2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Vulnerabilities in Distributed Model Predictive Control based on Jacobi-Gauss Decomposition

P. Chanfreut[1], J. M. Maestre[1], and H. Ishii[2]

[1]Department of System and Automation Engineering, University of Seville, Spain.
[2]Department of Computer Science, Tokyo Institute of Technology, Japan.

***Abstract*— This paper deals with the application of distributed model predictive control (DMPC) to a multi-agent system, where we consider the presence of malicious agents that inject information to disrupt its evolution. In particular, a cooperation based MPC scheme is threatened by non-conforming local controllers whose impact affects the entire plant and generates a security breach. We illustrate the results by numerical simulations.**

I. Introduction

Model predictive control (MPC) is a control strategy framed within the field of optimal control and whose inherent features have made it increasingly important in many industry applications [1]. An MPC controller can deal with multiple inputs and outputs, nonlinear dynamics, multiple objectives and also with constraints on the system variables [2]. Although it has been referred to as a single control strategy, MPC encloses a set of techniques which share certain commonalities, such as the use of prediction models, the optimization of cost functions or the receding horizon implementation.

The increasing presence of large-scale networked systems in which control plays a decisive role gives the distributed approach particular relevance nowadays [3], [4]. The degree of communication and cooperation between controllers is critical for global performance, especially when dealing with highly interacting systems. The latter is a differentiating characteristic of different DMPC approaches, as discussed in [5]. In this respect, problems based in the cooperation of controllers with a view to reaching the best possible overall performance have been widely studied, e.g., [6], [7].

This work is underpinned by the cooperation-based DMPC scheme proposed in [8]. Its core is an iterative negotiation in which the agents exchange and update their current information and in which parallel optimizations of a global performance index are carried out. Therefore, the plantwide problem is addressed with an objective function that assesses the overall evolution and that considers globally the effects of the couplings and the impact of the decisions taken by all controllers. In [9], the feasibility of all intermediate iteration results and the optimality of the limit points generated by the associated algorithm are proved. In [10], stability for both state and output feedback are stated, as well as discussing other significant properties of cooperation-based MPC. In addition, [11] shows an application of the latter to power systems and compares the results with other control schemes.

The objective of this work is to study this algorithm under the presence of unreliable agents and to analyze potential vulnerabilities. In particular, we deal with the introduction of false information that alters the secure exchange between controllers and create loss in performance in the line of [12] and [13], which have investigated this problem for a dual decomposition based DMPC scheme. It will be shown that there are alternatives for acting locally and steering the global evolution towards the interest of some of the agents, being also derived different consequences depending on their course of action. The latter poses a threat to DMPC applications with economic incentives to alter the prices, e.g. the energy market [14], [15]. In this context, [16] and [17] study the problem of designing dynamic mechanisms and defining conditions to ensure reliable performance.

Hereafter, this paper is organized as follows. First, in Section II, the distributed cooperative algorithm is introduced, preceded by the definition of the coupled system we will work with. In Section III, vulnerabilities of the DMPC scheme are studied. We present possibilities available for malicious agents to introduce the false information, as well as a discussion of the optimization of some of them in order to reach a higher level of self-benefit. In Section IV, the theoretical presentation is followed by the simulation of the alternatives considered.

II. Problem setting

We consider a system composed of M coupled subsystems, each of which has discrete-time linear dynamics given by

$$x_i(k+1) = A_{ii}x_i(k)+B_{ii}u_i(k)+\sum_{j=1,j\neq i}^{M}[A_{ij}x_j(k)+B_{ij}u_j(k)] \tag{1}$$

where $x_i \in \mathbb{R}^{n_i}$ is the subsystem state, $u_i \in \mathbb{R}^{m_i}$ represents its input vector, and $A_{il} \in \mathbb{R}^{n_i\times n_l}$ and $B_{il} \in \mathbb{R}^{n_i\times m_l}$ are matrices of the corresponding dimensions for all $i,l = 1,...,M$. For simplicity, let us define $w_i(k) = \sum_{j=1,j\neq i}^{M}[A_{ij}x_j(k)+B_{ij}u_j(k)]$ as the mutual interaction term.

Due to the application of MPC, the use of sequence vectors that represent trajectories will be recurrent throughout this paper. For this reason, the matrix representation that arises from extending (1) over a control horizon N is given next:

$$\mathbf{x}_i(k) = G_{x_i} x_i(k|k) + G_{u_i}\mathbf{u}_i(k) + G_{w_i}\mathbf{w}_i(k) \qquad (2)$$

where

$$\mathbf{x}_i(k) = \begin{bmatrix} x_i(k+1|k) \\ x_i(k+2|k) \\ \vdots \\ x_i(k+N|k) \end{bmatrix}, \quad \mathbf{u}_i(k) = \begin{bmatrix} u_i(k|k) \\ u_i(k+1|k) \\ \vdots \\ u_i(k+N-1|k) \end{bmatrix},$$

$$\mathbf{w}_i(k) = \begin{bmatrix} w_i(k|k) \\ w_i(k+1|k) \\ \vdots \\ w_i(k+N-1|k) \end{bmatrix}, \; G_{w_i} = \begin{bmatrix} I & & & \\ A_{ii} & I & & \\ \vdots & & \ddots & \\ A_{ii}^{N-1} & \cdots & \cdots & I \end{bmatrix},$$

$$G_{x_i} = \begin{bmatrix} A_{ii} \\ A_{ii}^2 \\ \vdots \\ A_{ii}^N \end{bmatrix} \text{ and } G_{u_i} = \begin{bmatrix} B_{ii} & & & \\ A_{ii}B_{ii} & B_{ii} & & \\ \vdots & & \ddots & \\ A_{ii}^{N-1}B_{ii} & \cdots & \cdots & B_{ii} \end{bmatrix}.$$

From a centralized viewpoint, the components of the state and input vectors are defined to be the aggregation of every x_i and u_i, which leads to the model

$$x(k+1) = A_{\text{cen}}x(k) + B_{\text{cen}}u(k) \qquad (3)$$

where $x \in \mathbb{R}^{\sum_i n_i}$, $u \in \mathbb{R}^{\sum_i m_i}$, $A_{\text{cen}} \in \mathbb{R}^{\sum_i n_i \times \sum_i n_i}$ and $B_{\text{cen}} \in \mathbb{R}^{\sum_i n_i \times \sum_i m_i}$. Hence, these are defined as

$$A_{\text{cen}} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1M} \\ A_{21} & A_{22} & \cdots & A_{2M} \\ \vdots & & \ddots & \\ A_{M1} & A_{M2} & \cdots & A_{MM} \end{bmatrix}, \quad x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_M(k) \end{bmatrix},$$

$$B_{\text{cen}} = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1M} \\ B_{21} & B_{22} & \cdots & B_{2M} \\ \vdots & & \ddots & \\ B_{M1} & B_{M2} & \cdots & B_{MM} \end{bmatrix} \text{ and } u(k) = \begin{bmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_M(k) \end{bmatrix}.$$

Likewise, the extension of the centralized model over a control horizon N can be written as

$$\mathbf{x}(k) = G_x x(k|k) + G_u\mathbf{u}(k) \qquad (4)$$

where

$$\mathbf{x}(k) = \begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N|k) \end{bmatrix}, \quad \mathbf{u}(k) = \begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N-1|k) \end{bmatrix},$$

$$G_x = \begin{bmatrix} A_{\text{cen}} \\ A_{\text{cen}}^2 \\ \vdots \\ A_{\text{cen}}^N \end{bmatrix} \text{ and } G_u = \begin{bmatrix} B_{\text{cen}} & & & \\ A_{\text{cen}}B_{\text{cen}} & B_{\text{cen}} & & \\ \vdots & & \ddots & \\ A_{\text{cen}}^{N-1}B_{\text{cen}} & \cdots & \cdots & B_{\text{cen}} \end{bmatrix}.$$

are matrices whose dimensions are calculated accordingly.

In this work, external disturbances are not taken into account for simplicity. However, it is straightforward to include them in the problem formulation if it is necessary. Also, note that mutual interaction derived from local couplings disappear when the problem is formulated from a centralized perspective.

A. Controller design procedure

The control problem is tackled with a cooperative scheme in which systemwide control goals are attained by an iterative negotiation between the corresponding M local MPC controllers. The latter will be performed together with an exchange of information that allows the agents to have certain knowledge about the decisions taken by the neighbors.

Let p denote the iteration index, then it will be assumed that every controller i at iteration p knows $\mathbf{u}_j^{p-1}(k)$ for all $j \in \{1, ..., M\}\backslash\{i\}$. For simplicity, the iteration superscript p is omitted hereunder, i.e., the variable is associated to iteration p and is either defined or to be defined.

1) Objective function: The function to be minimized by all agents i at each time step is defined as a weighted sum of the subsystems performance indices as

$$\sum_{l=1}^{M} \lambda_l \phi_l(\mathbf{u}_i(k), \mathbf{u}_{j\neq i}^{p-1}(k); x(k|k)) \qquad (5)$$

where ϕ_l represents the reduced centralized objective for agent l, and λ_l is a weighting factor. The former is defined as a quadratic function determined by the state and input trajectories of subsystem l, such that $\phi_l(\mathbf{x}_l(k), \mathbf{u}_l(k))$ is

$$\sum_{t=0}^{N-1} \left(\|x_l(k+t+1|k) - x_{l,\text{ref}}\|_{Q_l}^2 + \|u_l(k+t|k)\|_{R_l}^2 \right) \qquad (6)$$

where $x_{l,\text{ref}}$ denotes the state's reference for agent l, and $Q_l \geq 0$ and $R_l > 0$ are symmetric weighting matrices. The interdependence on the state trajectories is avoided applying (2) so that an expression on every subsystem inputs and current states is obtained as in (5).

2) Centralized problem: Again, it is interesting to consider the optimization problem from a centralized perspective. Plantwide constraints on the state and input trajectory vector posed as $\widehat{A}_x\mathbf{x}(k) \leq \widehat{b}_x$ and $\widehat{A}_u\mathbf{u}(k) \leq \widehat{b}_u$ are considered, being both grouped into a single inequality for $\mathbf{u}(k)$ using the definition of $\mathbf{x}(k)$. The resulting problem is

$$\begin{aligned} &\min_{\mathbf{u}(k)} \; \frac{1}{2}\mathbf{u}(k)^T H \mathbf{u}(k) + F(k)^T \mathbf{u}(k) \\ &\text{s.t.} \\ &C\mathbf{u}(k) \leq c(k) \end{aligned} \qquad (7)$$

where

$$H = G_u^T \widehat{Q} G_u + \widehat{R}, \quad F(k) = G_u^T \widehat{Q} G_x \left(x(k|k) - x_{\text{ref}} \right),$$

$$C = \begin{bmatrix} \widehat{A}_x G_u \\ \widehat{A}_u \end{bmatrix}, \; c(k) = \begin{bmatrix} \widehat{b}_x - \widehat{A}_x G_x \left(x(k|k) - x_{\text{ref}} \right) \\ \widehat{b}_u \end{bmatrix},$$

$$\widehat{Q} = \begin{bmatrix} Q & & \\ & \ddots & \\ & & Q \end{bmatrix}, \quad Q = \begin{bmatrix} \lambda_1 Q_1 & & \\ & \ddots & \\ & & \lambda_M Q_M \end{bmatrix},$$

$$\widehat{R} = \begin{bmatrix} R & & \\ & \ddots & \\ & & R \end{bmatrix} \text{ and } R = \begin{bmatrix} \lambda_1 R_1 & & \\ & \ddots & \\ & & \lambda_M R_M \end{bmatrix}.$$

If we take into account that $\mathbf{u}(k)$ can be expressed as a function of $\mathbf{u}_i(k)$ as $\mathbf{u}(k) = \sum_{i=1}^{M} M_i \mathbf{u}_i(k)$, where M_i

are matrices in $\mathbb{R}^{(\sum_{i=1}^{M} m_i)N \times m_i N}$, then we can derive from (7) the distributed problems equivalent to optimizing the function in (5).

3) Distributed problems: Now we proceed to introduce the change of variables into (7), so that an expression on every subsystem input trajectory is deduced. The optimization of the latter on the variable $\mathbf{u}_i(k)$ leads to

$$\begin{aligned} \min_{\mathbf{u}_i(k)} \quad & \frac{1}{2}\mathbf{u}_i(k)^T H_i \mathbf{u}_i(k) + F_i(k)^T \mathbf{u}_i(k) \\ \text{s.t.} \quad & \\ & CM_i\mathbf{u}_i(k) \leq c(k) - C\sum_{j\neq i}^{M} M_j \mathbf{u}_j^{p-1}(k) \end{aligned} \tag{8}$$

where $H_i = M_i^T H M_i$ and $F_i(k) = M_i^T H \sum_{j\neq i}^{M} M_j \mathbf{u}_j^{p-1}(k) + M_i^T F(k)$.

B. Algorithm

The Jacobi-Gauss algorithm [18, pp. 219-223] described in [8] shows the sequence of steps implemented to solve the DMPC control problem. Different properties regarding its feasibility, optimality and stability are discussed in [11] and [10] as mentioned, and given in detail in [9].

III. Attacks to the DMPC scheme

So far, it has been assumed that the algorithm works in a reliable information exchange setting in which all the agents behave as the strategy indicates. In this section, we proceed to consider the presence of malicious controllers that threaten the overall performance by introducing false information.

In particular, the DMPC scheme studied above pursues a convenient global behavior, which gives rise to the possibility of being locally better off when applying a different input to the agreed one. This is the motivation for carrying out an attack. In this work, the local welfare of an agent i is assumed to be assessed by expression (6) whose associated optimization problem is given next, but note that the results can be adapted for different alternatives.

$$\begin{aligned} \min_{\mathbf{u}_i(k)} \quad & \frac{1}{2}\mathbf{u}_i(k)^T \overline{H}_i \mathbf{u}_i(k) + \overline{F}_i(k)^T \mathbf{u}_i(k) \\ \text{s.t.} \quad & \\ & CM_i\mathbf{u}_i(k) \leq c(k) - C\sum_{j\neq i}^{M} M_j \mathbf{u}_j^{p-1}(k) \end{aligned} \tag{9}$$

where $\overline{H}_i = G_{u_i}^T \widehat{Q}_i G_{u_i} + \widehat{R}_i$ and $\overline{F}_i(k)^T = \left(x_i(k|k) - x_{i,\mathrm{ref}}\right)^T G_{x_i}^T \widehat{Q}_i G_{u_i} + \mathbf{w}_i^{p-1}(k)^T G_{w_i}^T \widehat{Q}_i G_{u_i}$.

A. False reference attack

Henceforth, let $a \in \{1, .., M\}$ denote the malicious controller. The first attack presented takes action through a false reference, $x_{a,\mathrm{ref}}^f$, introduced in the function that the attacker minimizes:

$$\begin{aligned} & \sum_{l=1, l\neq a}^{M} \lambda_l \phi_l(\mathbf{u}_a(k), \mathbf{u}_{j\neq a}^{p-1}(k); x(k|k)) + \lambda_a \sum_{n=0}^{N-1} \|u_a(k+n|k)\|_{R_a}^2 \\ & + \lambda_a \sum_{n=0}^{N-1} \|x_a(k+n+1|k) - x_{a,\mathrm{ref}}^f\|_{Q_a}^2 \end{aligned} \tag{10}$$

1) Calculation of an optimal $x_{a,\mathrm{ref}}^f$: In this part, we focus on finding the greatest effectiveness when conducting a *false reference* attack. It should be remarked that x_{ref} represents the vector $x_{\mathrm{ref}} = \begin{bmatrix} x_{1,\mathrm{ref}}^T & x_{2,\mathrm{ref}}^T & \cdots & x_{M,\mathrm{ref}}^T \end{bmatrix}^T$, which can be written as $x_{\mathrm{ref}} = \sum_{i=1}^{M} P_i x_{i,\mathrm{ref}}$.

Analytically, the attack that a conducts leads to a redefinition per iteration of $F(k) = G_u^T \widehat{Q} G_x \left(x(k|k) - \sum_{i=1, i\neq a}^{M} P_i x_{i,\mathrm{ref}} - P_a x_{a,\mathrm{ref}}^f \right)$ that appears in the linear term of problem (8) objective function. For the attacker, $x_{a,\mathrm{ref}}^f$ comes into play as a variable that can be used to improve its cost in a trasparent way. The optimal value of the false reference can be calculated as the solution of a Stackelberg game where the attacker is the leader, which adjusts $x_{a,\mathrm{ref}}^f$ to optimize its cost. The dependence of $x_{a,\mathrm{ref}}^f$ on p is omitted to keep notation as simple as possible.

The change introduced by optimizing this misleading parameter is illustrated in Figure 1. Here, the prediction length is set to $N = 1$ and the number of subsystems to $M = 2$. Thus, it is possible to represent in the plane $u_1^p(k)/u_2^p(k)$ the impact of the attacks in the negotation process. The increased effectiveness is reflected in the tendency, over iterations, of the points $(u_1^p(k), u_2^p(k))$ to those curves which represent lower local cost for the attacker.

2) Unconstrained case: If there are no constraints, the analytical calculation of $x_{a,\mathrm{ref}}^{f*}$ leads to the matrix equation given below, which provides $N\sum_i m_i$ equalities to determine n_a unknowns:

$$\begin{aligned} & G_u^T \widehat{Q} G_x P_a x_{a,\mathrm{ref}}^{f*} = \\ & G_u^T \widehat{Q} G_x \left(x(k|k) - \sum_{i=1, i\neq a}^{M} P_i x_{i,\mathrm{ref}} \right) + {C_1}^{-1} C_2 \end{aligned} \tag{11}$$

where $C_1 = M_a {H_a^{-1}}^T \overline{H}_a H_a^{-1} M_a^T \in \mathbb{R}^{N\sum_i m_i \times N\sum_i m_i}$ and $C_2 = -M_a {H_a^{-1}}^T \overline{F}_a(k) + C_1 H^T \sum_{j\neq a}^{M} M_j \mathbf{u}_j^{p-1}(k) \in \mathbb{R}^{N\sum_i m_i}$.

This result stems from the following:

1) Definition of the optimal inputs trajectories considering (8). Since all square matrices H_j are assumed to be non-singular, $\mathbf{u}_{i,opt}(k) = -H_i^{-1} F_i(k)$ and $\mathbf{u}_{a,opt}(k, x_{a,\mathrm{ref}}^f) = -H_a^{-1} F_a(k, x_{a,\mathrm{ref}}^f)$.
2) After substituting in (9) we get $\frac{1}{2} F^T(k, x_{a,\mathrm{ref}}^f) C_1 F(k, x_{a,\mathrm{ref}}^f) + C_2^T F(k, x_{a,\mathrm{ref}}^f)$..
3) Introduction of the partial derivative $\partial \left(F(k, x_{a,\mathrm{ref}}^f) \right) / \partial x_{a,\mathrm{ref}}^f = -P_a^T G_x^T \widehat{Q} G_u$ and

application of the chain rule, i.e.,

$$\frac{\partial\left(\frac{1}{2}F^T(k,x^f_{a,\mathrm{ref}})C_1F(k,x^f_{a,\mathrm{ref}})+C_2^TF(k,x^f_{a,\mathrm{ref}})\right)}{\partial x^f_{a,\mathrm{ref}}}=$$
$$-P_a^TG_x^T\widehat{Q}G_u\left(C_1F(k,x^f_{a,\mathrm{ref}})+C_2\right)$$

4) Assuming the non-singularity of C_1, an expression for $F(k,x^{f*}_{a,\mathrm{ref}})$ is determined, such that $F(k,x^{f*}_{a,\mathrm{ref}}) = -{C_1}^{-1}C_2 \in \mathbb{R}^{N\sum_i m_i\times 1}$. The dependence of $F(k,x^{f*}_{a,\mathrm{ref}})$ on $x^{f*}_{a,\mathrm{ref}}$ leads to (11).

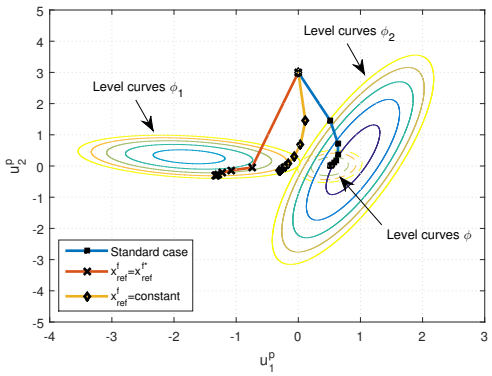


Fig. 1. Deviation in the negotation process as consequence of *false reference* attacks performed by agent 1.

B. Fake weights

In a *fake weights* attack the cause for the steering of the negotiation is a change in the values of the weighting factors λ_j $(j=1,...,M)$ in the attacker's objective function.

The first possibility is assuming that all λ_j for $j\neq a$ remain unaltered, but agent a optimizes

$$\sum_{l=1,l\neq a}^{M}\lambda_l\phi_l(\mathbf{u}_a(k),\mathbf{u}^{p-1}_{j\neq a}(k);x(k|k)) + \lambda^f_a\phi_a(\mathbf{u}_a(k),\mathbf{u}^{p-1}_{j\neq a}(k);x(k|k)) \tag{12}$$

with $\lambda^f_a > \lambda_a$, which entails that increased efforts are made to decrease the local cost of a.

1) Particular case: Selfish agent: As a specific case, we may consider an attack created by a selfish agent, who decides to optimize only its own subsystem by setting $\lambda_j = 0, \forall j \neq a$, and $\lambda_a = 1$. The input trajectory will be calculated irrespective of the cooperative feature of the DMPC scheme, solving in this case an optimization determined by (9) for $i=a$.

Similarly to the previous case, Figure 2 illustrates what would happen over iterations in the plane $u^p_1(k)/u^p_2(k)$, reflecting now the impact of acting in a selfish way.

C. Fake constraints

In a *fake constraints* attack the original DMPC framework is threatened by means of modifying the constraints imposed on the state's and input's evolution of subsystem a. That is,

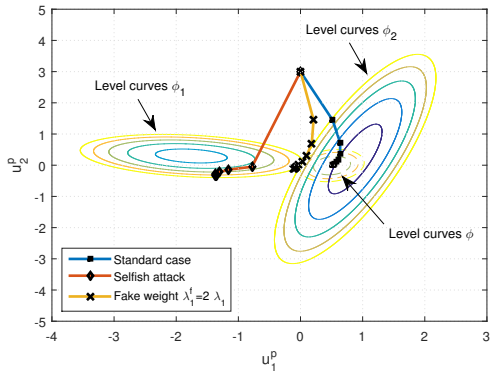


Fig. 2. Deviation in the negotation process as consequence of *fake weights* attacks performed by agent 1.

$$C^f M_a\mathbf{u}_a(k) \leq c^f(k) - C^f\sum_{j\neq a}^{M}M_a\mathbf{u}^{p-1}_j(k)$$

where C^f and $c^f(k)$ are misleading matrices.

Loosing or tightening the constraints is a cause of changes in the results provided. Therewith, we are in a similar situation as in the attacks above.

IV. Example: A Four Tank Plant

The purpose of this section is to show the risks and consequences for the algorithm entailed by acting under the attacks described in Section III.

The system used in this example consists of four interconnected water tanks whose scheme is presented in [19]. The plant works in a manner that the tanks at the top (3 and 4) discharge water into the ones at the bottom (1 and 2) and at the same time all of them are filled with a flow that comes from a storage tank. The latter is done via the actuation of two pumps, denoted as q_A and q_B.

The equations that model physically the evolution of the water levels h_i are given in [12] together with the associated LTI model:

$$x(k+1) = \begin{bmatrix} 0.9705 & 0 & 0.0205 & 0 \\ 0 & 0.9661 & 0 & 0.0195 \\ 0 & 0 & 0.9792 & 0 \\ 0 & 0 & 0 & 0.9802 \end{bmatrix} x(k) + \begin{bmatrix} 0.0068 & 0.0011 \\ 0.0002 & 0.0091 \\ 0 & 0.0137 \\ 0.0160 & 0 \end{bmatrix} u(k) \tag{13}$$

where the components of x and u represent respectively water levels and pump actions with respect to their corresponding operating point (see Table I).

A division of the global system into two subsystems is proposed: one composed of tanks 1 and 3, and, therefore, another one with tanks 2 and 4. Considering this, we decompose the matrix in (13) in the following matrices:

$$A_1 = \begin{bmatrix} 0.9705 & 0.0205 \\ 0 & 0.9792 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.9961 & 0.0195 \\ 0 & 0.9802 \end{bmatrix},$$

$$B_{1A} = \begin{bmatrix} 0.0068 \\ 0 \end{bmatrix}, \quad B_{1B} = \begin{bmatrix} 0.0011 \\ 0.0137 \end{bmatrix},$$

$$B_{2A} = \begin{bmatrix} 0.0002 \\ 0.0160 \end{bmatrix} \text{ and } B_{2B} = \begin{bmatrix} 0.0091 \\ 0 \end{bmatrix}.$$

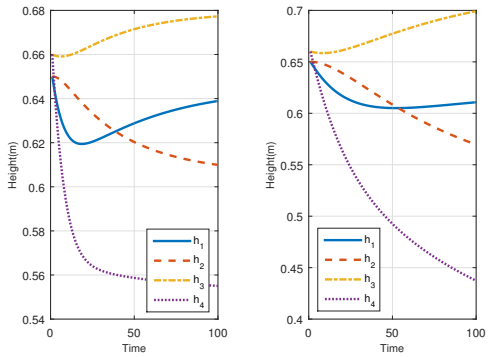
Regulation of the water levels in the tanks to $h_{1,\text{ref}} = 0.5\text{m}$, $h_{2,\text{ref}} = 0.6\text{m}$, $h_{3,\text{ref}} = 0.7\text{m}$ and $h_{4,\text{ref}} = 0.8\text{m}$ subject to the constraints specified in Table I is desired. Other parameters that are fixed are $N = 5$, $p_{max} = 50$, $\epsilon = 0.05$ and the weighting matrices as $Q_1 = Q_2 = I \in \mathbb{R}^2$ and $R_1 = R_2 = 0.01$.

TABLE I

OPERATING POINT AND CONSTRAINTS

$h_1^0 = 0.65\text{m}$	$q_A^0 = 1.63\text{m}^3/\text{h}$	$0.2\text{ m} \leq h_1(k), h_3(k) \leq 1.36\text{ m}$
$h_1^0 = 0.65\text{m}$	$q_B^0 = 2\text{m}^3/\text{h}$	$0.2\text{ m} \leq h_2(k), h_4(k) \leq 1.36\text{ m}$
$h_1^0 = 0.65\text{m}$		$0\text{ m}^3/\text{h} \leq q_A(k) \leq 3.26\text{ m}^3/\text{h}$
$h_1^0 = 0.65\text{m}$		$0\text{ m}^3/\text{h} \leq q_B(k) \leq 4\text{ m}^3/\text{h}$

1) False reference attacks: Firstly, a constant false reference, $x_{1,\text{ref}}^f = [-0.65, -0.3]^T$, is introduced by agent 1 from the beginning of the simulation. This causes a clear impact on the water levels arrived as shown in Figure 3(a). In particular, tank 4 is affected as we observe by comparing it with the standard outcome shown in Figure 7(a), which matches the expected centralized performance. Figure 3(b) and Figure 4 show respectively the water levels and the cumulative costs when the false reference is optimized by the attacker, proving how the deviation with respect to the standard results is widened in its favour.



(a) (b)

Fig. 3. Evolution of the tanks' water levels in the case in which an attack with $x_{1,\text{ref}}^f = [-0.65, -0.3]^T$ is performed (a), and in the one in which the false reference is optimized at each iteration (b).

2) Fake weights attacks: First, λ_1 in the objective function that agent 1 optimizes is increased, so λ_1^f becomes $2\lambda_1$. Figure 6 reflects how this is translated to the cumulative costs and Figure 5(a) presents the water levels' evolution. For the particular case of the *selfish agent* attack, Figure 5(b) is derived and the corresponding cumulative cost curves are shown in Figure 6.

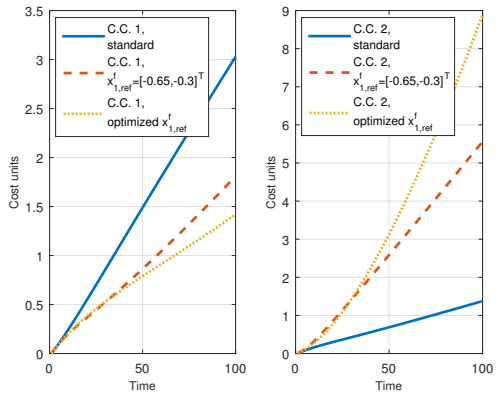
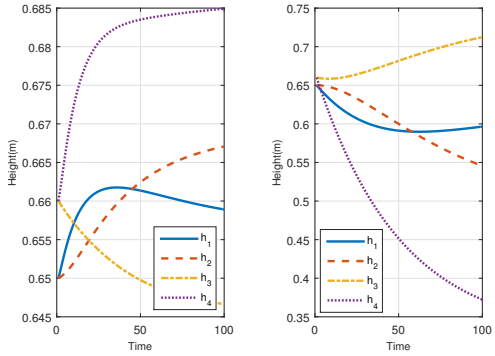


Fig. 4. Subsystems' cumulative costs in case of the *false reference* attacks compared with the standard results.



(a) (b)

Fig. 5. Evolution of the tanks' water levels in the in the case in which an attack with $\lambda_1^f = 2\lambda_1$ is performed (a) and in the one in which $\lambda_1^f = 1$ and $\lambda_2^f = 0$ (b).

3) Fake constraints attack: The algorithm has been simulated considering two cases in which agent 1 is the attacker. Firstly, the original interval which restricts the control of pump A is reduced to $1.55\text{m}^3/\text{h} \leq q_A(k) \leq 1.71\text{m}^3/\text{h}$. This results in Figure 7(b) and in the curves associated to fake constraints I in Figure 8. Secondly, a new limitation is applied, such that $0.63\text{m}^3/\text{h} \leq q_A(k) \leq 2\text{m}^3/\text{h}$. The latter case is indicated as fake constraints II in Figure 8.

V. CONCLUSIONS

In this paper we have highlighted some of the vulnerabilities of a distributed model predictive control scheme. These results are consistent with our previous research and show a need for improving distributed algorithms so that there are no incentives for malicious controllers to inject false information. We have also included several simulations to illustrate the consequences of the attacks in a distributed system. Future work will include an analysis of how this type of attacks can affect several theoretical properties of

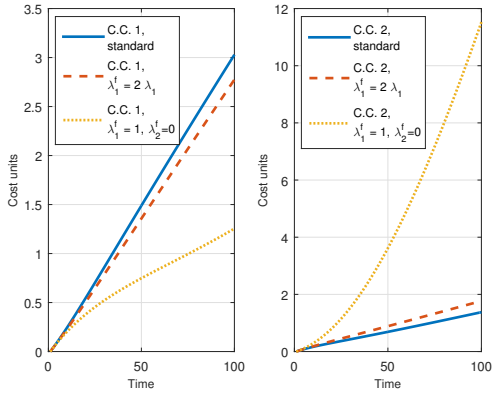
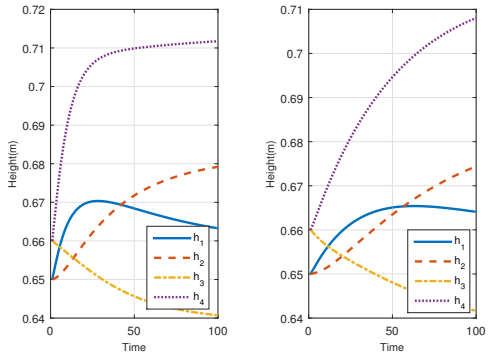


Fig. 6. Subsystems' cumulative costs in case of *fake weights* attacks compared with the standard results.



(a) (b)

Fig. 7. Evolution of the tanks' water levels in the standard case (a) and for an attack via a change in the constraints such that $1.55\mathrm{m}^3/\mathrm{h} \leq q_A(k) \leq 1.71\mathrm{m}^3/\mathrm{h}$ (b).

interest such as stability. Also, more realistic benchmarks and defense mechanisms will be considered.

ACKNOWLEDGEMENTS

Financial support by the Spanish MINECO project DPI2017-86918-R and the Japanese Society for the Promotion of Science (scholarship PE16048) is gratefully acknowledged.

REFERENCES

[1] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[2] E. F. Camacho and C. Bordons, *Model Predictive Control in the Process Industry*. Springer Science & Business Media, 2012.

[3] P. D. Christofides, R. Scattolini, D. M. de la Pena, and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Computers & Chemical Engineering*, vol. 51, pp. 21–41, 2013.

[4] J. M. Maestre and R. R. Negenborn, *Distributed Model Predictive Control Made Easy*. Springer Science & Business Media, 2013, vol. 69.

[5] J. B. Rawlings and B. T. Stewart, "Coordinating multiple optimization-based controllers: New opportunities and challenges," *Journal of Process Control*, vol. 18, no. 9, pp. 839–845, 2008.

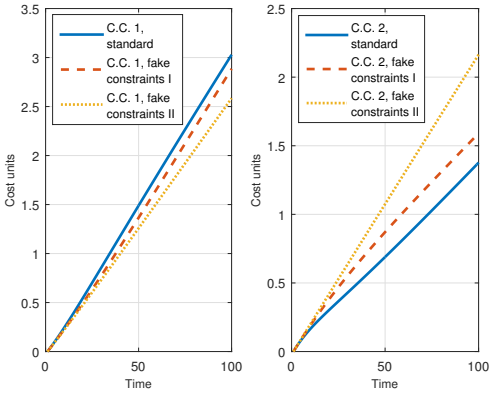


Fig. 8. Subsystems' cumulative costs in case of the *fake constraints* attacks compared with the standard results.

[6] J. Maestre, D. M. De La Pena, E. Camacho, and T. Alamo, "Distributed model predictive control based on agent negotiation," *Journal of Process Control*, vol. 21, no. 5, pp. 685–697, 2011.

[7] J. Maestre, D. Munoz De La Pena, and E. Camacho, "Distributed model predictive control based on a cooperative game," *Optimal Control Applications and Methods*, vol. 32, no. 2, pp. 153–176, 2011.

[8] A. N. Venkat, J. B. Rawlings, and S. J. Wright, "Stability and optimality of distributed model predictive control," in *Proc. 44th IEEE Conference on Decision and Control and 2005 European Control Conference*, 2005, pp. 6680–6685.

[9] A. N. Venkat, "Distributed model predictive control: theory and applications," Ph.D. dissertation, University of Wisconsin–Madison, 2006.

[10] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.

[11] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright, "Distributed MPC strategies with application to power system automatic generation control," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 6, pp. 1192–1206, 2008.

[12] P. Velarde, J. Maestre, H. Ishii, and R. Negenborn, "Scenario based defense mechanism for distributed model predictive control," in *Proc. 56th IEEE Conference on Decision and Control*, 2017.

[13] P. Velarde, J. M. Maestre, H. Ishii, and R. R. Negenborn, "Vulnerabilities in Lagrange-based distributed model predictive control," *Optimal Control Applications and Methods*, oca.2368. [Online]. Available: http://dx.doi.org/10.1002/oca.2368

[14] G. Wang, A. Kowli, M. Negrete-Pincetic, E. Shafieepoorfard, and S. Meyn, "A control theorist's perspective on dynamic competitive equilibria in electricity markets," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 4933–4938, 2011.

[15] Y. Okajima, K. Hirata, T. Murao, T. Hatanaka, V. Gupta, and K. Uchida, "Strategic behavior and market power of aggregators in energy demand networks," in *Proc. IEEE 56th Conference on Decision and Control*, 2017, pp. 694–701.

[16] T. Tanaka and V. Gupta, "Incentivizing truth-telling in mpc-based load frequency control," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 1549–1555.

[17] A. Gupta and T. Başar, "Dynamic incentive design in multi-stage linear-gaussian games with asymmetric information: A common information based approach," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014, pp. 414–419.

[18] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[19] I. Alvarado, D. Limon, D. Muñoz De La Peña, J. Maestre, M. Ridao, H. Scheu, W. Marquardt, R. Negenborn, B. De Schutter, F. Valencia *et al.*, "A comparative analysis of distributed mpc techniques applied to the hd-mpc four-tank benchmark," *Journal of Process Control*, vol. 21, no. 5, pp. 800–815, 2011.