherwise the local cost function $J_i(t)$ will degrade.

In the following section, a novel distributed Kalman Filter algorithm is proposed to provide optimal estimation $\widehat{\mathbf{x}}_i(t)$ of the state vector $\mathbf{x}_i(t)$ while compensating for the inter-sampling information loss due to the multi-rate nature of the systems under study.

4. DISTRIBUTED KALMAN FILTER

Consider the linear model in (9)–(12). We want to use the available measurements $\mathbf{y}$ to estimate the state of the system $\mathbf{x}$. To understand the distributed Kalman Filter equations, let us consider the process noise $\mathbf{v}_i(t)$ to be a discrete-time white noise for each subsystem i. The following covariance matrix for each agent can hence be defined:

$$\mathbb{E}\{\boldsymbol{v}_i(t)\boldsymbol{v}_i^T(t)\} = \mathbf{S}_{p_i}(t) \quad (23)$$

where $\mathbb{E}[\cdot]$ denotes the expectation of the argument, $\mathbf{S}_{p_i}(t)$ represents the covariance matrix of the process noise. Consider the measurement noise $\mathbf{z}_i(t)$ in (12) to be a discrete-time white noise. The following covariance matrix for the measurement noise $\mathbf{S}_{m_i}(t)$ can be similarly defined:

$$\mathbb{E}\{\boldsymbol{z}_i(t)\boldsymbol{z}_i^T(t)\} = \mathbf{S}_{m_i}(t). \quad (24)$$

Let the estimated states by the distributed Kalman Filter for a multi-rate system be given by:

$$\begin{aligned} \widehat{\mathbf{x}}_i(t+1|t) = {} & \mathbf{A}_i\widehat{\mathbf{x}}_i(t|t-1) + \mathbf{B}_{ii}\Delta\mathbf{u}_i(t) \\ & + \mathbf{L}_i(t)[\boldsymbol{\varphi}_i(t) - \boldsymbol{\Upsilon}_i(t)\mathbf{C}_i\widehat{\mathbf{x}}_i(t|t-1)] \\ & + \sum_{\substack{j=1\\ j\neq i}}^{m} [\mathbf{B}_{ij}\Delta\mathbf{u}_j(t) \end{aligned}$$

$$+ \mathbf{L}_j(t)[\boldsymbol{\varphi}_j(t) - \boldsymbol{\Upsilon}_j(t)\mathbf{C}_j\widehat{\mathbf{x}}_j(t|t-1)]\big], \tag{25}$$

where the terms $\mathbf{L}_i(t)$ and $\mathbf{L}_j(t)$ are referred as the *Kalman Gains*. From (25) it is clear that local estimators share their gains and also estimated states to accomplish their estimation task. Substituting (12) into (25) and combining with (9) we proceed to the next step to obtain the estimation error $\mathbf{e}_i(t+1|t) = \mathbf{x}_i(t+1|t) - \widehat{\mathbf{x}}_i(t+1|t)$ at sampling time t. The index $(t|t-1)$ refers to the information at sampling time t given knowledge of the process prior to sampling time t. Therefore,

$$\begin{aligned}\mathbf{e}_i(t+1|t) &= \mathbf{x}_i(t+1|t) - \widehat{\mathbf{x}}_i(t+1|t)\\ &= [\mathbf{A}_i - \mathbf{L}_i(t)\boldsymbol{\Upsilon}_i(t)\mathbf{C}_i]\mathbf{e}_i(t|t-1)\\ &\quad + \mathbf{D}_i\mathbf{v}_i(t) - \mathbf{L}_i(t)\boldsymbol{\Upsilon}_i(t)\mathbf{z}_i(t)\\ &\quad - \sum_{\substack{j=1\\ j\neq i}}^{m}\big(\mathbf{L}_j(t)\boldsymbol{\Upsilon}_j(t)\mathbf{C}_j\mathbf{e}_j(t|t-1)\\ &\quad + \mathbf{L}_j(t)\boldsymbol{\Upsilon}_j(t)\mathbf{z}_j(t)\big).\end{aligned} \tag{26}$$

Now to initialize the estimator algorithm, consider $\mathbb{E}[\mathbf{x}_i(0|-1)] = \widehat{\mathbf{x}}_i(0|-1)$ then $\mathbb{E}[\mathbf{e}_i(t|t-1)] = \mathbf{0}$, $\forall t$. This means we assume that the mean of the estimates should be equal to the mean of the expected value in Kalman Filter design. In order to develop the Kalman Filter for the multi-rate and distributed case we define a covariance matrix $\mathbf{S}_i(t)$ where,

$$\mathbf{S}_i(t+1) = \mathbb{E}\{\mathbf{e}_i(t+1|t)\mathbf{e}_i^T(t+1|t)\}. \tag{27}$$

By the properties of the vector covariance and expansion of the terms (Lee et al., 1992) we obtain the final form of the multi-rate distributed Kalman Filter as:

$$\begin{aligned}\mathbf{S}_i(t+1) &= \mathbf{A}_i\mathbf{S}_i(t)\mathbf{A}_i^T + \mathbf{D}_i\,\mathbf{S}_{p_i}(t)\,\mathbf{D}_i^T\\ &\quad - \mathbf{A}_i\mathbf{S}_i(t)\mathbf{C}_i^T\boldsymbol{\Upsilon}_i(t)\boldsymbol{\Omega}_i^{-1}(t)\boldsymbol{\Upsilon}_i(t)\mathbf{C}_i\mathbf{S}_i(t)\mathbf{A}_i^T\\ &\quad + \sum_{\substack{j=1\\ j\neq i}}^{m}\mathbf{A}_j\mathbf{S}_i(t)\mathbf{C}_j^T\boldsymbol{\Upsilon}_j(t)\boldsymbol{\Omega}_j^{-1}(t)\,\boldsymbol{\Upsilon}_j(t)\mathbf{C}_j\mathbf{S}_j(t)\mathbf{A}_j^T,\end{aligned} \tag{28}$$

with $\boldsymbol{\Omega}_i(t)$ and $\boldsymbol{\Omega}_j(t)$ positive definite and defined as:

$$\begin{aligned}\boldsymbol{\Omega}_i &= \boldsymbol{\Upsilon}_i(t)\mathbf{C}_i\mathbf{S}_i(t)\mathbf{C}_i^T\boldsymbol{\Upsilon}_i(t) + \boldsymbol{\Upsilon}_i(t)\mathbf{S}_{m_i}(t)\boldsymbol{\Upsilon}_i(t)\\ &\quad + [\mathbf{I}_{q\times q} - \boldsymbol{\Upsilon}_i(t)]\end{aligned} \tag{29}$$

$$\begin{aligned}\boldsymbol{\Omega}_j &= \boldsymbol{\Upsilon}_j(t)\mathbf{C}_j\mathbf{S}_j(t)\mathbf{C}_j^T\boldsymbol{\Upsilon}_j(t) + \boldsymbol{\Upsilon}_j(t)\mathbf{S}_{m_j}(t)\boldsymbol{\Upsilon}_j(t)\\ &\quad + [\mathbf{I}_{q\times q} - \boldsymbol{\Upsilon}_j(t)].\end{aligned} \tag{30}$$

It should be noted that (28) is the algebraic Ricatti equation. The solution of the Ricatti equation is found iteratively backwards in time by using (29) and (30). Then, the Kalman gains are computed as:

$$\mathbf{L}_i(t) = \mathbf{A}_i\mathbf{S}_i(t)\mathbf{C}_i^T\boldsymbol{\Upsilon}_i(t)\boldsymbol{\Omega}_i^{-1}(t), \tag{31}$$

$$\mathbf{L}_j(t) = \mathbf{A}_j\mathbf{S}_i(t)\mathbf{C}_j^T\boldsymbol{\Upsilon}_j(t)\boldsymbol{\Omega}_j^{-1}(t). \tag{32}$$

In order to guarantee the non-singularity of $\boldsymbol{\Omega}_i$ and $\boldsymbol{\Omega}_j$ at any time instant the extra terms $[\mathbf{I}_{q\times q} - \boldsymbol{\Upsilon}_i(t)]$ and $[\mathbf{I}_{q\times q} - \boldsymbol{\Upsilon}_j(t)]$ have been added to (29) and (30), respectively, in which $\mathbf{I}_{q\times q}$ is the identity matrix of size q-by-q, (Scattolini and Schiavoni, 1995). The matrix $[\mathbf{I}_{q\times q} - \boldsymbol{\Upsilon}(t)]$ only adds non-zero terms to the scalar diagonal elements of $\boldsymbol{\Omega}_i(t)$ and $\boldsymbol{\Omega}_j(t)$ during the output sampling mechanism and in no way affects the predictor equation.

5. SIMULATION RESULTS

To illustrate the efficiency of the proposed method for solving distributed control problems, the example of the high purity distillation column studied in Skogestad et al. (1988); Al-Gherwi et al. (2010) is considered. There, the outputs of the system, y_1 and y_2 are the top and bottom product compositions, respectively, and the inputs u_1 and u_2 are the reflux flowrate and the boil-up, respectively. As the composition dynamics are usually much slower than the flow dynamics the system can be considered as a multi-rate system. In particular, the following nominal state-space continuous-time model is considered:

$$\begin{aligned}\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} &= \begin{bmatrix}-0.0133 & 0\\ 0 & -0.0133\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}\\ &\quad + \begin{bmatrix}0.0117 & 0.0115\\ 0.0144 & 0.0146\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix},\end{aligned} \tag{33}$$

$$\boldsymbol{y} = \boldsymbol{x}. \tag{34}$$

The inputs are constrained to $-2 \leq u_1 \leq 2$ and $-1 \leq u_2 \leq 2$. The continuous-time model is discretized with a sampling time of 1 min. The parameters used for simulation purposes are the same as the parameters used by Al-Gherwi et al. (2010): $N_p = 20$, $N_c = 5$ and $\mathbf{Q} = \mathbf{R} = \mathbf{I}$. The set-point value for the first subsystem switches between one and zero every 200 minutes and for the second subsystem the set-point is zero. The nominal model was decomposed into two subsystems as follows:

Subsystem 1:

$$\dot{x}_1 = -0.0133x_1 + 0.0117u_1 + 0.0115u_2 \tag{35}$$

$$y_1 = x_1 \tag{36}$$

Subsystem 2:

$$\dot{x}_2 = -0.0133x_2 + 0.0146u_1 + 0.0144u_2 \tag{37}$$

$$y_2 = x_2. \tag{38}$$

The process and measurement noises for both subsystems are zero mean white noise sequences with covariances $\mathbf{S}_{p_1}(t) = \mathbf{S}_{p_2}(t) = \mathbf{S}_{m_1}(t) = \mathbf{S}_{m_2}(t) = 10^{-5}$. The closed-loop performance was studied using centralized MPC, Nash-based MPC with Distributed Kalman Filter, and Nash-based MPC with decentralized Kalman Filter for a multi-rate setting. The outputs and inputs for the asynchronous agents (agents in which both inputs sampling rate and outputs sampling rate are different) are depicted in Fig. 2. It can be seen from Fig. 2 that the proposed method in the presence of noise and input constraints represents good tracking performance which is close to the centralized case. Moreover, the distributed Nash-based MPC with a decentralized Kalman Filter shows slower tracking in comparison with the centralized approach and the proposed method. In Fig. 3 and Fig. 4, two scenarios have been considered. In the first scenario the goal is to analyze the degradation of the performance due to an infrequent output sampling. From the simulation results in Fig. 3 it can be seen that performance does not degrade significantly even in a noisy process with lack of output measurements. In the second scenario the aim is to analyze the effect of infrequent inputs on the system performance. The simulation results for the second scenario are depicted in Fig. 4 which shows satisfactory performance in the presence of noise. From Fig. 4 it can be seen that the (a)(b)(c)(d)

Fig. 2. Closed-loop response of asynchronous agents using Centralized approach in red (solid); Nash-based MPC with DKF (*proposed method*) in blue (dashed); Nash-based MPC with decentralized KF in green (dash-dotted).

(a)(b)

Fig. 3. Closed-loop response of different output sampling rates using the proposed method. The solid line represents $T_{y_2} = 3$; the dashed line represents $T_{y_2} = 6$; the dash-dotted line represents $T_{y_2} = 9$.

(a)(b)

Fig. 4. Closed-loop response of different input sampling using the proposed method. The solid line represents $T_{u_1} = 2$; the dashed line represents $T_{u_1} = 3$; the dash-dotted line represents $T_{u_1} = 4$.

Table 1. Simulation results and analysis of: Centralized (Cent), Nash-based MPC with distributed Kalman Filter (Nash-DiKF) and Nash-based MPC with decentralized Kalman Filter (Nash-DeKF).

Performance Evaluation - Cost Computation			
	Cent	**Nash-DiKF**	**Nash-DeKF**
Single-Rate	$T_{u_1} = 1$; $T_{u_2} = 1$; $T_{y_1} = 1$; $T_{y_2} = 1$		
Constrained	0.3752	0.5808	1.4025
Unconstrained	0.3063	0.5082	0.7756
Multi-Rate1	$T_{u_1} = 2$; $T_{u_2} = 1$; $T_{y_1} = 4$; $T_{y_2} = 9$		
Constrained	0.4955	0.6125	1.0464
Unconstrained	0.4332	0.5449	0.7956
Multi-Rate2	$T_{u_1} = 1$; $T_{u_2} = 1$; $T_{y_1} = 3$; $T_{y_2} = 9$		
Constrained	0.3373	0.5713	0.9070
Unconstrained	0.2713	0.5193	0.7666
Multi-Rate3	$T_{u_1} = 2$; $T_{u_2} = 4$; $T_{y_1} = 1$; $T_{y_2} = 1$		
Constrained	0.7570	0.6224	1.5659
Unconstrained	0.7022	0.5900	0.8435

proposed method presents even better performance than the centralized case for the second subsystem's output when the input sampling period increases. The main reason for this improvement with respect to the centralized scheme is that in the centralized case when the inputs are updated less frequently than the output measurements, the state computation at each sampling time can not be accomplished properly because of the missing input injection in the inter-sampling times. However this problem is not the case in the proposed method as the Kalman Filter corrects this fault by replacing the computed states with the estimated states. Table 1 gives the comparison of the costs computed by (20) as a mean over the complete simulation period. A comparison has been done between the three MPC schemes. The results of the analysis for the set-point tracking show better performance for the proposed method in comparison with the decentralized Kalman Filter case.

6. CONCLUSIONS AND FUTURE RESEARCH

Distributed MPC via a Nash game has been studied for multi-rate sampled-data systems. A distributed Kalman Filter has been proposed to provide the state values for inter-sampling times. The proposed method has been compared with a centralized scheme and also with a decentralized Kalman Filter scheme through simulations. The simulation results illustrate the efficiency of our method. The simulation has been done in MATLAB for possible cases including slow and fast input samples as well as slow and fast output samples. The proposed scheme offers efficient tracking for constrained problems including both process and measurement noise. Future research and theoretical contributions are required toward developing efficient algorithms with guaranteed properties, such as stability, robustness, convergence, and performance.

REFERENCES

W. Al-Gherwi, H. Budman, and A. Elkamel. Selection of control structures for distributed model predictive control in the presence of model errors. *Process Control*, 20(3):270–284, March 2010.

M. Embiruu and C. Fontes. Multirate multivariable generalized predictive control and its application to a slurry reactor for ethylene polymerization. *Chemical Engineering Science*, 61(17):5754–5767, September 2006.

L. Giovanini and J. Balderud. Game approach to distributed model predictive control. In *International Control Conference*, Glasgow,UK, December 2006.

R. S. Gopinath, B. W. Bequettelt, R. J. Roy, and H. Kaufman. Multirate mpc design for a nonlinear drug infusion system. In *American Control Conference*, pages 102–106, Baltimore, Maryland, USA, June-July 1994.

J. H. Lee, M. S. Gelormino, and M. Morari. Model predictive control of multi-rate sampled-data systems: a state-space approach. *International Journal of Control*, 55(1):153–191, January 1992.

S. Li, Y. Zhang, and Q. Zhu. Nash-optimization enhanced distributed model predictive control applied to the shell benchmark problem. *Information Sciences*, 170(2-4): 329–349, February 2005.

M. Mercangoz and F. J. Doyle. Distributed model predictive control of an experimental four-tank system. *Journal of Process Control*, 17(3):297–308, March 2007.

R. R. Negenborn, B. De Schutter, and J. Hellendoorn. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. *Engineering Applications of Artificial Intelligence*, 21(3):353–366, April 2008.

M. Ohshima, I. Hashimoto, H. Ohno, M. Takeda, T. Yoneyama, and F. Gotoh. Multirate multivariable model predictive control and its application to a polymerization reactor. *International Journal of Control*, 59 (3):731–742, 1994.

R. Scattolini. Architectures for distributed and hierarchical model predictive control - a review. *Journal of Process Control*, 19(5):723–731, 2009.

R. Scattolini and N. Schiavoni. A multi-rate model based predictive control. *IEEE Transactions on Automatic Control*, 40(6):1093–1097, June 1995.

S. Skogestad, M. Morari, and J. C. Doyle. Robust control of ill-conditioned plants: high-purity distillation. *IEEE Transactions on Automatic Control*, 33(12):1092–1105, December 1988.