Control Engineering Practice 114 (2021) 104879

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Scenario-based defense mechanism against vulnerabilities in Lagrange-based DMPC

José M. Maestre [a,*], Pablo Velarde [b], Hideaki Ishii [c], Rudy R. Negenborn [d]

[a] *Department of System and Automation Engineering, University of Seville, Spain*
[b] *Facultad de Ciencias de la Ingeniería e Industrias, Universidad UTE, Quito, Ecuador*
[c] *Department of Computer Science, Tokyo Institute of Technology, Japan*
[d] *Department of Maritime & Transport Technology, Delft University of Technology, The Netherlands*

ARTICLE INFO

Keywords:
Model predictive control
Distributed control
Multi-agent network
Cyber-security

ABSTRACT

In this paper, we present an analysis of the vulnerability of a distributed model predictive control (DMPC) scheme in the context of cyber-security. We consider different types of the so-called insider attacks. In particular, we consider the situation where one of the local controllers sends false information to others to manipulate costs for its own advantage. Then, we propose a popular scenario-based mechanism to protect or, at least, relieve the consequences of the attack in a typical DMPC negotiation process. The theoretical and algorithmic properties of this defense mechanism are also analyzed. A real case study based on a four tank plant is provided to illustrate both the consequences of the attacks and the defense mechanisms.

1. Introduction

Model predictive control (MPC) has become a popular control strategy due to the advantages it offers in comparison with other control approaches. An MPC controller can consider explicitly constraints on the manipulated variables and system states, nonlinearities on the model, delays, multiple objectives, etc. For this reason, this technique is widely used in numerous industrial applications, see, e.g., Camacho and Bordons (2004) and references therein. The main idea behind MPC is to calculate a control input sequence by solving a finite-horizon optimization problem (FHOP), based on the system model and its evolution, at each time instant. Only the first component of the control sequence is applied to the system at the current time; then the FHOP is solved again at the next time step (Grosso, Velarde, Ocampo-Martinez, Maestre, & Puig, 2017).

Nevertheless, geographically disperse systems such as road-traffic, logistics, transportation, water, and electrical networks may not allow to apply centralized MPC due to computational burden, issues with centralized modeling, data collection, and so on Negenborn and Maestre (2014). An alternative to deal with this kind of problems is to divide the whole system into subsystems, each one governed by a local MPC controller (or agent) that takes decisions and exchanges information with the other controllers under a negotiation process to obtain a possibly optimal global solution. This approach is called distributed MPC (DMPC) and has advantages such as ease of implementation, low computational effort in comparison with centralized MPC, modularity of the system, among others. In this regard, there are many types of possible implementations that can be adapted to the specific features of each problem (Negenborn & Maestre, 2014), e.g., sequential and parallel solutions, iterative and non-iterative methods, etc. For example, sequential and iterative architectures for DMPC are discussed in Liu, Chen, Muñoz de la Peña, and Christofides (2010). Tutorial reviews including design methods, algorithmic details, and an extensive discussion of the applications of DMPC are given in Christofides, Scattolini, de la Peña, and Liu (2013) and Maestre and Negenborn (2014).

A topic that deserves attention is the regular exchange of information during the negotiation process among the local DMPC controllers. Most DMPC schemes have been designed considering a coordinated negotiation process where all controllers work in a reliable way. However, a malicious controller could exploit the vulnerabilities of the network by sharing false information with other controllers, producing an undesirable behavior in the optimization process. At this point, it is possible to speak about cyber-security in the context of DMPC. Well-known examples include Stuxnet (Kushner, 2013), which was the first known worm designed to reprogram Siemens programmable logic controllers, forcing various shutdowns at the Natanz (Iran) nuclear plant (Albright, Brannan, & Walrond, 2010), and Crash Override, which caused massive blackouts in Ukraine in 2016 (Bindra, 2017). For this reason, in recent years, numerous investigations have devoted their efforts to finding solutions to these problems (Cheng, Shi, & Sinopoli, 2017). The goal of cyber-security is to protect systems against

[*] Corresponding author.
E-mail addresses: pepemaestre@us.es (J.M. Maestre), pablo.velarde@ute.edu.ec (P. Velarde), ishii@c.titech.ac.jp (H. Ishii), R.R.Negenborn@tudelft.nl (R.R. Negenborn).

https://doi.org/10.1016/j.conengprac.2021.104879
Received 23 February 2021; Received in revised form 30 May 2021; Accepted 22 June 2021
Available online 5 July 2021

threat by preserving the confidentiality, availability, and integrity of the exchanged information (Radvanovsky & Brodsky, 2016). Classical application areas are protection systems (Sheng, Chan, Li, Xianzhong, & Xiangjun, 2007), Internet users (Kritzinger & Von Solms, 2010), logistics (Li, Negenborn, & De Schutter, 2015; Nabais, Negenborn, Carmona, & Ayala, 2015; van Riessen, Negenborn, Lodewijks, & Dekker, 2015), power systems (Barreto, Giraldo, Cardenas, Mojica-Nava, & Quijano, 2014; Chakhchoukh & Ishii, 2015; Mc Namara, Negenborn, De Schutter, Lightbody, & McLoone, 2016; Soudbakhsh, Chakrabortty, & Annaswamy, 2017), among others. Control systems are not exempt from possible cyber-attacks (Teixeira, Sou, Sandberg, & Johansson, 2015; Zhu & Basar, 2015), with consequences ranging from performance loss to instability. A resilient multivariable control framework for large-scale urban traffic networks subject to several types of cyberattacks is addressed in Mercader and Haddad (2021). Also, Dhal and Roy (2013) presents cyber-security risk assessment for supervisory control and data acquisition (SCADA) and distributed control system networks.

In this context, cyber-security does not deal with security from a computer or telecommunications viewpoint, but with providing an additional layer of defense once attackers have already penetrated the control system. By exploiting the knowledge of the system dynamics, controllers can detect and isolate themselves from these threats, or at least mitigate their consequences. For example, in Ananduta, Maestre, Ocampo-Martinez, and Ishii (2020) Bayesian inference is used to detect noncompliant controllers in a distributed MPC setting so that negotiations can be stopped accordingly. In Maestre, Trodden, and Ishii (2018), the issue of noncompliance is addressed using a passive approach that allows agents to be protected against a limited amount of noncompliances in a distributed tube MPC setting. In Braun, Albrecht, and Lucia (2020) and Pierron, Árauz, Maestre, Cetinkaya, and Maniu (2020), stochastic MPC is used to build a tree and improve the controller performance in case of packet losses/attacks. This is also the approach followed in Yang, Li, Dai, and Xia (2019), where denial-of-service (DoS) attacks are considered, although in a setting where only one subsystem is allowed to optimize at each time step. DoS attacks are also dealt with within the framework of Sun, Zhang, and Shi (2019), where conditions are derived to guarantee exponential stability of the closed-loop system. A different strategy is followed in Romagnoli, Krogh, and Sinopoli (2019), where the control software is periodically reinstalled to mitigate the impact of cyber agressions. Another interesting line of work is given in Trodden, Maestre, and Ishii (2020), which studies the extent of the control input set that can be gained by an attacker before relevant properties such as the existence of invariant sets is lost. Other proposed methods include encryption and coding schemes as a means to secure communication (Darup, Redder, Shames, Farokhi, & Quevedo, 2017; Miao, Zhu, Pajic, & Pappas, 2016) and the use of artificial intelligence methods such as neural-networks (Wu et al., 2018) and machine learning (Chen, Wu, & Christofides, 2020) to detect cyber-attacks.

In this paper, we explore cyber-security issues in Lagrange-based DMPC schemes. Recent studies in this direction include the schemes, e.g., for dual (Velarde, Maestre, Ishii, & Negenborn, 2018), and Jacobi–Gauss decomposition (Chanfreut, Maestre, & Ishii, 2018), where anomalies can be detected during the negotiation process. Moreover, some works discuss techniques to reject malicious agents during the consensus process (Ananduta et al., 2020; Ishii et al., 2020; Onogawa et al., 2019; Yang et al., 2019). This work analyzes Lagrange based DMPC and, following Velarde, Maestre, Ishii, and Negenborn (2017), shows how a malicious controller can take advantage of the vulnerabilities of the scheme to increase its own benefit at the cost of other controllers. To mitigate the consequences of the attacks, scenario-based techniques are proposed to gain robustness (Calafiore & Fagiano, 2013). In particular, scenarios taken from trustworthy historical price information are used to robustify the control network against malicious controllers. As a matter of a fact, scenario-based MPC has been extensively used to deal with external uncertainties affecting control systems. For instance, multi-scenario MPC (MS-MPC) optimizes a single control sequence valid for all potential scenarios taking into account their probability of occurrences, e.g., in the context of water systems (Grosso et al., 2017; van Overloop, Weijs, & Dijkstra, 2008; Tian et al., 2019; Velarde, Tian, Sadowska, & Maestre, 2019) and smart grids control (Olivares, Lara, Cañizares, & Kazerani, 2015). Based on this background, we propose to incorporate this approach in the DMPC formulation as a way to secure dual decomposition DMPC and deal with the internal threats from the distributed network. Also, as a case study to illustrate the proposed defense method, we carry out experiments in a real four tank plant.

The remainder of this paper is organized as follows. First, dual decomposition based MPC is briefly introduced in Section 2. Section 3 presents a number of threat schemes in the context of Lagrange based DMPC, where a malicious controller can exploit the algorithm. Section 4 provides a secure dual decomposition technique based on MS-DMPC to mitigate the impact that an attacker can cause to the other controllers. Section 5 analyzes the impact of the proposed scheme regarding the satisfaction of algorithmic and theoretical properties of interest. In Section 6, the aforementioned case study is presented to show the effects of potential attacks and how the proposed mechanisms relieve this issue. Finally, conclusions are drawn in Section 7.

A preliminary version of this paper has appeared as Velarde et al. (2017); the current paper analyzes theoretical and algorithmic properties of the scenario based defense mechanism and moreover the results of the experiments based on the four tank plant are presented.

2. Dual decomposition based DMPC

In this section, we present a commonly used distributed optimization algorithm based on dual decomposition (Biegel, Stoustrup, & Andersen, 2014; Giselsson & Rantzer, 2014). Let us consider a distributed system composed of N subsystems defined by discrete-time linear time-invariant models. The dynamics of subsystem i are given by

$$x_i[k+1] = A_i x_i[k] + B_i u_i[k], \tag{1}$$

where $x_i \in \mathbb{R}^{n_{x,i}}$ and $u_i \in \mathbb{R}^{n_{u,i}}$ denote the states and input of the system, respectively. $A_i \in \mathbb{R}^{n_{x,i} \times n_{x,i}}$ is the state matrix and $B_i \in \mathbb{R}^{n_{x,i} \times n_{u,i}}$ represents the input matrix. Moreover, $n_{x,i}$ and $n_{u,i}$ represent the number of states and the number of inputs of the subsystem i, respectively. Each subsystem is subject to convex state and input constraints

$$x_i[k] \in \mathcal{X}_i, \quad \forall k \in \mathbb{Z}_+, \tag{2a}$$

$$u_i[k] \in \mathcal{U}_i, \quad \forall k \in \mathbb{Z}_+, \tag{2b}$$

where $\mathbb{Z}_+$ denotes the set of non-negative integer numbers. Let the aggregated vectors of states and inputs be $x[k] = \left[x_1[k]^{\mathrm{T}} \cdots x_N[k]^{\mathrm{T}}\right]^{\mathrm{T}}$ and $u[k] = \left[u_1[k]^{\mathrm{T}} \cdots u_N[k]^{\mathrm{T}}\right]^{\mathrm{T}}$, respectively, where $x \in \mathbb{R}^{n_x}$, $n_x = \sum_{i=1}^{N} n_{x,i}$, $u \in \mathbb{R}^{n_u}$, and $n_u = \sum_{i=1}^{N} n_{u,i}$. The N subsystems are also subject to constraints coupling the inputs:

$$Cu[k] = \sum_{i=1}^{N} C_i u_i[k] \leq c, \tag{3}$$

where $C \in \mathbb{R}^{n_c \times n_u}$, $C_i \in \mathbb{R}^{n_c \times n_{u_i}}$ and $c \in \mathbb{R}^{n_c}$. Note that this formulation can easily cover typical coupling constraints in the states and in the inputs.

We assume that a convex stage cost function for each subsystem is given by

$$\ell_i(x_i[k+1], u_i[k]). \tag{4}$$

Each subsystem i is controlled by a local MPC controller. The main idea of (centralized and distributed) MPC is to obtain a control signal by solving, at each time step, an FHOP that takes into account the prediction model of each subsystem. In particular, (1) is used to predict the evolution of the system along a given horizon N_{p} as a function of the sequence of inputs provided. In this way, it is possible to calculate a control sequence $u_i^*[k : k+N_\mathrm{p}-1]$ that optimizes (4) along the horizon. The first component of the optimal control sequence obtained is implemented at the current time step, and the problem is solved at the next time step following a receding horizon strategy. The optimization problem over a fixed time prediction horizon $N_\mathrm{p} \in \mathbb{Z}_+$ can be written as

$$u_i^*[k : k+N_\mathrm{p}-1] = \arg \min_{u_i[k:k+N_\mathrm{p}-1]} \sum_{j=k}^{k+N_\mathrm{p}-1} \ell_i(x_i[j+1], u_i[j]), \tag{5}$$

subject to (1)–(3), assuming that the predicted control actions and states of the rest of the subsystems are known.

From an overall perspective, the stage cost function becomes

$$\ell(x[k+1], u[k]) = \sum_{i=1}^{N} \ell_i(x_i[k+1], u_i[k]). \tag{6}$$

In this way, the optimization problem, from a global viewpoint, is given by

$$\min_{u[k:k+N_\mathrm{p}-1]} \sum_{j=k}^{k+N_\mathrm{p}-1} \ell(x[j+1], u[j]), \tag{7}$$

subject to (1)–(3).

Due to the coupling in (3), controllers have to share information. It is necessary to consider the role played by coupling variables explicitly. Hence, the controllers have to coordinate their actions using a negotiation process.

The dual decomposition approach consists of converting the *coupled* variables into local versions and then incentivize via cost the obtainment of a coordinated value. In this sense, the performance index is reformulated by means of the associated Lagrange multipliers as

$$L(\eta[k], \Lambda[k]) = \sum_{j=k}^{k+N_\mathrm{p}-1} (\ell(x[j+1], u[j]) + \lambda[j]^\mathrm{T}(Cu[j]-c)), \tag{8}$$

where $\eta[k] = \left[x[k+1 : k+N_\mathrm{p}]^\mathrm{T}, u[k : k+N_\mathrm{p}-1]^\mathrm{T}\right]^\mathrm{T}$ is defined as the vector composed of the states and inputs along the horizon N_p, $\lambda[j] \in \mathbb{R}^{n_c}$ are the multipliers associated with the coupling constraints (3), and $\Lambda[k] = \lambda[k : k+N_\mathrm{p}-1]$ is the sequence of the Lagrange multipliers along the horizon.

Remark 1. Lagrange multipliers can be interpreted as prices that are used to coordinate the subsystems regarding the fulfillment of coupling constraints (Biegel et al., 2014).

The optimal value of the problem for a given sequence of prices is defined as

$$g(\Lambda[k]) = \min_{u[k:k+N_\mathrm{p}-1]} \sum_{j=k}^{k+N_\mathrm{p}-1} (\ell(x[j+1], u[j]) + \lambda[j]^\mathrm{T}(Cu[j]-c)), \tag{9}$$

subject to (1) and (2), and allows to deal with (7) in a distributed manner by solving its dual problem

$$\begin{aligned} &\text{maximize } g(\Lambda[k]), \\ &\text{subject to } \Lambda[k] \geq 0, \end{aligned} \tag{10}$$

by using a distributed gradient search, where ≥ represents componentwise inequality. The distributed control problem solved by dual decomposition is summarized in Algorithm 1 (Biegel et al., 2014).

Dual decomposition has been used in several applications, e.g., building temperature regulation (Yushen, Shuai, Xie, & Johansson, 2014), coordinating a network of households (Larsen, Van Foreest, & Scherpen, 2014), ships (Zheng, Negenborn, & Lodewijks, 2017), and logistics (Li, Negenborn, & De Schutter, 2017).

Algorithm 1 Dual decomposition based DMPC.

```
1: Each controller initializes its prices (Lagrange multipliers) $\Lambda[k] \geq 0$.
   All agents have the same initial values for the Lagrange multipliers.
2: repeat
3: Each controller solves its local optimization problem with the
   current value of $\Lambda[k]$, i.e.,
     $$\min_{u_i[k:k+N_\mathrm{p}-1]} \sum_{j=k}^{k+N_\mathrm{p}-1} (\ell_i(x_i[j+1], u_i[j]) + \lambda[j]^T C_i u_i[j]), \tag{11a}$$
   subject to
     $$x_i[j+1] = A_i x_i[j] + B_i u_i[j], \quad \forall j \in [k : k+N_\mathrm{p}-1], \tag{11b}$$
     $$x_i[j] \in \mathcal{X}_i, \quad \forall j \in [k : k+N_\mathrm{p}-1], \tag{11c}$$
     $$u_i[j] \in \mathcal{U}_i, \quad \forall j \in [k : k+N_\mathrm{p}-1]. \tag{11d}$$
   The solution of the optimization problem is denoted as $x_i^*[k+1 : k+N_\mathrm{p}]$,
   $u_i^*[k : k+N_\mathrm{p}-1]$. Then these values are exchanged with other controllers.
4: Each controller $i$ determines the violations of the coupling constraints
   $s[k] \triangleq \sum_{i=1}^{N} C_i u_i^*[k] - c$, $S[k] = s[k : k+N_\mathrm{p}-1] \in \mathbb{R}^{N\mathrm{p} \times n_c}$ and
   calculates the new prices along the horizon $\Lambda[k] := \max[0, \Lambda[k] + \gamma S[k]]$,
   where $\gamma$ is the step size.
5: until $\max(S[k]) < \epsilon$, where $\epsilon$ is a prespecified threshold, or the
   maximum number of iterations reached.
6: Each subsystem implements the first component of the control
   sequence $u_i^*[k : k+N_\mathrm{p}-1]$.
7: Let $k = k+1$ and return to step 1.
```

3. Attacks in Lagrange-based DMPC

Algorithm 1 works in a reliable information exchange setting. If one of the controllers is malicious, the whole system can fail. In particular, we consider situations where one of the controllers is an attacker that uses false information affecting performance. For example, the attacker may manipulate the information shared with others to steer the overall negotiation process. This situation can be seen as a *Stackelberg game* (Başar & Srikant, 2002), where the attacker is the leader, and the rest of the local controllers become followers. Three different ways in which an attacker can take advantage by exchanging false information with other controllers of the subsystems are presented in this section.

3.1. False reference

A fake reference attack consists of the *strategic* use of the reference so as to steer the negotiation process. Any agent performing this attack is setting the local reference in a way that does not correspond to its *true* preference. Instead, it uses the altered reference as a means to make the controlled signal follow its desired value more closely at the expense of the rest of the controllers.

Let us consider that a controller $m \in \{1, \ldots, N\}$ attacks the rest of the controllers by using a false reference ($x_{m_\mathrm{ref}}^*$) to bias the negotiation. Therefore, the stage cost function optimized by controller m is given by

$$\ell_m^*(x_m[k+1], u_m[k]) = \ell_m(x_m[k+1] - x_{m_\mathrm{ref}}^*, u_m[k]). \tag{12}$$

The use of a false reference can steer the negotiation process towards a result that is more beneficial for the attacker. In this sense, there is no incentive for the controllers to be honest regarding their real references because they can be better off in this way from a local perspective. Therefore, other controllers could follow the same type of strategic behavior, i.e., they could *hide* their preferences to bias the negotiation. The resulting situation would be that of a dynamic non-cooperative game where malicious agents use their references strategically to attain better local costs at the expense of those controllers that set their optimization problems using values that reflect their actual preferences. Therefore, compliant agents would behave in a cooperative manner that would be exploited by malicious agents pursuing their self-interest.

Finally, Figs. 1 and 2.b show the case of the false reference attack for simple academic examples. Note that these figures are provided only for illustration purposes and therefore details regarding the numerical values employed are omitted.

3.2. Fake constraints

Another way in which the attacking controller m can take advantage from its neighbor subsystem carrying out the optimization problem is to use fake constraints, e.g.,

$$x_m[j] \in \mathcal{X}_m^*, \quad \forall j \in [k : k + N_\mathrm{p} - 1], \tag{13a}$$

$$u_m[j] \in \mathcal{U}_m^*, \quad \forall j \in [k : k + N_\mathrm{p} - 1], \tag{13b}$$

where $\mathcal{X}_m^*$ and $\mathcal{U}_m^*$ are the modified sets with *fake* constraints.

The remaining subsystems optimize their objective functions by considering their original constraints while the attacker uses constraints that steer the negotiation process by reducing its own cost function.

Fig. 2.c illustrates the effect of this attack, in terms of the cost using a simple academic example with two agents and two shared variables. As can be seen, the attack has a very strong impact on the negotiation because it limits the set of feasible values of the negotiated variables.

3.3. Fake prices

To obtain a better local cost, agent m can modify its performance index by including a new coefficient denoted as α by solving

$$\min_{u_m[k:k+N_\mathrm{p-1}]} \sum_{j=k}^{k+N_\mathrm{p}-1} \left(\ell_m(x_m[j+1]; u_m[j]) + \frac{\lambda[j]^T}{\alpha} C_m u_m[j] \right) \tag{14}$$

with $\alpha > 1$, subject to (11)(b)–(11)(d). The solution is biased towards the interests of agent m because (14) is equivalent to

$$\min_{u_m[k:k+N_\mathrm{p}-1]} \sum_{j=k}^{k+N_\mathrm{p}-1} \left(\alpha \ell_m(x_m[j+1]; u_m[j]) + \lambda[j]^T C_m u_m[j] \right), \tag{15}$$

so that the global stage cost becomes

$$\ell^*(x[k+1], u[k]) = \sum_{j \neq m} \ell_j(x_j[k+1], u_j[k]) + \alpha \ell_m(x[k+1], u[k]), \tag{16}$$

prioritizing the local cost of agent m by α during the execution of Algorithm 1.

Again, Fig. 2.d shows the effect of this attack on the trajectory followed by the negotiated variables and the corresponding local costs. As can be seen, the scaling factor α has a great power to promote the local interest of the malicious agent in the negotiation.

3.4. Attack tuning

All the considered attacks correspond to malicious modifications of local optimization problems so as to improve the corresponding utility at the expense of the rest of the system. This is possible because the distributed control algorithm considered, which is very popular due to its simplicity, does not discourage this type of strategical behavior to increase local performance.

How to *optimize* these modifications is problem specific and adjustments may be required depending on the reactions of other subsystems. Nevertheless, the local reference can be increased (decreased) above its current value to force the raise (decrease) of the corresponding controlled variable. Likewise, the constraints can be modified so as to restrict the optimization to certain values which result more beneficial for its local cost function. Finally, whenever the Lagrangian multipliers are divided, the overall optimization will be driven towards the local goal, with increased effects on local performance as α grows.

4. Secure scenario-based DMPC

As seen in the previous section, the negotiation process can be manipulated so that the values of the prices in the coordination mechanism deviate from their optimal values. It is necessary to implement a method that relieves the potential effects of an intentional attack whenever this situation is detected. To this end, we propose a scenario-based approach to robustify the control network against malicious controllers. In particular, trustworthy (possibly audited) price information based on historical data will be used to generate scenarios.

Since it is not possible to be certain about the type and number of attacks that can occur, we must first set the target attack levels to apply security measures, and then make sure that the actual attack level of the system to be protected falls within the level of the measure applied with some safety margins. To this end, the detection of attacks to activate the defense mechanism is based on two simple criteria that can be easily implemented.

Let $\Lambda[k]$ be the vector of prices calculated at time step k and $\widehat{\Lambda}[k]$ be a vector of prices with nominal values for the current situation of the system obtained from historical data. The triggers proposed for the activation of the defense mechanism are the following:

- Abnormal price values: this condition can be expressed as

  $$|\Lambda[k] - \widehat{\Lambda}[k]| \geq \kappa, \tag{17}$$

  where κ is a threshold that establishes a bound on the norm of the deviation of the prices with respect to their expected values.
- Abnormal performance: if the performance of the overall system with the current prices is worse than that with nominal prices, the coordination is not working properly. Hence, the following condition can be checked based on (9):

  $$g(\Lambda[k]) - g(\widehat{\Lambda}[k]) > 0. \tag{18}$$

  Note that (18) can be checked in a distributed fashion. Also, the fact that the local version of (18) is satisfied for many agents can be an indication that the coordination mechanism is under attack.

It must be noted that the conditions (17) and (18) are a little conservative and it is possible that some attacks are not detected. For example, it may happen that

$$g(\Lambda[k]) - g(\widehat{\Lambda}[k]) \leq 0 \tag{19}$$

with the system being under attack. Such situation corresponds to the case where the attack leads to an increase of cost that is below the increase generated by the use of nominal prices. Therefore, the effect of the attack is not significant enough to be detected. Likewise, any attack whose effect on the prices satisfies

$$|\Lambda[k] - \widehat{\Lambda}[k]| < \kappa, \tag{20}$$

with some constant $\kappa > 0$, will not be detected.

4.1. Scenario generation

Scenario generation is necessary to relieve the effects of an attacker inside the network. The availability of audited historical data is demanding for some problem settings, but is also reasonable for systems with periodic exogenous inputs such as power grids and water networks, where clear patterns can be identified depending on factors such as time, day of the week and weather. It might also be performed using a stochastic model to generate synthetic data (Schildbach & Morari, 2015), which is the approach we follow. Details are given below.

In order to generate different scenario evolutions, we assume a nominal state is available at time step k for a system with periodic state trajectories. The initial value of the model used for the MPC problem at each step is then perturbed by white noise as

$$\widetilde{\mathbf{x}}_\mathbf{i}[k] = \mathbf{x}_\mathbf{i}[k] + \mathrm{n}[k], \tag{21}$$

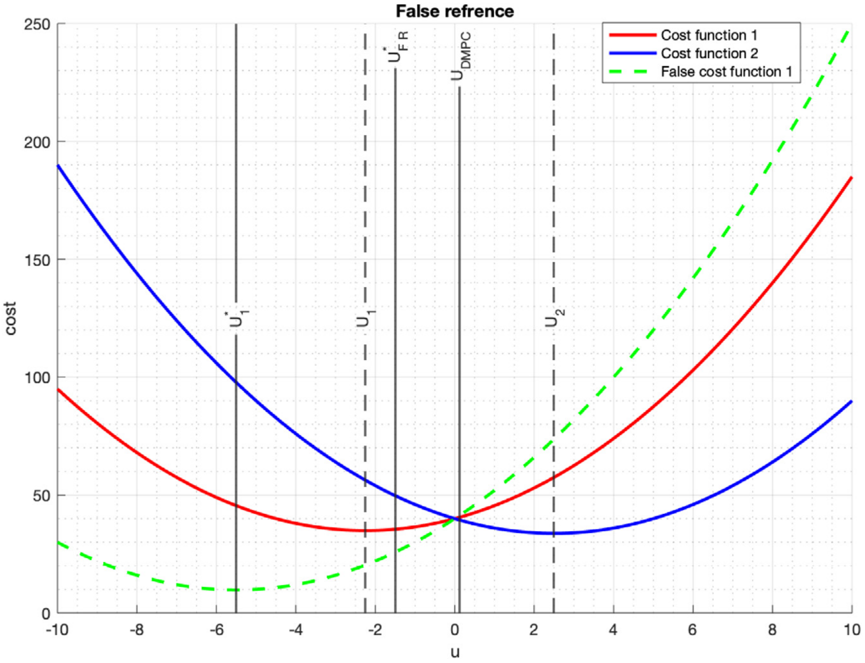


Fig. 1. Result of the negotiation process carried out by two integrators that share the same manipulated variable: U_1 and U_2 represent the minimizer of the corresponding local cost function, which penalizes quadratically the control effort and the reference tracking error. The use of a false reference by agent 1 modifies its true cost function (red line), displacing it to the left (green dashed line), and the same happens with its minimizer, which becomes U_1^*. As a consequence, the negotiated value is steered from U_{DMPC} to U_{FR}^*, which is indeed closer to U_1, attaining an improved local cost with respect to its true preferences (red line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

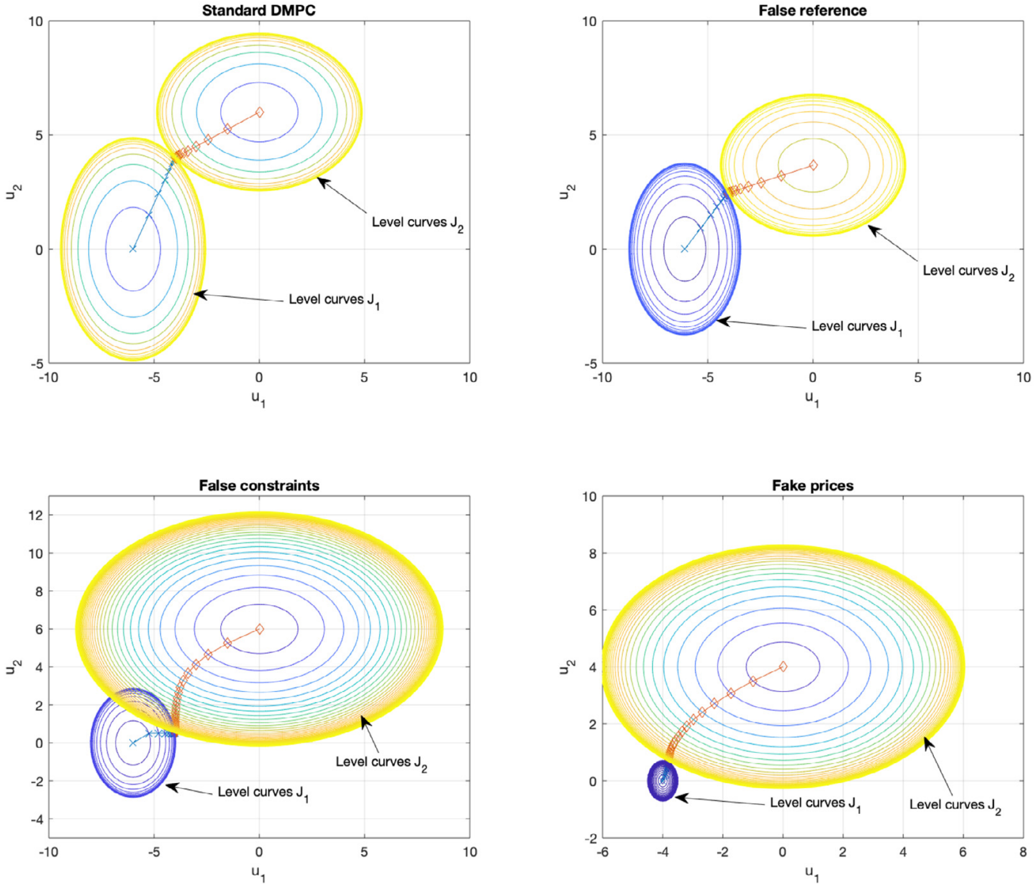


Fig. 2. Local cost level curves and trajectories of negotiated variables for a simple system with two input-coupled agents. The negotiation is performed under different conditions over the manipulated variables u_1 and u_2 (prediction horizon $N_p = 1$): (a) standard DMPC (upper-left); (b) false reference (upper-right); (c) fake constraints (lower-left); and (d) fake prices (lower-right). In all the attacks considered, the negotiation outcome is closer to the minimizer of agent 1, resulting in increased costs for agent 2.

where $\widetilde{\mathbf{x}}_{\mathbf{i}}[k]$ represents the perturbed measurement of each state containing noise $\mathrm{n}[k] \sim \mathcal{N}(\mu, \sigma^2)$, i.e., a Gaussian distribution with mean μ and standard deviation σ. In this way, several experiments are repeated, and the price information $\lambda_i[k]$ is collected as a scenario for each state variable. Starting from the state $\widetilde{x}_i[k]$, the FHOP provides us with predictions of the state, input, and Lagrange multipliers. In particular, the scenario generation FHOP is formulated as follows:

$$\min_{u_i[k:k+N_\mathrm{p}-1]} \sum_{l=k}^{k+N_\mathrm{p}-1} (\ell_i(\widetilde{x}_i[l+1], u_i[l]) + \lambda[l]^T C_i u_i[l]), \tag{22a}$$

subject to

$$\widetilde{x}_i[l+1] = A_i \widetilde{x}_i[l] + B_i u_i[l], \quad \forall l \in [k : k+N_\mathrm{p}-1], \tag{22b}$$

$$\widetilde{x}_i[l] \in \mathcal{X}_i, \quad \forall l \in [k : k+N_\mathrm{p}-1], \tag{22c}$$

$$u_i[l] \in \mathcal{U}_i, \quad \forall l \in [k : k+N_\mathrm{p}-1]. \tag{22d}$$

Note that $\lambda_i[k : k+N_\mathrm{p}-1]$ is computed at each time instant k. Hence, the set of trustworthy price scenarios of each controller i can be expressed as

$$\widehat{\Lambda}_i[k] = \{\lambda_i^1[k : k+N_\mathrm{p}-1], \lambda_i^2[k : k+N_\mathrm{p}-1], \ldots, \lambda_i^{N_s}[k : k+N_\mathrm{p}-1]\}, \tag{23}$$

where N_s is the number of scenarios generated, $\lambda_i^s \in \mathbb{R}^{n_c}$ corresponds to the sth scenario of the controller i for $s \in [1, N_s]$. These scenarios are stored in a database. Note that offline training is carried out to store that information in a database to be used in an online way.

Remark 2. Since our method relies on using data from previous executions, it is more suitable for periodic systems and repetitive tasks. Non-periodic-behavior systems could be considered as well if enough information from the past is available so that prices can be analyzed for any possible system state.

4.2. Multi-scenario DMPC (MS-DMPC)

MS-DMPC provides robustness by computing a unique control input that ensures the satisfaction of the local constraints for all the potential trajectories determined by the set of scenarios.

One issue that deserves special attention is the number of scenarios (N_s) that guarantees the robustness of the whole system. A higher number of scenarios results in an over conservative control input and may lead to significant computation burden. The problem formulation of MS-DMPC for each controller $i \in [1, N]$ at each time instant k is expressed as

$$\min_{u_i[k:k+N_\mathrm{p}-1]} \sum_{s=0}^{N_s} \rho_i^s \sum_{j=k}^{k+N_\mathrm{p}-1} (\ell_i(x_i^s[j+1], u_i[j]) + \lambda_i^s[j]^T C_i u_i[j]), \tag{24a}$$

subject to

$$x_i^s[j+1] = A_i^s x_i^s[j] + B_i u_i[j], \tag{24b}$$

$$x_i^s[j] \in \mathcal{X}_i, \quad \forall j \in [k : k+N_\mathrm{p}-1], \quad \forall s \in [0, N_s], \tag{24c}$$

$$u_i[j] \in \mathcal{U}_i, \quad \forall j \in [k : k+N_\mathrm{p}-1], \tag{24d}$$

where ρ_i^s is the probability of occurrence of each scenario s, which depends on how frequently it appears in the scenario database. Hence, $\sum_{s=0}^{N_s} \rho_i^s = 1$. The set of scenarios for each controller under the defense mechanism is formulated as

$$\left\{\lambda_i^0[k : k+N_\mathrm{p}-1]\right\} \cup \widehat{\Lambda}_i[k],$$

where $\lambda_i^0[k : k+N_\mathrm{p}-1]$ results from the price update at each *inner* iteration of the dual decomposition DMPC algorithm. Therefore, the negotiation process when the defense mechanism is active combines at each *inner* iteration the current value of the sequence λ_i^0 and the trustworthy price scenarios. Note that the combination is weighted and the weights need not be constant, i.e., they could be updated at each time step to allow for the modulation of the defense mechanism. Therefore, the weight ρ_i^s assigned to the sequence of current prices $\lambda_i^0[k : k+N_\mathrm{p}-1]$ can be reduced as long as abnormal behavior persists or restored as current prices converge again to their expected values. In particular, $\rho_i^s = 0$ reflects that current prices cannot be trusted whereas $\rho_i^s = 1$ indicates that the coordination system is working as expected. In this work, we consider an abrupt transition of ρ_i^s between the previously mentioned extreme cases once an attack is detected, but smoother update rules are also possible.

Table 1
Parameters of the Plant.

Parameter	Value	Parameter	Value
h_1^0	0.57 m	a_1	1.31×10^{-4} m^2
h_2^0	0.62 m	a_2	1.507×10^{-4} m^2
h_3^0	0.58 m	a_3	9.627×10^{-5} m^2
h_4^0	0.64 m	a_4	8.31×10^{-5} m^2
q_A^0	1.63 m^3/h	γ_a	0.30
q_B^0	2.00 m^3/h	γ_b	0.40
$S_{\{1,2,3,4\}}$	0.03 m^2		

Remark 3. The scenarios in $\hat{\Lambda}_i[k : k+N_\mathrm{p}-1]$ are based on trustworthy historical data obtained in similar situations. For this reason, the set $\hat{\Lambda}_i[k]$ may change with time. How to retrieve the most appropriate $\widehat{\Lambda}_i[k]$ for the current state of the system is a matter of future study. At this moment, the topic goes beyond the scope of this work.

Remark 4. The probability for each scenario can be modified depending on how much the local agent suspects that it is under attack. Each agent can learn based on previous behaviors and when it has been attacked. In this manner, the confidence of the information from its neighbors is reduced in the event of an attack. Otherwise, it is increased. For example, $\rho_i^0 = 0$ if there is no confidence in the coordination process.

Remark 5. The defense mechanism is based on prices, which capture the interlocking influence from the rest of the system. From this viewpoint, prices may be abnormal due to the influence of more than one agent. The only issue here comes if one agent modifies its constraints, but even this can be mitigated by switching to a decentralized operation based on historical prices (e.g., if the probability of the current scenario becomes zero).

5. Theoretical and algorithmic properties of the defense mechanism

The proposed method has been designed as a defense mechanism to be used when attacks are detected. In this section, we briefly discuss the consequences that follow from its utilization from a theoretical viewpoint.

In the first place, we recall that dual decomposition is a coordination mechanism that allows solving a centralized control problem in a distributed fashion. In particular, it provides a way of decoupling shared constraints so that the computation of the solution of the control problem can be performed by different entities in the control network. Hence, any theoretical property such as stability or robustness that needs to be guaranteed must be derived from the optimization problem being distributed. Nevertheless, any modification of the prices with respect to their optimal values makes the distributed solution different from the centralized one. For this reason, the impact of the defense mechanism is as follows:

- Feasibility of the local optimization problems: prices enter as additional linear terms on the cost of the optimization problem. Hence, they do not affect the domain of the value function of the local control law. For this reason, the feasibility of local quadratic programming problems is not endangered by the use of the defense mechanism.

- Convergence of the DMPC algorithm: this property depends on the update of the prices, which is performed by means of a gradient search. However, to ensure convergence, the probability ρ_i^0 assigned to $\lambda_i^0[k : k+N_\mathrm{p}-1]$ has to be greater than zero. Note that if $\rho_i^0 = 0$, the agent using the defense mechanism becomes *disconnected* from the updates of the prices. This will steer the coupled variables towards the values set by the defender. Hence, if the attacker is using a fake constraint set that excludes the values of the shared variables used by the defender, convergence will be lost and the algorithm will terminate when the maximum number of iterations is reached. In order to avoid issues in this regard, we will assume that all coupled variables are subject to the same set of constraints and $\rho_i^0 > 0$.
- Coupled constraints satisfaction: in dual decomposition, whenever the iterations are stopped before convergence has been attained, the value of the local variables related with the coupling constraints may differ. In other words, iterates need not be feasible regarding the satisfaction of coupling constraints. In this situation, a projection onto the feasible set to fulfill these constraints is necessary. Given that dual decomposition typically converges slowly, it is not rare that iterations have to be stopped prematurely due to the timing constraints imposed by the sampling rate. Here, the same situation may occur and the same solutions used in standard dual decomposition have to be employed, e.g., a coordinator layer makes the projection, agents agree to implement a mean value of their shared variables, and so on. Nevertheless, note that there is no problem regarding the satisfaction of coupled constraints when convergence is attained. Hence, from a practical viewpoint, there is no impact of the defense mechanism in this regard.
- Local constraints satisfaction: these constraints are satisfied because they depend only on local optimization variables. Given that the local problems are feasible, they will be satisfied.
- Stability: this property is inherited from the centralized control problem being distributed. Depending on how it is achieved, the attacks and the defense mechanism may have an impact on it. In particular, there is a high risk of losing the stability guarantees of the DMPC scheme if the property depends on the fulfillment of coupled constraints, as was shown before, e.g., a terminal region/terminal cost approach. In the case where the satisfaction of this property is critical, it is advisable to use robust formulations of the local controllers with respect to the value of the shared variables. Also, the use of a coordination layer or a supervisor may help to guarantee stability.
- Optimality: given that the distributed solution is not computed with optimal price values, there is a loss of optimality with the defense mechanism. Nevertheless, note that the attack generates a loss of optimality as well, so this property was lost in any case. The loss of optimality can be quantified. To this end, let us define

$$g_i(\lambda_i) = \min_{u_i[k:k+N_\mathrm{p}-1]} \sum_{j=k}^{k+N_\mathrm{p}-1} \left(\ell_i(x_i[j+1], u_i[j]) + \lambda_i[j]^T C_i u_i[j]\right) \tag{25}$$

subject to (11)(b)–(11)(d), where $\lambda_i[j]$ is the price used by agent i at time j. Here, we take

$$\lambda_i[j] = \sum_{l=0}^{N_s} \rho_i^s \lambda_i^l[j] \tag{26}$$

if agent i is implementing the defense mechanism. Note that the attacker may be using fake prices. It can be easily checked that the following inequalities must hold:

$$\sum_{i=0}^{N} g_i(\lambda_i[j]) \geq \sum_{i=0}^{N} g_i(\hat{\lambda}_i[j]) \geq \sum_{i=0}^{N} g_i(\lambda_i^*[j]) \geq \sum_{i=0}^{N} g_i(\mathbf{0}), \tag{27}$$

where $\lambda_i^*[k]$ is the optimal price obtained by standard dual decomposition when no attack is performed and $\hat{\lambda}_i[k]$ are nominal values according to (18). As can be seen, the global cost in case of attack is greater than the nominal cost (according to the trigger for the activation of the defense mechanism) and the optimal global cost, which in turn is greater than the sum of local costs when they are optimized selfishly. Note that the first, second, and fourth expressions in (27) can be easily calculated during the iterations of the DMPC scheme. Hence, (27) provides us with a means to calculate a suboptimality bound of the dual decomposition scheme when the defense mechanism is implemented.

6. Case study and results

In this section, we present a case study based on experiments with the four tanks system (Johansson, 2000), which can be partitioned into two subsystems. The case study is used to test the effects of attacks with standard dual decomposition DMPC and the aforementioned scenario based approaches to robustify the overall system. We must remark that the attacks and defense mechanisms applied are only given for illustration purposes. The attacks are relatively simple and have not been optimized by any means. The same holds for the defense mechanism proposed. Instead, we have used a reduced set of the scenarios to show the effectiveness of the defense mechanism even if it is applied empirically.

6.1. Description of the four tank plant

The four tank plant used consists of a modification to the four interconnected water tanks presented by Johansson (2000) to test control techniques using industrial-type instrumentation and control systems (Alvarado et al., 2011). A picture of the plant used in our experiment is shown in Fig. 3a. A plant diagram can be seen in Fig. 3b. It is composed of four tanks: two top tanks (3 and 4), which discharge into two bottom ones (1 and 2). Each tank is filled with the flow from a storage tank situated at the bottom of the plant by two pumps (q_A and q_B). The input flows are regulated by three-way valves.

The system is modeled as follows:

$$\frac{dh_1}{dt} = -\frac{a_1}{S_1}\sqrt{2gh_1} + \frac{a_3}{S_3}\sqrt{2gh_3} + \frac{\gamma_a}{S_1}q_A, \tag{28}$$

$$\frac{dh_2}{dt} = -\frac{a_2}{S_2}\sqrt{2gh_2} + \frac{a_4}{S_4}\sqrt{2gh_4} + \frac{\gamma_b}{S_2}q_B, \tag{29}$$

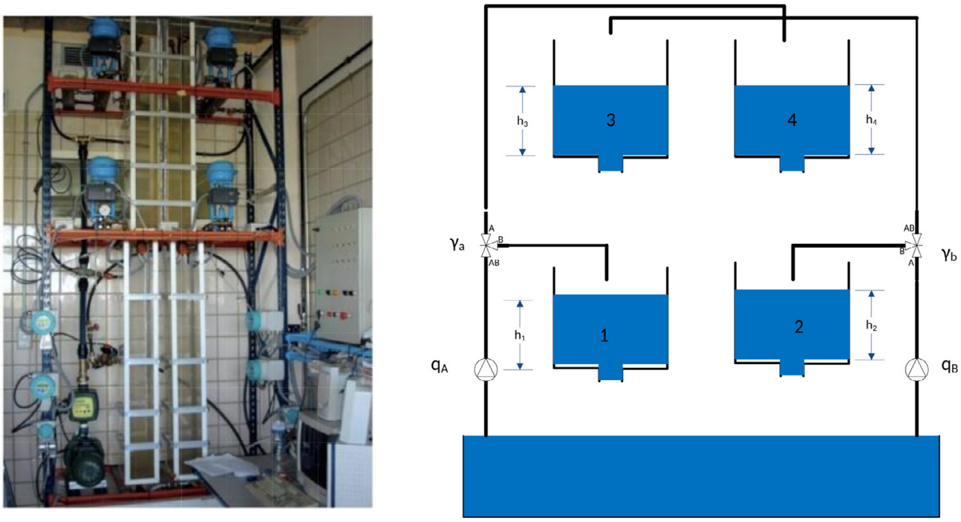
$$\frac{dh_3}{dt} = -\frac{a_3}{S_3}\sqrt{2gh_3} + \frac{(1-\gamma_b)}{S_3}q_B, \tag{30}$$

$$\frac{dh_4}{dt} = -\frac{a_4}{S_4}\sqrt{2gh_4} + \frac{(1-\gamma_a)}{S_4}q_A, \tag{31}$$

where x_i is the flow level of each ith tank, S_i is the cross section, a_i is the discharge constant, g is the value of the gravity, and γ_j is the ratio of the three-way jth valve, with $j \in \{a, b\}$. The values of these parameters are given in Table 1.

The centralized discrete-time linear time-invariant model was obtained at operating point (q_A^0, q_B^0, and h_i^0), with a sample time of 5 s. The system is defined as

$$x[k+1] = \begin{bmatrix} 0.9325 & 0 & 0.0419 & 0 \\ 0 & 0.9297 & 0 & 0.0390 \\ 0 & 0 & 0.9566 & 0 \\ 0 & 0 & 0 & 0.9595 \end{bmatrix} x[k] + \begin{bmatrix} 0.0134 & 0.0006 \\ 0.0006 & 0.0179 \\ 0 & 0.0272 \\ 0.0317 & 0 \end{bmatrix} u[k], \tag{32}$$



(a)

(b)

Fig. 3. The four tank system.

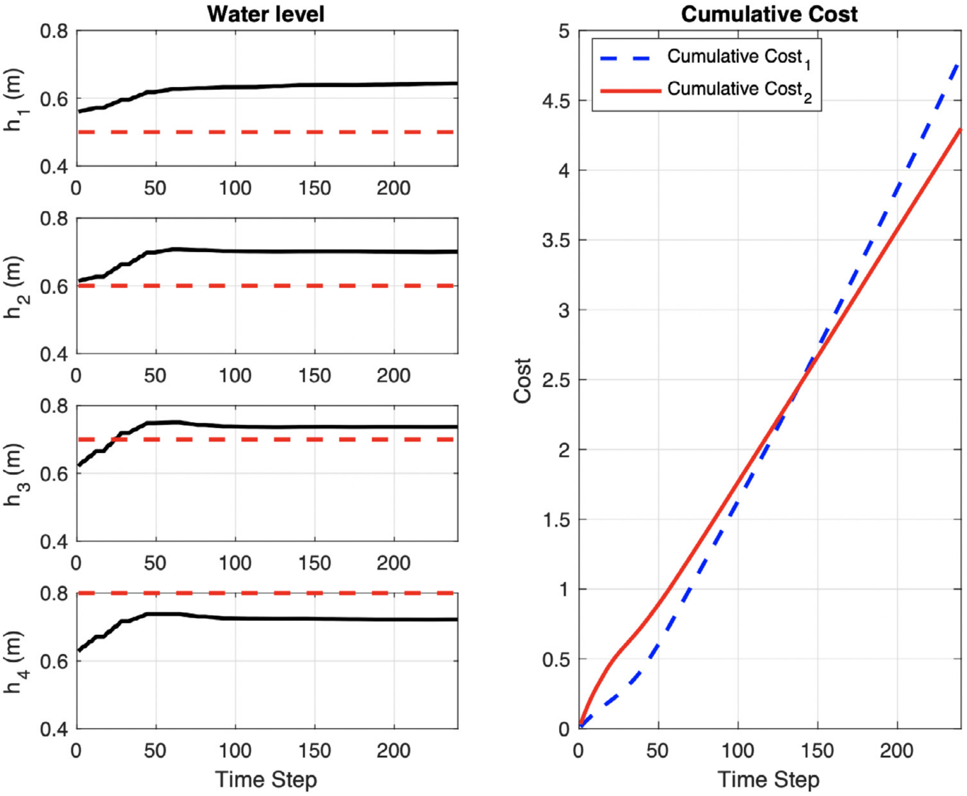


Fig. 4. Water levels and cumulative costs by using standard DMPC approach.

where $x_i[k] = h_i[k] - h_i^0$ for $i \in \{1,2,3,4\}$ and $u_j[k] = q_j[k] - q_j^0$ for $j \in \{A,B\}$, for each time instant $k \in \mathbb{Z}_+$. Also, Table 1 shows the parameters employed in the water-tank process.

In order to ensure the correct performance of the plant and its equipment, the system is subject to constraints given by

$$0.2\,\mathrm{m} \le h_1[k],\ h_3[k] \le 1.36\,\mathrm{m}, \tag{33a}$$

$$0.2\,\mathrm{m} \le h_2[k],\ h_4[k] \le 1.36\,\mathrm{m}, \tag{33b}$$

$$0\,\mathrm{m}^3/\mathrm{h} \le q_A[k] \le 3.26\,\mathrm{m}^3/\mathrm{h}, \tag{33c}$$

$$0\,\mathrm{m}^3/\mathrm{h} \le q_B[k] \le 4\,\mathrm{m}^3/\mathrm{h}. \tag{33d}$$

The optimization problem for this system is given by

$$J(x[k], u[k]) = \tag{34}$$
$$\min_{u[k:k+N_\mathrm{p}-1]} \sum_{l=k}^{k+N_\mathrm{p}-1} \left(x[l+1] - x_\mathrm{ref}\right)^T Q \left(x[l+1] - x_\mathrm{ref}\right) + u^T[l] R u[l],$$

subject to (33). Here, x_ref is the given reference level for the state, and $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$ are the weights for the states and the control inputs, respectively.

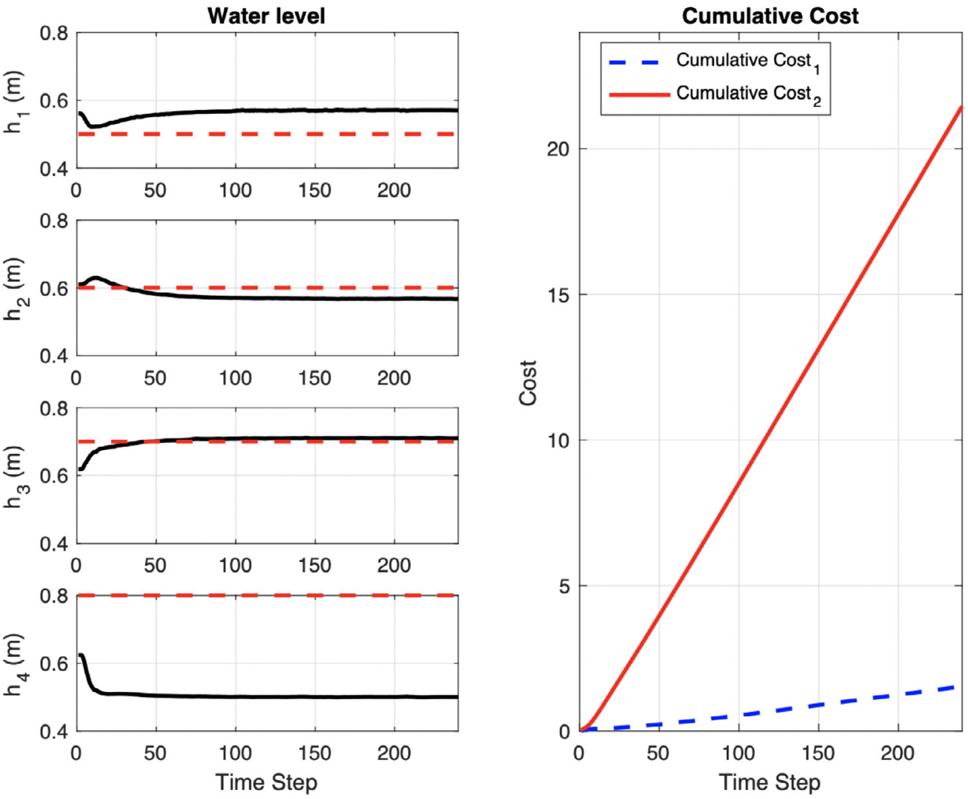


Fig. 5. Water levels and cumulative costs by using a false reference attack.

6.2. Standard dual decomposition DMPC

The overall system can be decomposed into two input-coupled subsystems following the partitioning proposed in Alvarado et al. (2011), which accounts for its structure and the coupling between subsystems when pumps are activated. In particular, the first subsystem is composed of tanks 1 and 3, and the second one is composed of tanks 2 and 4. The optimization problem for each subsystem i is expressed as

$$S_i: \quad J_i(x_i, u_i) = \tag{35a}$$

$$\min_{u_i[k:k+N_\mathrm{p}-1]} \sum_{l=k}^{k+N_\mathrm{p}-1} \left(x_i[l+1] - x_{i\mathrm{ref}}\right)^T Q_i \left(x_i[l+1] - x_{i\mathrm{ref}}\right) + u_i^T[l] R_i u_i[l],$$

subject to

$$x_i[l+1] = A_i x[l] + B_i u_i[l], \ \forall l \in [k : k + N_\mathrm{p} - 1], \tag{35b}$$

$$x_i[l] \in \mathcal{X}_i, \ \forall l \in [k : k + N_\mathrm{p} - 1], \tag{35c}$$

$$u_i[l] \in \mathcal{U}_i, \ \forall l \in [k : k + N_\mathrm{p} - 1]. \tag{35d}$$

Also, both subsystems are subject to the coupling constraint

$$u_1[l] = u_2[l], \ \forall l \in [k : k + N_\mathrm{p} - 1]. \tag{36}$$

Here $x_1[k] = [h_i[k] - h_i^0]_{i\in\{1,3\}}$, $x_2[k] = [h_i[k] - h_i^0]_{i\in\{2,4\}}$, and $u_1[k]$ and $u_2[k]$ correspond to $q_A[k] - q_A^0$ and $q_B[k] - q_B^0$, respectively. Moreover, A_1, A_2, B_1, and B_2 are time invariant matrices of the system.

The experiments were carried out by using a prediction horizon of $N_\mathrm{p} = 5$ along a test time of 20 min, i.e., 240 time steps. The reference levels were established as $h_{1\mathrm{ref}} = 0.5$ m, $h_{2\mathrm{ref}} = 0.6$ m, $h_{3\mathrm{ref}} = 0.7$ m, and $h_{4\mathrm{ref}} = 0.8$ m. Moreover, Q and R were respectively set as $I_{2\times2}$ and $0.01 I_{2\times2}$ for both subsystems to prioritize the reference tracking over the control effort, with $I_{2\times2}$ being the unit matrix of corresponding dimension.

In our experiments, the setpoint has been intentionally set to make it difficult for the agents to control the system. Even if no attacks are performed, the control system is unable to reach its target, forcing local controllers to negotiate in order to attain a trade-off between their corresponding goals. In this condition, it is easier to see how one agent can take advantage of the other one by manipulating the negotiation process. Likewise, the four-tank plant is very versatile and the problem setup has been slightly modified so as to increase the coupling between the subsystems. In particular, we have included a reference value for each tank, whereas it is usual to focus only on the two bottom tanks, which are considered as the outputs of the plant (Alvarado et al., 2011; Johansson, 2000).

Remark 6. A reachable setpoint might seem a better choice for the experiments, but it brings several complications. To begin with, what is the selfish incentive that the attacker has to modify its local problem if it is at equilibrium? At equilibrium it reaches the lowest possible cost according to its true preferences, and therefore there is no possibility to be better off by altering its local problem. Clearly, it might attempt to modify the transient so as to attain a better trajectory and accumulated cost as it reaches its target, but this requires dynamically optimizing the attack in a way that is beyond the scope of our work. Another possibility is to consider that the attacker has a purely malicious behavior and desires to disrupt the normal operation of the system, maintaining its attack at all times. While this will surely modify the steady-state values (as it is illustrated for example in Figs. 1 and 2), it deviates from the assumption of rationality in the attacks that is considered in our work. All things considered, and given that the focus of the article is on the effects of the attacks and the defense mechanism, the setpoint was intentionally set unreachable to avoid these issues and make it difficult for the agents to control the system. Nevertheless, if this would not be the case, our experiments show that the system variables of the agents using the defense mechanism will tend to their nominal values.

Fig. 4 shows the results for both agents obtained by using the standard dual decomposition based DMPC. Note that once convergence

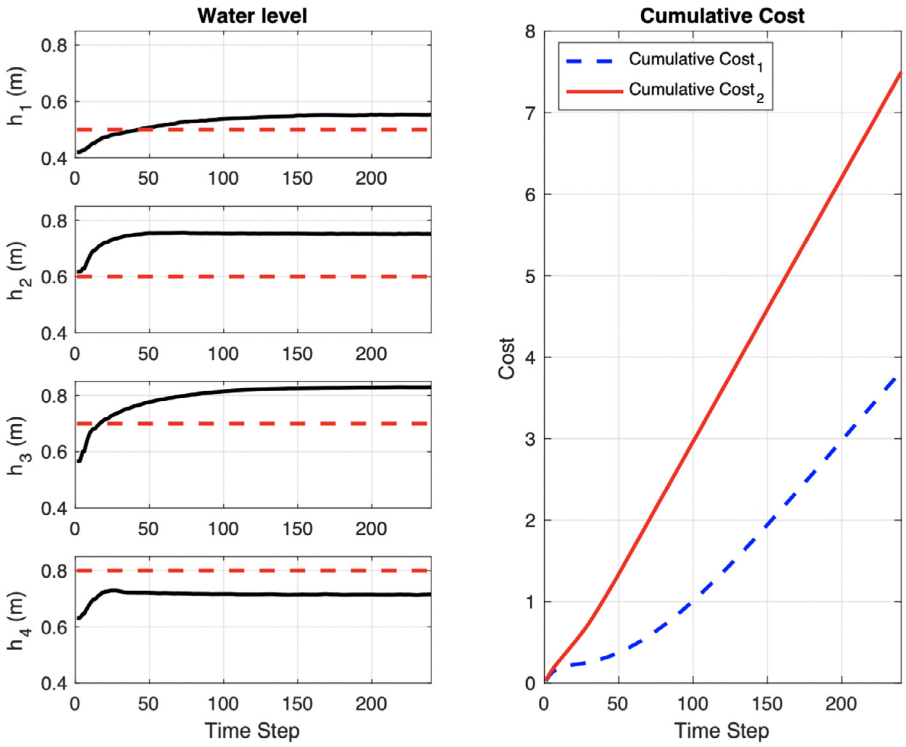


Fig. 6. Water levels and cumulative costs by using a "fake constraints" approach.

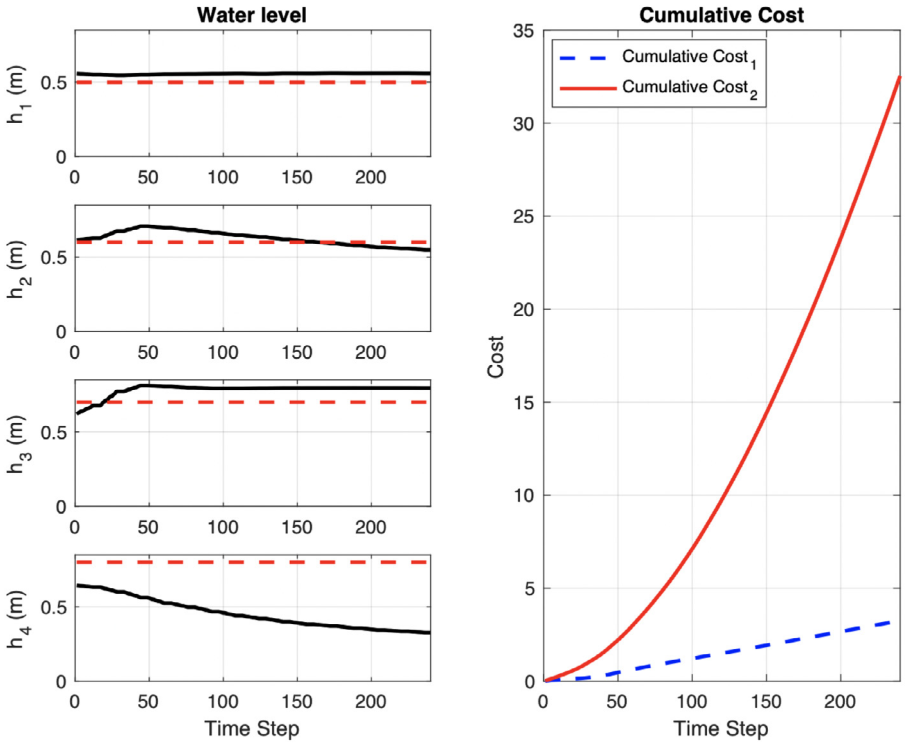


Fig. 7. Water levels and cumulative costs by using a fake prices approach.

is attained, the performance of the distributed control system becomes that of centralized MPC. The water level of each tank and the cumulative cost for each agent are shown in this figure. Notice that the cumulative cost of agent 1 is higher than that of agent 2. The final water levels in each tank were: $h_1 = 0.64$ m, $h_2 = 0.70$ m, $h_3 = 0.73$ m, and $h_4 = 0.72$ m. The red dashed lines represent the given reference levels for each tank.

6.3. Attacks in the control network

Fig. 5 shows the results by carrying out a false reference approach by agent 1. The false reference for the controller 1 was set

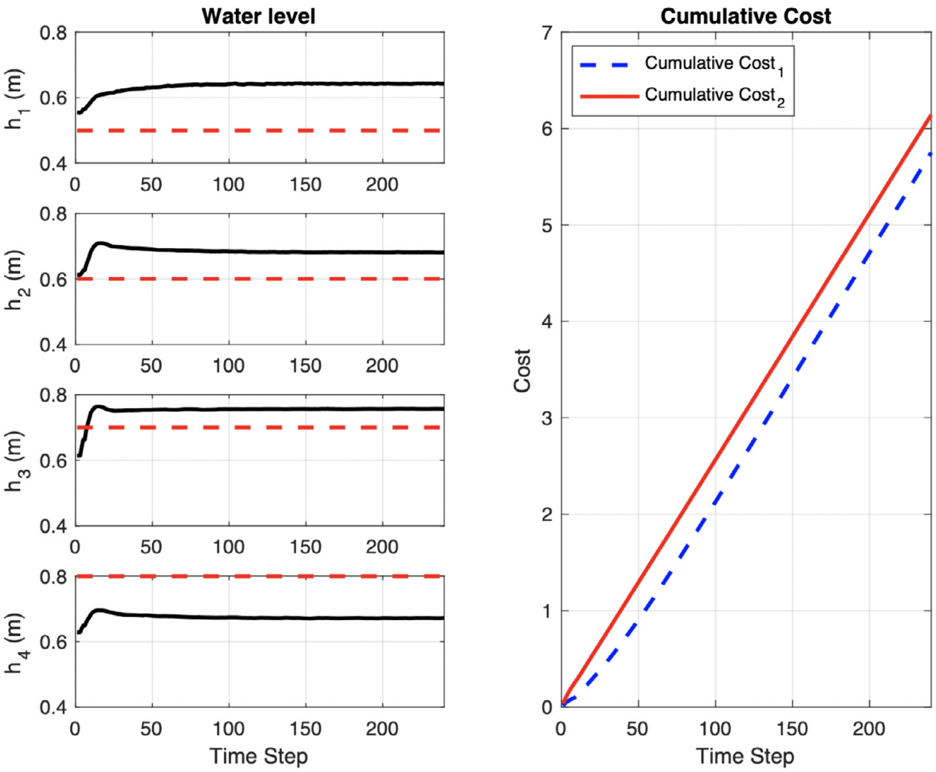


Fig. 8. Water levels and cumulative costs by using a MS-DMPC and fake prices approaches.

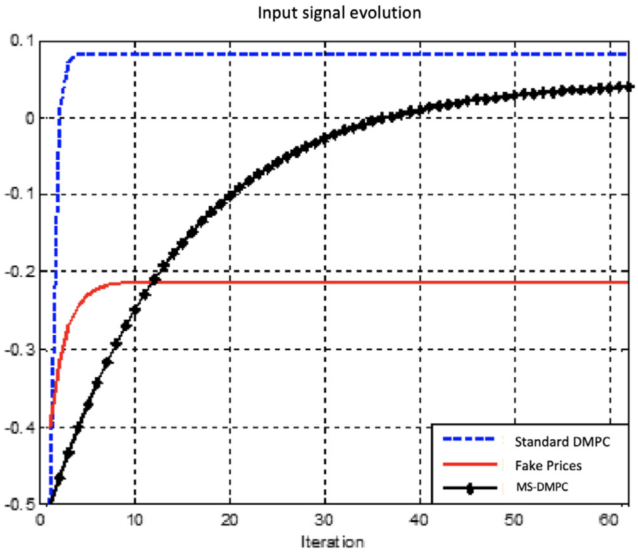


Fig. 9. Input signal evolution at time instant $k = 30$ for standard DMPC, attacker agent, and MS-DMPC during the negotiation process.

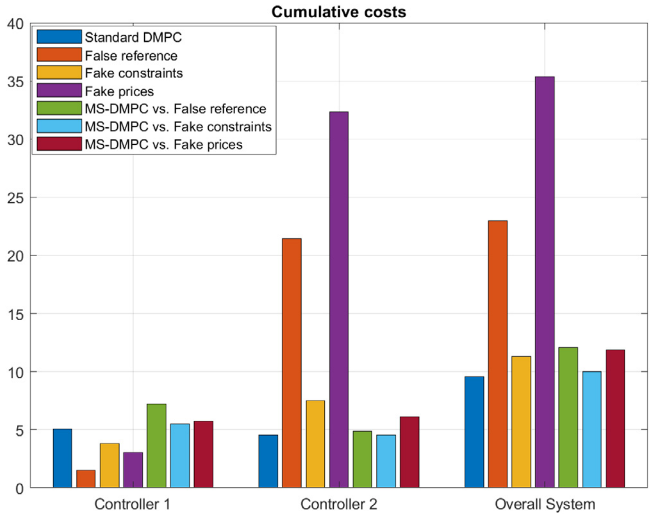


Fig. 10. Cumulative cost by using standard DMPC, attacks, and defense scenario-based methods.

as $[h^*_{1\text{ref}}; h^*_{3\text{ref}}] = [0; 0.58]$. This false reference cheats agent 2 and takes advantage from the coordination algorithm. As can be seen, the cumulative cost for agent 1 is reduced remarkably at expenses of agent 2, which sees its cumulative cost increased. The final water levels for agent 1 are closer to the original references.

The results obtained by using the fake constraints approach are shown in Fig. 6. The fake constraints for control inputs which correspond to the controller 1 were set as

$$u_1 \in \mathcal{U}_1^*, \tag{37}$$

where $\mathcal{U}_1^* = 0.5 \times \mathcal{U}_1$. As can be seen, the optimization process is affected and reduces the cost for agent 1. The water level that corresponds to tank 1 is closer to the given reference level ($|h_1 - h_{1\text{ref}}| = 0.039$ m) compared to the standard DMPC ($|h_1 - h_{1\text{ref}}| = 0.124$ m). In this way, controller 1 takes advantage of the negotiation process and obtains benefits at the cost of performance losses in the other controller.

Fig. 7 shows the water levels of each tank and the cumulative cost for both subsystems by carrying out a fake prices approach. This attack was performed by using the coefficient $\alpha = 10$ into the optimization problem of controller 1. Notice that the cumulative cost for agent 2 is higher than that obtained by the attacker. Also, the water levels in tank 4 is farther from its original reference.

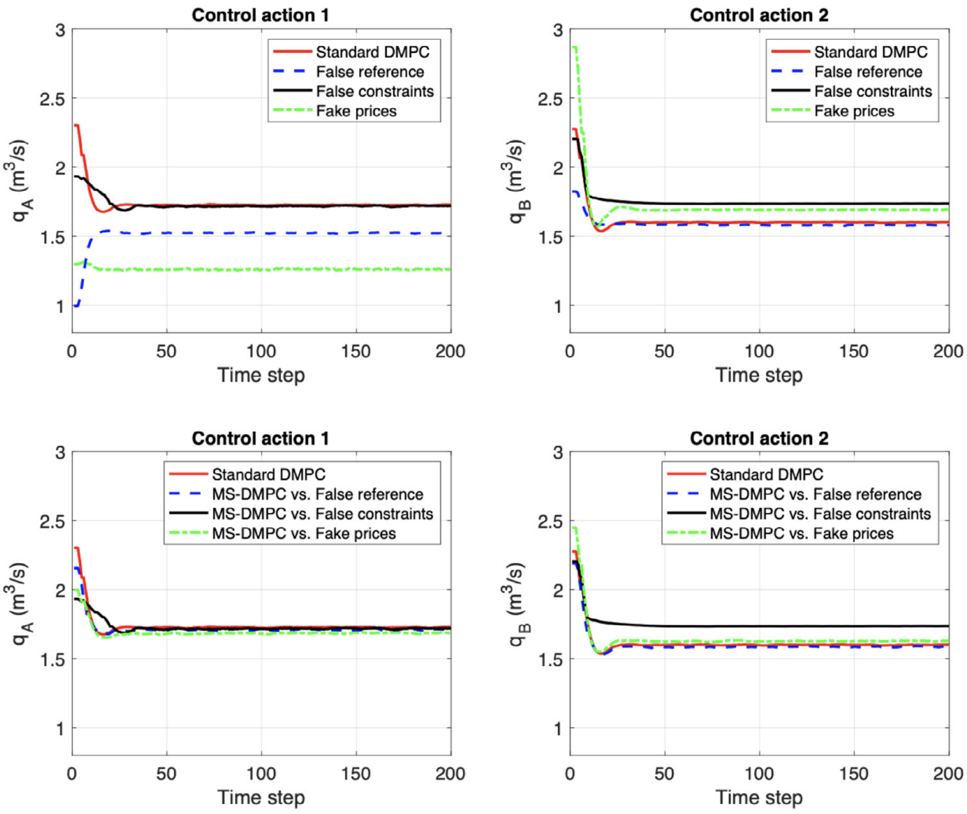


Fig. 11. Evolution of control actions using standard DMPC, attacks, and defense scenario-based methods.

As it has been shown, the malicious controller always increases the cumulative cost of the second agent. The malicious modifications of the first agent's problem lead to a new minimizer, generating an increase of costs with respect to the original overall cost function. Nevertheless, the net impact on the trajectory of each specific variable of the second agent is uncertain, and some of them might even *improve* despite the increase in local costs. For example, Figs. 5 and 7 show that under the false reference and fake prices attacks, the trajectory of the level of tank 2 reduces its steady-state error, while in tank 4, it is increased. This results in a higher cumulative cost of agent 2 due to the effect of the attacker. Moreover, in a problem with multiple agents, it might even happen that some subsystems different from the malicious one decrease their costs despite the increase of overall cost. Therefore, while the malicious agent will degrade global performance pursuing its self interest, it is necessary to evaluate carefully its effect in the rest of the subsystem's variables and costs.

6.4. Defense mechanism

In this subsection, the scenario defense mechanism described in Section 4 is applied to obtain a certain robustness in the attacked agent. In particular, agent 2 carries out the optimization problem by considering a collection of $N_s = 9$ trustworthy price scenarios obtained from historical data, where each scenario has a probability of occurrence of $\rho^s = 1/10$. Also, the corresponding prices have been disturbed by a Gaussian noise with mean and standard deviation values being 0.087 and 0.0045, respectively. Fig. 8 shows the water levels for each tank and the cumulative costs when agent 2 uses the MS-DMPC mechanism to reduce the impact of the fake prices attack of agent 1. In comparison to the vulnerable case in Fig. 7, the cumulative costs and plots are closer to those obtained under a reliable negotiation process in Fig. 4.

To highlight the advantages of the MS-DMPC approach behind the consensus process, Fig. 9 shows the evolution of the input signal for the standard DMPC, the attack performed by fake prices attack, and the proposed MS-DMPC mechanism at time instant $k = 30$. At this time step, the negotiation between the controllers to attain an agreement regarding the value of the control action converges after 63 iterations. As can be seen in the figure, the agreed value can be steered by the attacker. However, the MS-DMPC mechanism creates a consensus close to that under the normal operation.

Fig. 10 shows the cumulative cost for each agent and the cumulative cost of the overall system for the attack schemes and the results obtained with the MS-DMPC. Notice that the consequences of the attacks regarding cumulative costs are reduced. In this sense, agent 1 increases its cumulative cost while the cumulative cost function of agent 2 is reduced remarkably compared with the cumulative costs obtained under attack.

Fig. 11 presents the evolution of the control signals during the different situations assessed. In particular, the upper row shows how control signals 1 and 2 deviate from standard operation in case of attack. The lower row presents the results when the proposed heuristic defense mechanism is applied. Taking into account that the control actions are the outcome of the distributed method – they are generated from the minimizer of the overall constrained optimization problem –, it is clear that larger deviations happen when no defensive measurements are implemented. These effects are evident both during the initial transient phase and the steady-state values that result from the negotiation between local controllers. In particular, only the fake constraints attack seems to succeed in steering control signal 2 in both situations. However, this is no surprise given the previously explained reasons, i.e., active constraints *limit* the value that control signals can have during the negotiation.

Regarding the moment when the proposed defensive mechanism is implemented, an attack detection trigger can be activated using the suboptimality bounds of the dual decomposition scheme in (27). In this regard, costs are very revealing. As shown by Fig. 10, the cumulative cost of the overall system when using standard DMPC is 9.59, whereas this value increases to 35.39 when the system is under a false price attack. Once this abnormal behavior is detected, the defensive method is able to mitigate the harm, decreasing the cumulative cost to 11.89.

7. Conclusions and future directions

In this paper, cyber-security issues in DMPC have been considered. An analysis of the vulnerability of a popular Lagrange-based DMPC scheme has been presented. In particular, we have shown how a malicious agent can use false information in its local optimization problem or in the information exchange to steer the coordination when dual decomposition is used. By a case study involving a real distributed system as the four tank plant, we have carried out and illustrated the potential of these attack mechanisms. We have also addressed the problem of providing robustness to DMPC for defending it from a malicious controller by implementing scenario-based mechanisms. Moreover, we have also analyzed the main theoretical and algorithmic properties of dual decomposition when this defense mechanism is applied.

From a local viewpoint, prices capture the interlocking influence of the rest of the system and are the means through which coordination is canalized. The proposed defense mechanism is triggered by the detection of abnormal performance and/or prices for the current state of the system. After an attack has been detected, the defense method starts employing price scenarios from previous executions of the system to mitigate its impact. To this end, current prices are weighted and combined with historical price scenarios. Moreover, it is also possible to have local subsystems working solely based on price scenarios to obtain a *nominal* level of coordination, thus ignoring the attempts of the malicious agent to steer the negotiation. Therefore, the proposed method has the potential to deal with attacks different than those that have been explicitly considered in our experiments. For example, it could handle a situation in which the attacker combines two of the considered attacks (e.g., fake reference and fake prices) and also scenarios with more than one malicious subsystem.

Finally, we have considered in this work the case of insider attacks, i.e., the case where one agent within the distributed system becomes malicious. We believe that the proposed defense mechanism might deal with other type of attacks as well, e.g., jamming attacks that disrupt the communication between local controllers, but this topic deserves careful research to define first the main attacks of interest and later on a proper benchmark with appropriate key performance indicators. Indeed, this is a very relevant topic that has not been explicitly addressed in the DMPC literature in a systematic and structured way. Therefore, we plan to explore in the future the mitigation power of the proposed method against other common cyberattacks and also in different situations, e.g., systems with many agents that include several attackers. Likewise, it will be interesting to confirm whether the defense mechanism implemented in one agent can *protect* other affected subsystems that are not using this method. Furthermore, the regulation of the parameter ρ_i^l as an index of trustworthiness of each scenario will be investigated. It can be considered as a confidence index given by the neighbors based on their experiences in previous consensus processes.Likewise, future research will focus on formalizing more attack models and defense mechanisms, analyzing their effects, and considering other case studies in the domains of power systems and logistics.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The financial supports from project GESVIP, Spain (ref. US-1265917) and JSPS, Japan under Grant-in-Aid for Scientific Research (B) Grant No. 18H01460 are gratefully acknowledged.

References

Albright, D., Brannan, P., & Walrond, C. (2010). *Did stuxnet take out 1,000 centrifuges at the natanz enrichment plant?*. Institute for Science and International Security.

Alvarado, I., Limon, D., Muñoz de la Peña, D., Maestre, J., Ridao, M., Scheu, H., et al. (2011). A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark. *Journal of Process Control*, *21*(5), 800–815.

Ananduta, W., Maestre, J. M., Ocampo-Martinez, C., & Ishii, H. (2020). Resilient distributed model predictive control for energy management of interconnected microgrids. *Optimal Control Applications & Methods*, *41*(1), 146–169.

Barreto, C., Giraldo, J., Cardenas, A. A., Mojica-Nava, E., & Quijano, N. (2014). Control systems for the power grid and their resiliency to attacks. *IEEE Security & Privacy*, *12*(6), 15–23.

Başar, T., & Srikant, R. (2002). A stackelberg network game with a large number of followers. *Journal of Optimization Theory and Applications*, *115*(3), 479–490.

Biegel, B., Stoustrup, J., & Andersen, P. (2014). Distributed MPC via dual decomposition. *Distributed model predictive control made easy*, (pp. 179–192). Springer.

Bindra, A. (2017). Securing the power grid: Protecting smart grids and connected power systems from cyberattacks. *IEEE Power Electronics Magazine*, *4*(3), 20–27.

Braun, S., Albrecht, S., & Lucia, S. (2020). Hierarchical attack identification for distributed robust nonlinear control. In *Proceedings of the 21st ifac world congress*, Berlin, Germany (pp. 6191–6198).

Calafiore, G. C., & Fagiano, L. (2013). Stochastic model predictive control of LPV systems via scenario optimization. *Automatica*, *49*(6), 1861–1866.

Camacho, E. F., & Bordons, C. (2004). *Model predictive control* (2nd ed.). London, England: Springer-Verlag.

Chakhchoukh, Y., & Ishii, H. (2015). Coordinated cyber-attacks on the measurement function in hybrid state estimation. *IEEE Transactions on Power Systems*, *30*(5), 2487–2497.

Chanfreut, P., Maestre, J. M., & Ishii, H. (2018). Vulnerabilities in distributed model predictive control based on jacobi-gauss decomposition. In *Proceedings of the 2018 european control conference (ECC)*, Limassol, Cyprus (pp. 2587–2592).

Chen, S., Wu, Z., & Christofides, P. D. (2020). Cyber-attack detection and resilient operation of nonlinear processes under economic model predictive control. *Computers & Chemical Engineering*, *136*, Article 106806.

Cheng, P., Shi, L., & Sinopoli, B. (2017). Guest editorial special issue on secure control of cyber-physical systems. *IEEE Transactions on Control of Network Systems*, *4*(1), 1–3.

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

Darup, M. S., Redder, A., Shames, I., Farokhi, F., & Quevedo, D. (2017). Towards encrypted MPC for linear constrained systems. *IEEE Control Systems Letters*, *2*(2), 195–200.

Dhal, R., & Roy, S. (2013). Vulnerability of continuous-time network synchronization processes: A minimum energy perspective. In *Proceedings of the 52nd IEEE conference on decision and control*, Florence, Italy (pp. 823–828).

Giselsson, P., & Rantzer, A. (2014). Generalized accelerated gradient methods for distributed MPC based on dual decomposition. In *Distributed model predictive control made easy* (pp. 309–325). Springer, (chapter 11).

Grosso, J. M., Velarde, P., Ocampo-Martinez, C., Maestre, J. M., & Puig, V. (2017). Stochastic model predictive control approaches applied to drinking water networks. *Optimal Control Applications & Methods*, *38*(4), 541–558.

Ishii, H., Yoshizawa, S., Fujimoto, Y., Ono, I., Onoda, T., & Hayashi, Y. (2020). Cyber security for voltage control of distribution systems under data falsification attacks. In *Design and analysis of distributed energy management systems* (pp. 145–165). Springer.

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.

Kritzinger, E., & Von Solms, S. (2010). Cyber security for home users: A new way of protection through awareness enforcement. *Computers & Security*, *29*(8), 840–847.

Kushner, D. (2013). The real story of stuxnet. *IEEE Spectrum*, *3*(50), 48–53.

Larsen, G., Van Foreest, N., & Scherpen, J. (2014). Distributed MPC applied to a network of households with micro-CHP and heat storage. *IEEE Transactions on Smart Grid*, *5*(4), 2106–2114.

Li, L., Negenborn, R. R., & De Schutter, B. (2015). Intermodal freight transport planning - a receding horizon control approach. *Transportation Research Part C (Emerging Technologies)*, *60*, 77–95.

Li, L., Negenborn, R. R., & De Schutter, B. (2017). Distributed model predictive control for cooperative synchromodal freight transport. *Transportation Research Part E: Logistics and Transportation Review*, *105*, 240–260.

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

Maestre, J. M., & Negenborn, R. R. (2014). *Distributed model predictive control made easy*. Springer.

Maestre, J. M., Trodden, P. A., & Ishii, H. (2018). A distributed model predictive control scheme with robustness against noncompliant controllers. In *Proceedings of the 57th IEEE conference on decision and control (CDC)*, Miami, Florida, USA (pp. 3704–3709).

Mc Namara, P., Negenborn, R. R., De Schutter, B., Lightbody, G., & McLoone, S. (2016). Distributed MPC for frequency regulation in multi-terminal HVDC grids. *Control Engineering Practice*, *46*, 176–187.

Mercader, P., & Haddad, J. (2021). Resilient multivariable perimeter control of urban road networks under cyberattacks. *Control Engineering Practice*, *109*, Article 104718.

Miao, F., Zhu, Q., Pajic, M., & Pappas, G. J. (2016). Coding schemes for securing cyber-physical systems against stealthy data injection attacks. *IEEE Transactions on Control of Network Systems*, *4*(1), 106–117.

Nabais, J., Negenborn, R. R., Carmona, R. B., & Ayala, M. (2015). Achieving transport modal split targets at intermodal freight hubs using a model predictive approach. *Transportation Research Part C (Emerging Technologies)*, *60*, 278–297.

Negenborn, R. R., & Maestre, J. M. (2014). Distributed model predictive control: An overview and roadmap of future research opportunities. *IEEE Control Systems*, *34*(4), 87–97.

Olivares, D. E., Lara, J. D., Cañizares, C. A., & Kazerani, M. (2015). Stochastic-predictive energy management system for isolated microgrids. *IEEE Transactions on Smart Grid*, *6*(6), 2681–2693.

Onogawa, M., Yoshizawa, S., Fujimoto, Y., Ishii, H., Ono, I., Onoda, T., et al. (2019). Enhancing security for voltage control of distribution systems under data falsification attacks. In *Proceedings of the 2019 american control conference (ACC)*, Philadelphia, USA (pp. 3249–3254).

van Overloop, P. J., Weijs, S., & Dijkstra, S. (2008). Multiple model predictive control on a drainage canal system. *Control Engineering Practice*, *16*(5), 531–540.

Pierron, T., Árauz, T., Maestre, J., Cetinkaya, A., & Maniu, C. S. (2020). Tree-based model predictive control for jamming attacks. In *Proceedings of the 2020 european control conference (ECC)*, St. Petersburg, Russia (pp. 948–953).

Radvanovsky, R., & Brodsky, J. (2016). *Handbook of scada/ control systems security*. CRC Press.

van Riessen, B., Negenborn, R. R., Lodewijks, G., & Dekker, R. (2015). Impact and relevance of transit disturbances on planning in intermodal container networks using disturbance cost analysis. *Maritime Economics & Logistics*, *17*, 440–463.

Romagnoli, R., Krogh, B. H., & Sinopoli, B. (2019). Design of software rejuvenation for CPS security using invariant sets. In *Proceedings of the american control conference (ACC)*, Philadelphia, USA (pp. 3740–3745).

Schildbach, G., & Morari, M. (2015). Scenario MPC for linear time-varying systems with individual chance constraints. In *Proceedings of the 2015 american control conference*, Chicago, Illinois (pp. 415–421).

Sheng, S., Chan, W., Li, K., Xianzhong, D., & Xiangjun, Z. (2007). Context information-based cyber security defense of protection system. *IEEE Transactions on Power Delivery*, *22*(3), 1477–1481.

Soudbakhsh, D., Chakrabortty, A., & Annaswamy, A. M. (2017). A delay-aware cyber-physical architecture for wide-area control of power systems. *Control Engineering Practice*, *60*, 171–182.

Sun, Q., Zhang, K., & Shi, Y. (2019). Resilient model predictive control of cyber–physical systems under dos attacks. *IEEE Transactions on Industrial Informatics*, *16*(7), 4920–4927.

Teixeira, A., Sou, K. C., Sandberg, H., & Johansson, K. H. (2015). Secure control systems: A quantitative risk management approach. *IEEE Control Systems*, *35*(1), 24–45.

Tian, X., Guo, Y., Negenborn, R. R., Wei, L., Lin, N. M., & Maestre, J. M. (2019). Multi-scenario model predictive control based on genetic algorithms for level regulation of open water systems under ensemble forecasts. *Water Resources Management*, *33*(9), 3025–3040.

Trodden, P., Maestre, J., & Ishii, H. (2020). Actuation attacks on constrained linear systems: A set-theoretic analysis. In *Proceedings of the 21st IFAC world congress*, Berlin, Germany (pp. 7045–7050).

Velarde, P., Maestre, J. M., Ishii, H., & Negenborn, R. R. (2017). Scenario-based defense mechanism for distributed model predictive control. In *Proceedings of the 56th IEEE annual conference on decision and control (CDC)*, Melbourne, Australia (pp. 6171–6176).

Velarde, P., Maestre, J. M., Ishii, H., & Negenborn, R. R. (2018). Vulnerabilities in Lagrange-based distributed model predictive control. *Optimal Control Applications & Methods*, *39*(2), 601–621.

Velarde, P., Tian, X., Sadowska, A., & Maestre, J. (2019). Scenario-based hierarchical and distributed MPC for water resources management with dynamical uncertainty. *Water Resources Management*, *33*(2), 677–696.

Wu, Z., Albalawi, F., Zhang, J., Zhang, Z., Durand, H., & Christofides, P. D. (2018). Detecting and handling cyber-attacks in model predictive control of chemical processes. *Mathematics*, *6*(10), 173.

Yang, H., Li, Y., Dai, L., & Xia, Y. (2019). MPC-based defense strategy for distributed networked control systems under dos attacks. *Systems & Control Letters*, *128*, 9–18.

Yushen, L., Shuai, L., Xie, L., & Johansson, K. (2014). A scenario-based distributed stochastic MPC for building temperature regulation. In *Proceedings of the 2014 IEEE international conference on automation science and engineering*, New Taipei, Taiwan (pp. 1091–1096).

Zheng, H., Negenborn, R. R., & Lodewijks, G. (2017). Closed-loop scheduling and control of waterborne AGVs for energy-efficient inter terminal transport. *Transportation Research Part E: Logistics and Transportation Review*, *105*, 261–278.

Zhu, Q., & Basar, T. (2015). Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: Games-in-games principle for optimal cross-layer resilient control systems. *IEEE Control Systems*, *35*(1), 46–65.