Reconfiguration of flow-based networks with back-up components using robust economic MPC[☆]

Carlos Trapiello [a,*], Vicenç Puig [a,b], Gabriela Cembrano [b]

[a] Advanced Control Systems Group, Universitat Politècnica de Catalunya (UPC), Rambla Sant Nebridi 10, Terrassa 08222, Spain
[b] Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, Barcelona 08028, Spain

ARTICLE INFO

Keywords:
Back-up components
System reconfiguration
Flow-based networks
Robust control

ABSTRACT

This paper addresses the post-fault selection of an actuators configuration for flow-based networks with back-up components. The proposed reconfiguration methodology consists of an offline and an online phase. On the one hand, an offline analysis looks for the minimal configurations for which the economic cost of the (best) steady-state trajectory that can be achieved using a robust model predictive control (MPC) policy is admissible. On the other hand, at fault detection time, an online search for the best actuators configuration to cope with the transient induced by the fault is conducted in the superset of each minimal configuration calculated offline. With this strategy, the final new configuration is computed by sequentially solving a set of mixed-integer programs whose constraints are derived from single-layer robust MPC schemes coupled with local controllers designed for the *a priori* minimal configurations identified offline. A portion of a water transport network is used to show the performance the proposed solution.

1. Introduction

Generalized flow-based networks (FNs) model many safety–critical infrastructures such as water distribution networks, power distribution networks, etc [1]. Accordingly, it is of paramount importance the implementation of secure control techniques for this type of systems from both: the design phase, with the installation of redundant physical components; and the operational phase, devising fault-tolerant control (FTC) algorithms that take full advantage of system redundancy to maintain an admissible performance after a fault. Notably, the detection of a component fault in FNs is often followed by the isolation of the affected area to prevent the spread of potentially damaging effects, as well as to repair the faulty elements. In this scenario, response procedures typically consist of a set of heuristic rules where different interventions comprising the activation of back-up elements are carried out, e.g. opening normally-closed valves to distribute the flow through secondary pipes or activating auxiliary pumps to feed certain tanks/storage elements. Consequently, the main aim of this paper is to propose an automated procedure for selecting the appropriate back-up actuators after a component fault (thus selecting a new system configuration) for constrained FNs subject to uncertain periodic disturbances such as exogenous flow demands and periodically varying power prices.

In nominal operation, a large number of solutions use model predictive control (MPC) schemes to control FNs due to their ability to efficiently control complex processes [2,3]. In particular, economic MPC strategies [4] have been presented as a convenient approach to regulate the high economic costs associated with the operation of large-scale FNs [5,6]. Two main economic MPC architectures have been proposed: (I) double-layer schemes composed of an upper layer dynamic real-time optimizer that plans the optimal system steady-state trajectory and a low-level predictive controller that tracks the previous Ref. [7]; (II) single-layer schemes where the economic cost function is included in the computation of the control law, thus allowing to assess the economic cost during the transients [8]. In addition, the periodic nature of the disturbances that normally affected FNs causes that, in some cases, the best way to operate the network is the imposition of a cyclic steady-state operation [9,10].

Concerning the analysis of the post-fault *configuration selection* problem within the automatic control field, the study of the FTC capabilities granted by different configurations of actuators/sensors was mainly developed by Staroswiecki in the context of unconstrained systems [11–13]. Nevertheless, the consideration in this case of constrained systems precludes a direct application of the above techniques, since structural and performance methods must be extended by considering feasibility issues. On the other hand, the system reconfiguration with back-up components has been investigated through the three-tank

☆ This work has been supported by the Margarita Salas grant from the Spanish Ministry of Universities funded by the European Union NexGenerationEU.
* Corresponding author.
E-mail address: carlos.trapiello@upc.edu (C. Trapiello).

benchmark [14], and the different solutions that arose. However, this problem does not consider the selection among several back-up components. The selection of alternative actuators was studied within the hybrid systems framework in [15], whereas in [16] the authors combine the offline test of structural properties with an online mixed-integer program (MIP) in charge of the back-up actuators selection. Other approaches like [17], use a combination of residual-based algorithms and logical calculi in the reconfiguration of cyber–physical systems. Nonetheless, none of the above references explicitly considers system uncertainty in the configuration selection.

Notably, the post-fault selection of a new admissible configuration in a system with back-up components should not only provide certain stability and performance guarantees, but it should also be performed under some optimality criterion like, for example, minimizing the number of alternative elements to be activated. Consequently, the problem under investigation extends the standard reconfiguration problem in FTC (cf. [18, Chapter 8]) by including the set of healthy components that are used to reconfigure the system as a problem variable that must be optimized. Furthermore, the robust configuration selection for FNs introduces the following additional challenges:

- **C1** The formulation of stability guarantees in the configuration selection procedure. The model inconsistencies provoked by a fault may cause that the explored candidate configurations are not able to reach the steady-state trajectory imposed for the nominal configuration. However, if steady-state first-principle models are used to select the configuration that yields the closest to the nominal steady-state operation, then feasibility problems may arise at the control layer due to the transient induced by the fault.
- **C2** The uncertainty consideration. Robust MPC schemes usually devise a suboptimal control policy through the *a priori* design of a local controller in charge of compensating the effect of uncertainty sources [19,20]. Hence, the prior selection of the set of actuators that are used in the design of a local controller, and thus that will be activated, may have a high impact on the optimality of the selected configuration.
- **C3** The large-scale of FNs precludes an online evaluation of all the possible alternative configurations.
- **C4** The robust control of FNs subject to algebraic equations describing the static relations in the system is still a topic under investigation.

Of special relevance in this paper is the single-layer MPC scheme proposed for tracking in [21], and later adapted to the economic MPC in [22]. In this approach, the system states are extended with the inclusion of a virtual model which is forced to converge towards the best attainable tracking (or economic) objective. In particular, the formulation of the stability ingredients with respect to the virtual model (which can be updated coherently with the plant model in the configuration selection) allows to propose a response methodology that addresses **C1**, and that guarantees that for the new configuration there exists a control policy that steers the system towards its best attainable steady-state trajectory.

The main contribution of this paper is the proposal of a new methodology for the post-fault configuration selection for constrained FNs based on robust economic MPC schemes. In the proposed methodology, a configuration is considered admissible if it allows to robustly steer the system towards a steady-state cyclic trajectory that yields an appropriate economic operation cost. To that end, the monotonicity of the (best) steady-state economic cost associated with the different configurations is exploited for conducting an offline search of the minimal configurations that return an admissible cost. These minimal configurations have a double purpose: design a local controller to compensate the effect of the uncertainty (**C2**); and filter out non-admissible configurations, thus reducing the computational complexity (**C3**). The final new configuration is selected online by sequentially solving an MIP in the superset of each minimal configuration identified offline. The stability of the system operating in the new configuration is achieved by enforcing that the constraints used by a single-layer robust MPC scheme are satisfied at fault detection time. Besides, a novel formulation to include the matrix that distributes the uncertainty in the static equations of the network as an optimization variable in the computation of the steady-state trajectory planner (**C4**) is proposed. A portion of a water transport network is used to show the performance of the proposed solution.

The remainder of this paper is organized as follows: Section 1 is concluded with some common notation. Section 2 introduces the system under study and some preliminary concepts, whereas Section 3 formulates the problem statement. In Section 4, the stability of each configuration is addressed. Section 5 analyses the performance assessment of the different configurations. Section 6 presents the proposed approach for the robust configuration selection. The considered case study is detailed in Section 7. Finally, Section 8 draws the main conclusions of the paper.

1.1. Notation

Bold letters are used to denote a sequence of T values of the signal, that is, $\boldsymbol{x}_T = \{x(0), \ldots, x(T-1)\}$. The set of positive integer numbers including the origin is denoted as $\mathbb{I}_n = \{0, 1, \ldots, n\}$. The Minkowski sum of two sets $\mathcal{X}$ and $\mathcal{Y}$ is defined by $\mathcal{X} \oplus \mathcal{Y} \triangleq \{x + y : x \in \mathcal{X}, y \in \mathcal{Y}\}$; the Pontryagin set difference is defined by $\mathcal{X} \ominus \mathcal{Y} \triangleq \{x : x \oplus \mathcal{Y} \subseteq \mathcal{X}\}$; a zonotopic set $\mathcal{Z} \triangleq \{c + H\xi : \xi \in \mathbb{R}^m, \|\xi\|_\infty \leq 1\}$ with center $c \in \mathbb{R}^n$ and generators matrix $H \in \mathbb{R}^{n\times m}$ is denoted by $\mathcal{Z} = \langle c, H\rangle$. Additionally, $\otimes$ stands for the Kronecker product and $\mathbf{1}_n$ represents a vector of n ones.

2. System definition and preliminary concepts

Let us consider the uncertain control-oriented model of a FN described by the following set of linear discrete difference-algebraic equations [1]

$$x(k+1) = Ax(k) + B\Sigma u(k) + B_d d(k) + B_w w(k), \tag{1a}$$

$$0 = E\Sigma u(k) + E_d d(k) + E_w w(k), \tag{1b}$$

where $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{R}^{n_u}$, $d(k) \in \mathbb{R}^{n_d}$ and $w(k) \in \mathbb{R}^{n_w}$ are the state, input, predicted and unknown disturbance vectors of the system at time $k \in \mathbb{N}$, respectively. System matrices $A, B, B_d, B_w, E, E_d, E_w$ are of suitable dimensions dictated by the network topology.

For any time instant k, system (1) is subject to hard state and control polytopic constraints given by

$$\begin{aligned} \mathcal{X} &= \{x(k) : H_x x(k) \leq h_x\} \subset \mathbb{R}^{n_x}, \\ \mathcal{U} &= \{u(k) : H_u u(k) \leq h_u\} \subset \mathbb{R}^{n_u}, \end{aligned} \tag{2}$$

where H_x, H_u (h_x, h_u) are real matrices (vectors) with dimensions consistent with the number of state and input constraints. The number of equations n_e in (1b) satisfies $n_e < n_u$. Besides, for all k, the unknown disturbance satisfies $w(k) \in \mathcal{W}$ with

$$\mathcal{W} = \langle 0, I_{n_w}\rangle \subset \mathbb{R}^{n_w}. \tag{3}$$

Remark 1. Note that, as long as $w(k)$ is zonotopically bounded, a zero-centered unitary box representation like (3) can be obtained by performing: (I) a change of coordinates that shifts the uncertainty center to the zero; (II) a coherent modification of the matrices (B_w, E_w).

Assumption 1. The following conditions are satisfied:

(a) The states in $x(k)$ are observable at time instant k.
(b) The input set $\mathcal{U}$ contains the origin, i.e. $0 \in \mathcal{U}$.
(c) The predicted disturbance signal $d(k)$ has a periodic behavior with known period T, i.e. $d(k) = d(k+T)$.
(d) The uncertainty affecting the static Eqs. (1b) is related with the error on the flow consumption prediction $d(k)$, and thus can be measured at current time instant. That is, a vector $\hat{w}(k) \in \mathcal{W}$ such that $E_w(w(k) - \hat{w}(k)) = 0$ can be computed at k.

Assumption 1(b) reflects that active actuators can always be turned off if necessary. Assumption 1(c) is introduced to exploit the temporal redundancy existing in the demands of flow-based networks because of the existence of daily/weekly periodic behaviors associated with human habits [23]. Besides, possible uncertainties in the periodic forecast can be embedded into the uncertain variable $w(k)$. On the other hand, Assumption 1(d) imposes that the current perturbation affecting the static nodes is known at current k, but unknown at future samples. Note that this assumption does not involve the disturbances affecting the system dynamics, where process disturbances cannot be known at current time instant.

2.1. Actuator configurations

Let $C_0 = \{a_1, a_2, \ldots, a_{n_u}\}$ denote the full set of actuators (nominal plus back-up actuators) of the system, where a_i stands for the ith actuator. Furthermore, let $C_i \subseteq C_0$, $i \in \{0, 1, \ldots, 2^{n_u} - 1\}$, denote the possible actuator configurations that arise from the selection of a subset of the n_u actuators. In the sequel, the cardinality of configuration C_i, i.e. the number of elements that it contains, is denoted as $|C_i|$. In addition, let 2^{C_0} term the power set $2^{C_0} = \{C_i : C_i \subseteq C_0\}$ of all system configurations.

The configuration selection matrix Σ in (1) is such that $\Sigma = diag(\delta)$, where $\delta \in \{0, 1\}^{n_u}$ is a binary vector that rules the activation of a specific configuration. Hereafter, $\Sigma^{[i]}$ is used to denote that the system is operating in configuration C_i. Note that the superscript $^{[i]}$ is in brackets to avoid confusion with powers. As an example, consider a system with $n_u = 3$ actuators $C_0 = \{a_1, a_2, a_3\}$, as well as the configuration $C_1 = \{a_1, a_2\}$, then $\Sigma^{[1]} = diag([1; 1; 0])$.

In the sequel, the set of available actuators is considered to be split into $C_0 = C_n \cup C_a$, such that C_n denotes the nominal configuration and C_a encompasses the remaining actuators considered as back-up elements. Consistently, in nominal operation the system operates with $\Sigma^{[n]}$.

Definition 1 (*Predecessors/Successors [13]*)**.** Given the set of actuators C_0 and a configuration $C_i \subseteq C_0$, the predecessors $\mathbb{P}(C_i)$ and successors $\mathbb{S}(C_i)$ of C_i are defined as

$$\mathbb{P}(C_i) = \{C_j : C_i \subseteq C_j \subseteq C_0\},$$
$$\mathbb{S}(C_i) = \{C_j : C_j \subseteq C_i\}.$$

Definition 2 (*Predecessors After Fault*)**.** Given the set of actuators C_0 and a configuration $C_i \subset C_0$, the set of remaining predecessors of configuration C_i after an outage of the actuators in C_{out} is defined as

$$\mathbb{P}(C_i|C_{out}) = \{C_j : C_i \subseteq C_j \wedge C_j \subseteq (C_0 \setminus C_{out})\}.$$

Definition 3 (*Span*)**.** The span of property $\mathcal{P}$ is the set $S_p(\mathcal{P})$ of all configurations in 2^{C_0} that satisfy $\mathcal{P}$, that is

$$S_p(\mathcal{P}) = \{C_i \in 2^{C_0} : \mathcal{P}(C_i)\}.$$

2.2. Fault detection and isolation

The following standard assumption in the FTC literature (see as e.g. [24,25]) is made about the existence of a fault detection and isolation (FDI) block that monitors the system operation. This allows studying separately the FDI and FTC problems.

Assumption 2. An FDI block is available and able to detect and isolate the presence of actuator faults. The faulty components are immediately shut down.

The design of efficient FDI blocks for FNs has been thoroughly investigated using several approaches [26–28]. Accordingly, the following time sequence is established:

- **Fault time** (k_f): An actuator fault occurs in the subset of actuators $C_{out} \subseteq C_n$, such that the system operates in the faulty configuration $C_f = C_n \setminus C_{out}$.
- **Detection time** (k_d): At $k_d \geq k_f$ an FDI block detects and isolates the faulty configuration C_f, modifying the selection matrix $\Sigma^{[n]} \to \Sigma^{[f]}$.

2.3. Performance assessment

Hereafter, it is considered that the closed-loop performance of the system operating in the different configurations is assessed by means of an economic time varying stage cost function $l : 2^{C_0} \times \mathbb{N} \times \mathcal{X} \times \mathcal{U} \to \mathbb{R}$. As regards the cost function, the following assumption is introduced.

Assumption 3. For a given configuration C_i, the cost function $l(\cdot)$ is assumed to be positive; convex in (x, u) for all k; and periodic, i.e. $l(C_i, k, x, u) = l(C_i, k+T, x, u)$.

Note that the first two conditions on Assumption 3 are introduced for convergence issues, whereas the periodicity constraint in the stage function typically follows from the periodic patterns in electricity pricing.

Consistently, the system performance is assessed as the average of the stage cost function obtained by the closed-loop trajectories. This can be written as

$$\mathcal{L}_{\infty}^{[i]}(x_0, \boldsymbol{u}_{\infty}) = \lim_{m \to \infty} \frac{1}{m} \sum_{k=0}^{m-1} l(C_i, k, x(k), u(k)), \tag{4}$$

where x_0 is the initial state and $\boldsymbol{u}_{\infty}$ is the set of corresponding closed-loop input trajectories. In this regard, due to periodic nature of the systems under study, it is well-known that the optimal trajectories of the system without unknown disturbances can be obtained by solving a finite horizon open-loop problem that optimizes the average cost over a single period T [22, Theorem 1]. For the sake of simplified notation, the optimal T-period average stage cost value that can be achieved by the system operating on configuration C_i is termed as $\mathcal{L}_T^{[i]}$.

Remark 2. Note that the computation of the optimal trajectory should be performed by taking into consideration the effect of the uncertainty sources in the cost function, while guaranteeing a robust constraint satisfaction. However, due to the computational complexity of the methods that consider the uncertainty in the stage cost predictions [29,30], in the remainder of the article the approach considered in [5,31] will be followed, and only the stage cost of the nominal (non-disturbed) system is minimized.

2.4. Admissibility

Given a generic property $\mathcal{P}$, its assessment on the configuration C_i is formulated as

$$\begin{cases} \mathcal{P}(C_i) & \implies \quad \mathcal{P} \text{ is satisfied on } C_i, \\ \neg\mathcal{P}(C_i) & \implies \quad \mathcal{P} \text{ is not satisfied on } C_i. \end{cases}$$

Accordingly, the following definition is introduced concerning the admissibility of an arbitrary configuration C_i.

Definition 4 (*Admissibility*)**.** Given system (1) with configuration C_i imposed through matrix $\Sigma^{[i]}$. Then, the admissibility of C_i at time k^* is expressed as

$$\mathcal{A}(C_i|k^*) = \mathcal{A}^{st}(C_i|k^*) \wedge \mathcal{A}^{pf}(C_i), \tag{5}$$

where

1. **Stability admissibility:** $\mathcal{A}^{st}(C_i|k^*)$ if starting at $x(k^*)$, the closed-loop system with C_i converges asymptotically towards a neighborhood of the steady-state trajectory with cost $\mathcal{L}_T^{[i]}$, while guaranteeing the robust constraint satisfaction for all $k \geq k^*$.
2. **Performance admissibility:** $\mathcal{A}^{pf}(C_i)$ if the optimal average cost of C_i is lower than β times the cost obtained for the nominal configuration C_n, i.e. if $\mathcal{L}_T^{[i]} \leq \beta\mathcal{L}_T^{[n]}$ for a given $\beta \geq 1$ defined by the user.

Remark 3. Under an MPC control policy, $\mathcal{A}^{st}(C_i|k)$ is equivalent to verify $x(k) \in \mathbb{X}^{[i]}$, where $\mathbb{X}^{[i]}$ is the region of attraction of a robust MPC controller that uses configuration C_i. On this subject, the consideration of large-scale systems usually precludes the explicit computation of $\mathbb{X}^{[i]}$.

3. Problem statement

Consider system (1) running in nominal configuration C_n. If an actuator fault causes that the resulting faulty configuration is not admissible at fault detection time (i.e. $\neg\mathcal{A}(C_f|k_d)$), then the available back-up components can be brought into play seeking for a new admissible configuration denoted as C_{new}. The search for a new configuration is formulated as solving the following problem [16]

$$\begin{aligned} C_{new} = \quad & \arg\min_{C_i} J(C_i), \\ \text{s.t.} \quad & C_i \in \mathbb{P}(C_f|C_{out}), \\ & C_i \in S_p(\mathcal{A}(\cdot|k_d)), \end{aligned} \tag{6}$$

where $J(C_i)$ stands for the overall cost of configuration C_i according to some pre-established criteria that rule the configuration selection. The admissibility conditions that must satisfy the new configuration are addressed in Sections 4 and 5, whereas the online solution of (6) is studied in Section 6.

4. Stability admissibility

Motivated by the discussion presented in Section 1, the robust control of a FN with configuration C_i at time k (i.e. $\mathcal{A}^{st}(C_i|k)$), is addressed using a single-layer economic MPC control scheme [22,32].

4.1. Robust planner

Firstly, the optimal T-periodic average stage cost for an N-horizon MPC controller is computed. In this regard, the so-called *robust planner* optimization problem for the system with configuration C_i is posed as

$$\begin{aligned} &\min_{x_0, \boldsymbol{u}_T} \ \mathcal{L}_T^{[i]} = \frac{1}{T}\sum_{j=0}^{T-1} l(C_i, j, x(j), u(j)), \\ &\text{s.t. } x(j+1) = Ax(j) + B\Sigma^{[i]}u(j) + B_d d(j), && \forall j \in \mathbb{I}_{T-1}, \\ &0 = E\Sigma^{[i]}u(j) + E_d d(j), && \forall j \in \mathbb{I}_{T-1}, \\ &x(T) = x(0) = x_0, \\ &x(j) \in \mathcal{X}^{[i]}(N), && \forall j \in \mathbb{I}_{T}, \\ &u(j) \in \mathcal{U}^{[i]}(N), && \forall j \in \mathbb{I}_{T-1}, \end{aligned} \tag{7}$$

where the optimal points $\boldsymbol{x}_o^{[i]}, \boldsymbol{u}_o^{[i]}$ of (7) are denoted as *robust planner trajectories*. Besides, $\mathcal{X}^{[i]}(N)$ and $\mathcal{U}^{[i]}(N)$ represent a tightened set of state and input constraints for configuration C_i whose computation will be detailed later.

4.2. Uncertainty attenuation in the predictions

Below, the linearity of the system is exploited to separate the effect of the uncertainty in the predictions. To that end, let $\tilde{x}(k) \in \mathbb{R}^{n_x}$ term the nominal predictions (i.e. without considering the disturbances), then the error $e(k) = x(k) - \tilde{x}(k) \in \mathbb{R}^{n_x}$ evolves according to

$$e(k+1) = Ae(k) + B\Sigma^{[i]}e_u(k) + B_w w(k), \tag{8a}$$

$$0 = E\Sigma^{[i]}e_u(k) + E_w w(k), \tag{8b}$$

where $e_u(k) = u(k) - \tilde{u}(k) \in \mathbb{R}^{n_u}$ and $\tilde{u}(k)$ is the nominal input that yields $\tilde{x}(k)$.

Following the proposal of [5], an auxiliary input $h(k) \in \mathbb{R}^{n_u - n_e}$ is introduced in order to impose the satisfaction of (1b) for any possible $w(k) \in \mathcal{W}$. This auxiliary input $h(k)$ is derived from the explicit solution of (8b), in such a way that $e_u(k)$ can rewritten as

$$e_u(k) = M_w^{[i]}w(k) + M_v^{[i]}h(k), \tag{9}$$

for some matrices $M_v^{[i]}$ and $M_w^{[i]}$. Hence, the robust satisfaction of (8b) can be guaranteed by designing some matrices $M_v^{[i]}$ and $M_w^{[i]}$ such that satisfy (the computation of $M_v^{[i]}$ and $M_w^{[i]}$ is addressed in Section 4.4)

$$E\Sigma^{[i]}M_v^{[i]} = 0, \tag{10a}$$

$$E_w + E\Sigma^{[i]}M_w^{[i]} = 0. \tag{10b}$$

From (9) and (10), system (8) can rewritten as

$$e(k+1) = Ae(k) + \hat{B}^{[i]}h(k) + \hat{B}_w^{[i]}w(k), \tag{11}$$

where $\hat{B}^{[i]} = B\Sigma^{[i]}M_v^{[i]}$ and $\hat{B}_w^{[i]} = B_w + B\Sigma^{[i]}M_w^{[i]}$.

Therefore, in order to attenuate the uncertainty propagation characterized by (11), a suboptimal solution to the control problem is typically obtained through the *a priori* design of a linear control law of the form $h(k) = K^{[i]}e(k)$, with matrix $K^{[i]}$ designed such that $A + \hat{B}^{[i]}K^{[i]}$ is an asymptotically stable matrix [33,34].

4.3. Single-layer robust MPC

Here, the single-layer robust MPC policy is presented. To that end, the following cost function is introduced [22]

$$\begin{aligned} &V(x, C_i, \boldsymbol{d}_{k:k+N}; \boldsymbol{u}_N, x_0^v, \boldsymbol{u}_T^v) \\ &= V_t(x, C_i, \boldsymbol{d}_{k:k+N}; \boldsymbol{u}_N, x_0^v, \boldsymbol{u}_T^v) + V_p(C_i, \boldsymbol{d}_{k:k+N}; x_0^v, \boldsymbol{u}_T^v), \end{aligned} \tag{12}$$

where the parameters that define the optimization at time k are: the system state x, the current configuration C_i and the sequence of flow demand predictions $\boldsymbol{d}_{k:k+N}$. Besides, the optimization variables are: the N-sequence of control input variables $\boldsymbol{u}_N$, as well as x_0^v and $\boldsymbol{u}_T^v$ which are decision variables associated with the introduction of an artificial reference denoted by means of the superscript ^{v}. The term $V_t(\cdot)$ is introduced to penalize the error between the open-loop trajectories and the artificial reference, whereas $V_p(\cdot)$ is used to penalize the average stage cost of the artificial reference. These objectives are formulated as

$$V_t(\cdot) = \sum_{j=0}^{N-1} \|x(j) - x^v(j)\|_Q^2 + \|u(j) - u^v(j)\|_R^2, \tag{13a}$$

$$V_p(\cdot) = \frac{1}{T}\sum_{j=0}^{T-1} l(C_i, k+j, x^v(j), u^v(j)), \tag{13b}$$

with $Q = Q^T \succ 0$, $R = R^T \succ 0$ and $N \leq T$.

Accordingly, the optimal trajectories of the robust economic MPC are obtained from the solution of the following finite horizon control problem

$$\min_{\boldsymbol{u}_N, x_0^v, \boldsymbol{u}_T^v} V(x, C_i, \boldsymbol{d}_{k:k+N}; \boldsymbol{u}_N, x_0^v, \boldsymbol{u}_T^v),$$

$$\text{s.t.} \quad x(0) = x, \tag{14a}$$

$$x(j+1) = Ax(j) + B\Sigma^{[i]}u(j) + B_d d(k+j), \tag{14b}$$

$$0 = E\Sigma^{[i]}u(j) + E_d d(k+j), \qquad \forall j \in \mathbb{I}_{N-1}, \tag{14c}$$

$$x(N) = x^v(N), \tag{14d}$$

$$x^v(j+1) = Ax^v(j) + B\Sigma^{[i]}u^v(j) + B_d d(k+j), \tag{14e}$$

$$0 = E\Sigma^{[i]}u^v(j) + E_d d(k+j), \qquad \forall j \in \mathbb{I}_{N-1}, \tag{14f}$$

$$x(j) \in \mathcal{X}^{[i]}(j), \quad x^v(j) \in \mathcal{X}^{[i]}(N), \qquad \forall j \in \mathbb{I}_N, \tag{14g}$$

$$u(j) \in \mathcal{U}^{[i]}(j), \quad u^v(j) \in \mathcal{U}^{[i]}(N), \qquad \forall j \in \mathbb{I}_{N-1}, \tag{14h}$$

$$x^v(T) = x^v(0) = x_0^v, \tag{14i}$$

with the tightened set of constraints

$$\begin{aligned}
&\mathcal{X}^{[i]}(0) = \mathcal{X},\\
&\mathcal{U}^{[i]}(0) = \mathcal{U} \ominus M_w^{[i]}\mathcal{W},\\
&\mathcal{X}^{[i]}(j) = \mathcal{X} \ominus \mathcal{R}^{[i]}(j),\\
&\mathcal{U}^{[i]}(j) = \mathcal{U} \ominus M_w^{[i]}\mathcal{W} \ominus M_v^{[i]}K^{[i]}\mathcal{R}^{[i]}(j),\\
&\mathcal{R}^{[i]}(j) = \bigoplus_{0}^{j-1} \mathcal{Q}^{[i]}(l),\\
&\mathcal{Q}^{[i]}(l) = (A + \hat{B}^{[i]}K^{[i]})^l \hat{B}_w^{[i]}\mathcal{W}.
\end{aligned} \tag{15}$$

Assumption 4. The closed-loop matrix $A + \hat{B}^{[i]}K^{[i]}$ satisfies $(A + \hat{B}^{[i]}K^{[i]})^{N-1}\hat{B}_w^{[i]}w = 0$, $\forall w \in \mathcal{W}$; the sets $\mathcal{X}^{[i]}(j)$ and $\mathcal{U}^{[i]}(j)$ are non-empty for $j \in \mathbb{I}_N$.

Finally, the following control law is introduced

$$u(k) = u^*(0|k) + M_w^{[i]}\hat{w}(k), \tag{16}$$

where $u^*(0|k)$ denotes the first optimum value of (14) computed at k.

Theorem 1 (*[5, Theorem 1]*)**.** *If the conditions given in Assumption 4 hold, then the system (1) in configuration C_i controlled by the control law (16) is recursively feasible and the robust planner trajectory $\boldsymbol{x}_o^{[i]}$ that yields the average state cost $\mathcal{L}_T^{[i]}$ is input-to-state stable.*

Remark 4. Through the dead-beat controller introduced in Assumption 4, Eq. (14d) becomes a terminal ingredient for ensuring the closed-loop stability without the need of computing a robust terminal invariant set [32]. Note that this set computation may be intractable for large-scale systems. Assumption 4 can be relaxed by imposing that $\max_{w\in\mathcal{W}} \|(A + \hat{B}^{[i]}K^{[i]})^{N-1}\hat{B}_w^{[i]}w\|_\infty = \|(A + \hat{B}^{[i]}K^{[i]})^{N-1}\hat{B}_w^{[i]}\|_\infty$ is below a pre-specified threshold that relates with the numerical precision of the computer ((3) has been used in the derivation of previous equality).

4.4. Parametrized solution of the robust planner

Regarding the constraints introduced in (10), matrix $M_v^{[i]}$ can be designed as an orthonormal basis to the null space of the $n_e \times n_u$ matrix $E\Sigma^{[i]}$ (with $n_e < n_u$). However, on the other hand, an inappropriate selection of matrix $M_w^{[i]}$, which is in charge of distributing among the different actuators the compensations of the uncertainty in the static nodes (cf. (15)), may have a harmful effect on the optimal operation of the system or even generate an infeasible solution region. In this regard, in [5,35] the Moore–Penrose pseudo-inverse is used in order to compute matrix $M_w^{[i]}$, whereas in [36,37] the matrix $M_w^{[i]}$ results from an actuators permutation selected by the user. Nevertheless, none of the above designs takes into account the state and input constraints. In order to address this problem, the fact $M_w^{[i]}$ is affine to the tube of uncertain trajectories expressed by (11) is exploited by including it as an optimization variable in the robust planner computation.

Corollary 1. *Under Assumption 4, the convex optimization problem (17) is a robust planner for configuration C_i.*

$$\min_{M_w^{[i]}, x_0, \boldsymbol{u}_T, \Omega_x, \Omega_u} \mathcal{L}_T^{[i]} = \frac{1}{T}\sum_{j=0}^{T-1} l(C_i, j, x(j), u(j)),$$

$$\text{s.t.} \quad x(j+1) = Ax(j) + B\Sigma^{[i]}u(j) + B_d d(j), \tag{17a}$$

$$0 = E\Sigma^{[i]}u(j) + E_d d(j), \qquad \forall j \in \mathbb{I}_{T-1}, \tag{17b}$$

$$x(T) = x(0) = x_0, \tag{17c}$$

$$E\Sigma^{[i]}M_w^{[i]} = -E_w, \tag{17d}$$

$$H_x x(j) \leq h_x - \Gamma - \Omega_x \mathbf{1}_{Nn_w}, \qquad \forall j \in \mathbb{I}_T, \tag{17e}$$

$$H_u u(j) \leq h_u - \Delta - \Omega_u \mathbf{1}_{Nn_w}, \qquad \forall j \in \mathbb{I}_{T-1}, \tag{17f}$$

$$\Lambda(I_N \otimes M_w^{[i]}) \leq \Omega_x, \; -\Lambda(I_N \otimes M_w^{[i]}) \leq \Omega_x, \tag{17g}$$

$$\Xi(I_N \otimes M_w^{[i]}) \leq \Omega_u, \; -\Xi(I_N \otimes M_w^{[i]}) \leq \Omega_u, \tag{17h}$$

$$\begin{aligned}
&\Gamma = |H_x G_a(N)|\mathbf{1}_{Nn_w}, \qquad \Delta = |H_u M_v^{[i]}K^{[i]}G_a(N)|\mathbf{1}_{Nn_w},\\
&\Lambda = H_x G_b(N), \qquad \Xi = H_u(\tilde{I} + M_v^{[i]}K^{[i]}G_b(N)),\\
&G_a(N) = \left[(A + \hat{B}^{[i]}K^{[i]})^{N-1}B_w \ \dots \ B_w\right],\\
&G_b(N) = \left[(A + \hat{B}^{[i]}K^{[i]})^{N-1}B \ \dots \ B\right],\\
&\tilde{I} = [I_{n_u} \ 0_{n_u \times n_u(N-1)}].
\end{aligned}$$

Proof. The proof is presented in Appendix. □

It must be pointed out that in single-layer economic MPC controllers [9,22], the robust planner is only introduced in order to proof stability properties, and thus not required to be solved. Conversely, here, the optimization (17) is solved with a double purpose: (I) computing $M_w^{[i]}$; (II) computing the optimal average stage cost $\mathcal{L}_T^{[i]}$ that can be attained by configuration C_i. Observe that (17) is solved offline, and therefore the computational complexity added with the parametrization of $M_w^{[i]}$ is not a problem.

5. Performance admissibility

The best attainable average steady-state cost $\mathcal{L}_T^{[i]}$ is used to assess the performance admissibility of configuration C_i. To that end, the time independence of the robust planner trajectories

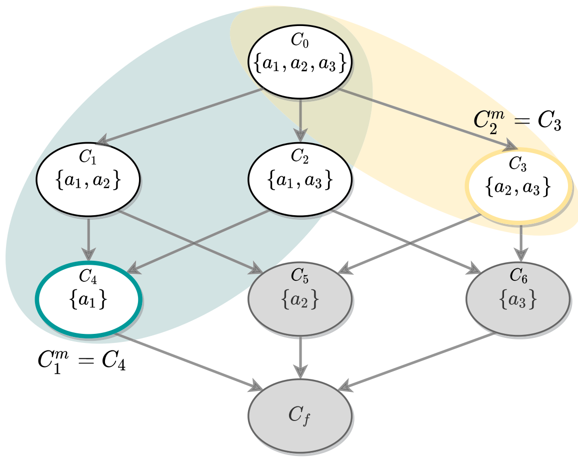

Fig. 1. Lattice of $\mathbb{P}(C_f|C_{out})$: $S_p(\mathcal{A}^{pf})$ white nodes; $\mathbb{P}(C_1^m)$ blue ellipse; $\mathbb{P}(C_2^m)$ yellow ellipse.

is exploited to filter out offline those configurations that yield a non-admissible steady-state behavior. However, note that the robust planner depends on the local controller designed to attenuate the uncertainty $K^{[j]}$ (see Section 4.2), which may be designed using a different configuration $C_j \neq C_i$, as long as $C_j \subseteq C_i$, i.e. if C_j is used by the local controller this configuration must be necessarily activated. In order to emphasize this difference, hereafter we denote as $\mathcal{L}_T^{[i|j]}$ the average stage cost obtained by solving the robust planner optimization for C_i for a local controller $K^{[j]}$ computed using C_j (with $C_j \subseteq C_i$).

Lemma 1. *Under Assumption 1(b), the performance admissibility $\mathcal{A}^{pf}(\cdot)$ constitutes a bottom-up monotonous (BUM) property, that is,*

$$\mathcal{A}^{pf}(C_i) \implies \mathcal{A}^{pf}(C_l), \ \forall C_l \in \mathbb{P}(C_i).$$

Proof. For any C_l, with $C_l \supset C_i$, the robust planner optimization can be formulated in such a way that yields the cost $\mathcal{L}_T^{[l|i]}$. On the other hand, the robust planner optimization for C_i can be posed similarly to the one that yields $\mathcal{L}_T^{[l|i]}$ plus an additional constraint that sets to zero the elements in $C_l \setminus C_i$, which is possible from Assumption 1(b). Therefore, from optimality it follows that $\mathcal{L}_T^{[i|i]} \geq \mathcal{L}_T^{[l|i]}$, and thus $\mathcal{A}^{pf}(C_i) \implies \mathcal{A}^{pf}(C_l)$. $\square$

5.1. Search for minimal configurations

Definition 5 (*Set of Minimal Configurations*)**.** The set of minimal elements of a subset of configurations $Q \subseteq 2^{C_0}$ is defined by

$$m(Q) = \{C_i \in Q : C_j \subset C_i \implies C_j \notin Q\}.$$

Here, we focus on searching for the set of minimal configurations $\mathcal{M}^{[f]}$ that are performance admissible for the faulty configuration C_f, that is,

$$\mathcal{M}^{[f]} = m(S_p(\mathcal{A}^{pf}) \cap \mathbb{P}(C_f|C_{out})).$$

For the sake of simplified notation, we consider that $\mathcal{M}^{[f]}$ is conformed by $n_l = |\mathcal{M}^{[f]}|$ minimal configurations denoted by $C_l^m, \ \forall l \in \{1, \ldots, n_l\}$. That is, we introduce the bijective mapping $\phi : \mathbb{N} \to \mathbb{N}$ to link the indexes of C_i and C_l^m, such that $l = \phi(i), \forall i : C_i \in \mathcal{M}^{[f]}$. As an example, consider the lattice of configurations in Fig. 1, then $\mathcal{M}^{[f]} = \{C_1^m, \ C_2^m\}$ with $C_1^m = C_4$ and $C_2^m = C_3$.

Note that a new admissible configuration must be such that $C_{new} \in S_p(\mathcal{A}^{pf}) \cap \mathbb{P}(C_f|C_{out})$, and thus from Definition 5 it follows that $C_{new} \in \mathbb{P}(C_1^m) \cup \cdots \cup \mathbb{P}(C_{n_l}^m)$. In other words, we only need to conduct the online search on the predecessors of the minimal configurations in $\mathcal{M}^{[f]}$. Nonetheless, although intended to be run offline, the search for minimal configurations can become intractable in the case that a large number of configurations must be explored. In this regard, Algorithm 1 exploits the bottom-up monotonicity of $\mathcal{A}^{pf}(\cdot)$ in Lemma 1 to conduct an efficient search for minimal configurations.

Algorithm 1 takes as inputs the set $\mathcal{I} = \{C_1, \ldots, C_{n_a}\}$ of n_a different candidate configurations that have been sorted following a cardinality ordering (i.e., $|C_i| \geq |C_{i+1}|$) and the BUM property $\mathcal{P}$ that is desired to be evaluated; and returns the minimal set of configurations $\mathcal{M}$. Given a configuration that satisfies $\mathcal{P}$, the algorithm favors to continue the search in its strict successors, whereas if $\mathcal{P}$ is not satisfied, it returns to the previous admissible configuration with lower cardinality. On the other hand, whenever a minimal configuration is found, the search is restarted from the top of the remaining lattice.

Notably, the exploration of the next strict successor of a configuration C_x in the set of remaining candidate configurations $\mathcal{I}$ is performed by function `next_strict_successor` that is called in line 7 of Algorithm 1. This function, first searches for configuration C_x in $\mathcal{I}$, and then it profits from the cardinality ordering of the configurations in $\mathcal{I}$ to explore the highest cardinality strict successor of C_x.

In the case that the strict successor C_y satisfies $\mathcal{P}$ (lines 13–15), C_y becomes the new C_x and all its predecessors can be eliminated from the search due to $\mathcal{P}$ being BUM. In the case that C_y does not satisfy $\mathcal{P}$ (lines 16–17), all its successors can be removed from the search due to $\mathcal{P}$ being BUM, and the search continues in the next strict successor of C_x. The above elimination of successor configurations guarantees that if no strict successor exists (lines 8–11), then the explored configuration is minimal and all its predecessors can be eliminated. At that point, the search is restarted from the highest cardinality remaining configuration (line 11). The performance of Algorithm 1 is illustrated in the case study presented in Section 7.

6. Solution of the reconfiguration problem

This section aims at solving the reconfiguration problem (6) subject to the admissibility criterion presented in Definition 4. In this regard, the stability of the FN in a specific configuration is addressed using the single-layer robust MPC scheme in Section 4. Nevertheless, as discussed in Section 5, robust control schemes are designed as a two-step procedure where: (I) a state feedback control law is designed to compensate the uncertainty (see Section 4.2) using configuration C_j; (II) the evolution of the nominal (no uncertainty) trajectory for configuration $C_i \supseteq C_j$ is optimized for a tightened set of constrains computed for the *a priori* selected configuration C_j (see Section 4.3).

Hence, the *a priori* selection of the actuators used by the local controller may have a high impact on the optimality of the solution of the configuration selection problem (6). Ideally, since it offers more degrees of freedom for the design, the best approach is to use the same configuration for both: the local controller and the nominal predictions ($C_j = C_i$), instead of a subset of the actuators ($C_j \subset C_i$). However, this would imply to assess the stability admissibility after a fault in each candidate configuration independently. On this subject, note that for large-scale systems with multiple back-up components, and thus with a huge number of possible new configurations, this candidate by candidate online assessment becomes intractable.

Therefore, the configuration selection (6) is approximated basing the uncertainty compensation on the minimal configurations C_l^m that yield an admissible performance (see Section 5.1). By these means, two different phases are considered:

Algorithm 1 Search of minimal configurations over a BUM property.

```
Input: 𝓘, 𝓟
Output: 𝓜
𝓜 ← ∅                                          ▷ Initialize the empty set
if ¬𝓟(𝓘(1)) then
    Stop                                       ▷ No solution
end if
C_x ← 𝓘(1)
while 𝓘 ≠ ∅ do
    C_y ← next_strict_successor(𝓘, C_x)
    if C_y = ∅ then
        𝓜 ← {𝓜, C_x}                           ▷ C_x is minimal
        Remove ℙ(C_x) from 𝓘
        C_x ← 𝓘(1)
    else
        if 𝓟(C_y) then
            C_x ← C_y
            Remove ℙ(C_y) from 𝓘
        else
            Remove 𝕊(C_y) from 𝓘
        end if
    end if
end while

function next_strict_successor(𝓘, C_x)
    i ← 1
    while all(is_member(𝓘(i), C_x)) = False do
        i ← i + 1
        if i > |𝓘| then
            error('No C_x in 𝓘')
        end if
    end while
    if i = |𝓘| then
        C_y ← ∅                                ▷ C_x is the last conf.
    else
        j ← i + 1
        while all(is_member(𝓘(j), 𝓘(i))) = False do
            j ← j + 1
            if j > |𝓘| then
                C_y ← ∅
                Break all                      ▷ Breaks all the loops
            end if
        end while
        C_y ← 𝓘(j)
    end if
    return C_y
end function
```

- *Offline*: the set of minimal configurations that yield an admissible steady-state cost (i.e. such that $\mathcal{A}^{pf}(C_l^m)$), are identified offline by means of Algorithm 1. A local controller $K^{[l]}$ is computed for $C_l^m \in \mathcal{M}^{[f]}$.
- *Online*: given a fault in C_{out}, the search for the optimal admissible configuration (6) that allows to handle the transient induced by the fault is conducted online in the remaining predecessors of each minimal configuration identified offline, that is, in $\mathbb{P}(C_l^m|C_{out})$.

6.1. Online MIP optimization

Given a fault in C_{out}, the online configuration search in the remaining predecessors of a minimal configuration can be posed as an MIP that selects the optimal configuration that satisfies the constraints of a single-layer robust MPC (14). To that end, let $\mathcal{X}_m^{[l]}(j), \mathcal{U}_m^{[l]}(j)$ represent the tightened set of state and input constraints for the minimal configuration C_l^m computed according to (15). Moreover, let $\delta_m^{[l]}$ denote a binary vector with: the elements in C_l^m activated (and thus not considered as variables in the optimization problem), and the elements in $(C_0 \setminus C_{out}) \setminus C_l^m$ as binary optimization variables. Consistently, the selection of a new configuration $C_{new}^{(l)}$ in the set the set $\mathbb{P}(C_l^m|C_{out})$ can be posed as the following MIP

$$C_{new}^{(l)} = \underset{\delta_m^{[i]}, \mathbf{u}_N, x_0^v, \mathbf{u}_N^v}{\arg\min}\ J(C_i),$$

$$\text{s. t.}\quad x(0) = x, \tag{18a}$$

$$x(j+1) = Ax(j) + Bz(j) + B_d d(k+j), \tag{18b}$$

$$0 = Ez(j) + E_d d(k+j), \qquad j \in \mathbb{I}_{N-1}, \tag{18c}$$

$$x(N) = x^v(N), \tag{18d}$$

$$x^v(j+1) = Ax^v(j) + Bz^v(j) + B_d d(k+j), \tag{18e}$$

$$0 = Ez^v(j) + E_d d(k+j), \qquad j \in \mathbb{I}_{N-1}, \tag{18f}$$

$$x^v(T) = x^v(0) = x_0^v, \tag{18g}$$

$$x(j) \in \mathcal{X}_m^{[l]}(j), \qquad x^v(j) \in \mathcal{X}_m^{[l]}(N), \qquad j \in \mathbb{I}_N, \tag{18h}$$

$$u(j) \in \mathcal{U}_m^{[l]}(j), \qquad u^v(i) \in \mathcal{U}_m^{[l]}(N), \qquad j \in \mathbb{I}_{N-1}, \tag{18i}$$

$$z(j) = diag(\delta_m^{[l]})u(j), \tag{18j}$$

$$z^v(j) = diag(\delta_m^{[l]})u^v(j), \qquad j \in \mathbb{I}_{N-1}, \tag{18k}$$

where the product of continuous and logic variables in (18j)–(18k) can be transformed into equivalent linear integer inequalities [38].

Note that, by means of the single-layer MPC scheme, the integer program (18) updates coherently the model used for control and the model used for the virtual planner. This ensures the generation of a reachable trajectory for the new configuration (retrieved from the binary vector $\delta_m^{[l]}$), since the terminal ingredient used for stability (18d) is also modified coherently with the virtual planner. Consequently, the system in configuration $C_{new}^{(l)}$ will satisfy the stability admissibility $\mathcal{A}^{st}(C_{new}^{(l)}|k_d)$, whereas, on the other hand, since $C_{new}^{(l)} \in \mathbb{P}(C_l^m|C_{out})$, from Lemma 1 it follows that $\mathcal{A}^{pf}(C_{new}^{(l)})$, and thus $C_{new}^{(l)}$ is admissible.

Remark 5. The satisfaction of $\mathcal{A}^{pf}(C_i)$ for a given C_i, requires the tightened sets of constraints $\mathcal{X}^{[i]}(N)$ and $\mathcal{U}^{[i]}(N)$ to be non empty. In the case that this is not satisfied, then the offline search for minimal configurations that satisfy $\mathcal{A}^{pf}(\cdot)$ will filter out C_i as non admissible. Besides, from (15), it follows that $\mathcal{X}^{[i]}(N) \subseteq \mathcal{X}^{[i]}(j),\ \forall j \in \mathbb{I}_{N-1}$ (similarly for $\mathcal{U}^{[i]}(j)$), and thus, if $\mathcal{A}^{pf(\cdot)}$ is satisfied, then the tightened set of constraints are guaranteed to be non empty for all $j \in \mathbb{I}_N$.

Remark 6. The proposed methodology implicitly assumes that the whole system reconfiguration takes less than one sampling time, and thus delays are negligible. If that is not the case, the online optimization (18) should be adapted to deal with the (previously modeled) input-delays caused by the reconfiguration as proposed in [39–41].

6.2. Decision between sets

Section 6.1 formulates an MIP for obtaining (if it exists) a new configuration $C_{new}^{(l)}$ for each one of the sets $C_l^m \in \mathcal{M}^{[f]}$. Below, a

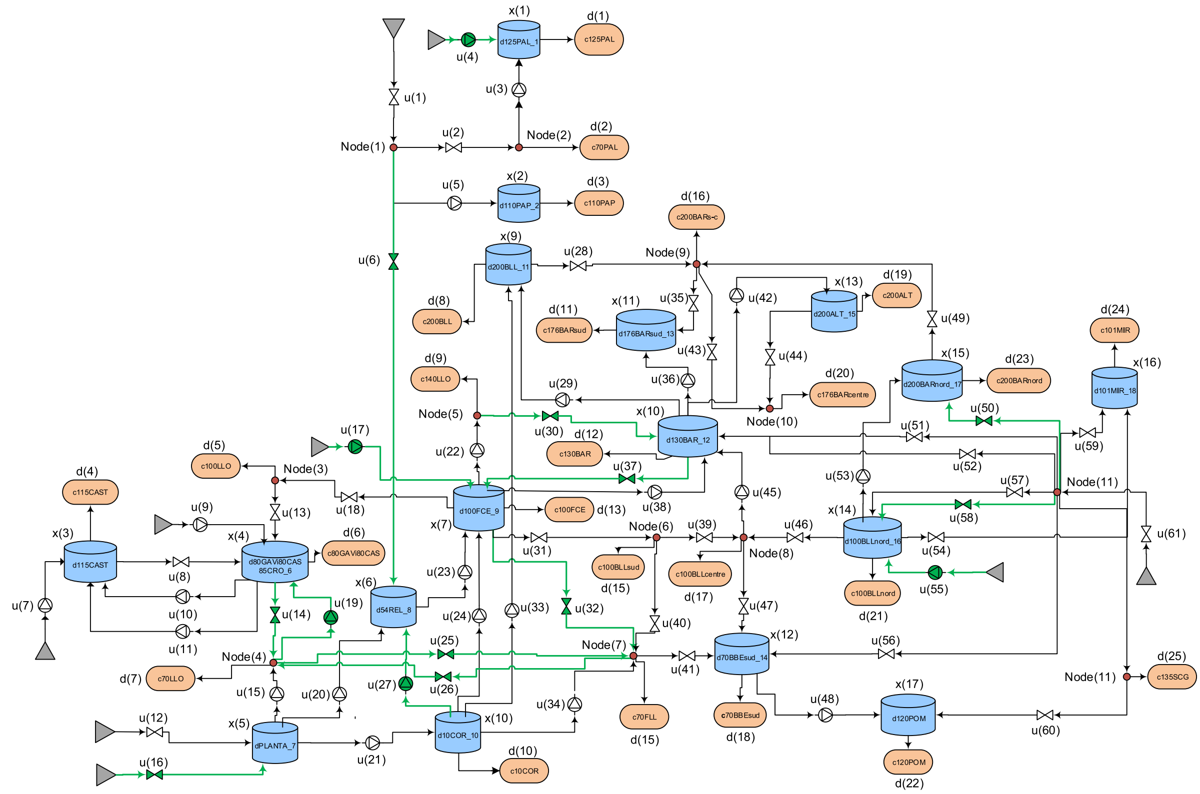

Fig. 2. DWTN description. Nominal actuators (black); back-up actuators (green).

Algorithm 2 Sequential configuration search

Input: $P_l,\ \ l \in \{1, \ldots, n_l\}$
Output: C_{new}

```
J* ← 10^10                                   ▷ Initialize large value
C_new ← ∅                                     ▷ Initialize empty configuration
for l = 1 to n_l do
    P̃_l ← min{J(ψ_l) : ψ_l ∈ Ψ_l ∧ J(ψ_l) ≤ J*}
    if is_feasible(P̃_l) then
        J* ← solution(P̃_l)
        C_new ← argmin(P̃_l)
    end if
end for
if is_empty(C_new) then
    No solution
end if
```

sequential method is followed in order to decide how to conduct the search among the different sets and select a final C_{new}. In this sequential approach, the information retrieved from solving the configuration selection problem in one set is used for limiting the search space in the remaining sets.

For simplicity, let us characterize each of the n_l optimizations (18) as the optimization problem

$$(\mathrm{P}_l) \quad \min\{J(\psi_l) \ :\ \psi_l \in \Psi_l\}, \tag{19}$$

where the vector ψ_l encompasses the different decision variables and Ψ_l is the feasibility set obtained for C_l^m. Hence, using the notation in (19), Algorithm 2 reflects the steps followed to compute the final configuration C_{new}. Note that, any intermediate solution C_{new} found by Algorithm 2 is an admissible solution to Eq. (6). This allows, if necessary, to interrupt the search if some time restrictions must be met.

7. Case study

The proposed case study is based on the aggregated version of the drinking water transport network (DWTN) of the city of Barcelona. This DWTN consists of: 9 water sources, 17 water tanks, 61 actuators (37 valves and 24 pumps), 12 nodes and 25 demands (cf. Fig. 2 for a schematic representation of the network). The details of the system can be found in the technical report [42], including network equations as well as tanks and actuators limits. Besides, the predicted water demand used in the simulations has been obtained from historical water consumption data, and it can be found in the supplementary material of [43]. The robustness of the network model has been enhanced by considering uncertainties in the water demand predictions. In this regard, similarly to [43], the prediction error is bounded in the set

$$\mathcal{W} = \{w(k) \in \mathbb{R}^{25} \ :\ |w(k)| \le \bar{w}\},$$

where the maximum prediction error $\bar{w} \in \mathbb{R}^{25}$ is set as the 5% of the maximum expected demand during the tests, i.e. $\bar{w}_i = 0.05 \max_k d_i(k)$. In the simulations presented below, the values of $w(k)$ have been randomly generated following a uniform distribution bounded within $\mathcal{W}$.

Following the problem statement presented in Section 2, the network actuators have been randomly partitioned into nominal C_n and back-up components C_a, with $|C_n| = 46$ and $|C_a| = 15$ (see Table 1). Note that this artificial division has been carried out with illustrative purposes, however, in real-world applications this partition is specific of each system design. The considered division is illustrated in Fig. 2, where the back-up elements appear highlighted in green.

7.1. Management criteria

The stage cost used to asses the FN performance (Section 2.3) operating in configuration C_i takes into account the following criteria [44,45]:

Table 1
Elements partition.

	Pumps (a_i)	Valves (a_i)
Nominal ($i =$)	3, 5, 9, 10, 11, 15, 20, 21, 22, 23, 24, 29, 33, 34, 36, 38, 42, 48, 53	1, 2, 7, 8, 12, 13, 18, 28, 31, 32, 35, 39, 40, 41, 43, 44, 45, 46, 47, 49, 51, 52, 54, 56, 57, 59, 60, 61
Back-up ($i =$)	4, 17, 19, 27, 55	6, 14, 16, 25, 26, 30, 32, 37, 50, 58

1. **Minimizing water production and transport costs:** This term accounts for the economic costs associated with the drinking water production (water treatment) and transporting (pumping). The performance index to be minimized is described by

   $$f_1(C_i, k) = (\alpha_1^T + \alpha_2^T(k))\Sigma^{[i]}u(k),$$

   where α_1 accounts for the fixed economic cost of the water according to its source (treatment plant, dwell, etc.) and $\alpha_2(k)$ is associated with the economic cost of pumping the water. In the simulations, vector $\alpha_2(k)$ presents a $T = 24$ h cyclic pattern that relates with the daily variations in the electricity rate.
2. **Safety storage term:** The satisfaction of water demands has been imposed as a hard constraint in the network model that should be fulfilled at every time instant. As a consequence, the stored water volume is preferably maintained around a given safety value as a risk prevention mechanism. This concept is formulated as

   $$f_2(k) = (x(k) - x_{sf})^T W_x(x(k) - x_{sf}),$$

   where $x(k) \in \mathbb{R}^{17}$ denotes the water volume in the tanks and $x_{sf} \in \mathbb{R}^{17}$ denotes the safety storage volume. Particularly, the safety volume has been designed as $x_{sf} = 0.75(\bar{x} - \underline{x})$, where $\bar{x}$ and $\underline{x}$ represent the maximum and minimum accepted tank volumes, respectively. Moreover, the weighting matrix is set to $W_x = diag(1/(\bar{x} - \underline{x}))$, in order to penalize the deviation from the safety volume proportionally to the size of each one of the tanks.
3. **Smoothness of the control actions:** The variations of the control signal between consecutive sampling intervals is also penalized. By denoting $\Delta u(k) = u(k) - u(k-1)$, this objective is formulated as

   $$f_3(C_i, k) = \Delta u(k)^T \Sigma^{[i]} W_u \Sigma^{[i]} \Delta u(k),$$

   where the weighting matrix is designed as $W_u = I_{61}$.

Accordingly, the stage cost is made up of a weighted sum of the previous terms

$$l(C_i, k, x(k), u(k)) = \lambda_1 f_1(C_i, k) + \lambda_2 f_2(k) + \lambda_3 f_3(C_i, k),$$

with $\lambda_1 = 1$, $\lambda_2 = 0.05$ and $\lambda_3 = 0.01$.

7.2. Configuration selection criteria

In the sequel, the selection of a new configuration C_{new} after a fault in the components in C_{out} (see Section 3) is performed by minimizing the following multi-objective criteria $J = [J_1,\ J_2]$, where

1. *Objective 1 (J_1):* minimize number of back-up actuators activated after the fault.
2. *Objective 2 (J_2):* minimize the expected performance-loss during the transient induced by the fault.

The multi-objective optimization is addressed considering a lexicographic ordering among the previous objectives, i.e. the optimization of J_1 is infinitely more important than the optimization of J_2.

7.3. Robust MPC tuning parameters

Here, the computation of the tuning parameters required for computing the single-layer MPC control law in Section 4.3 for an arbitrary configuration C_i, as well as for the pair $(M_w^{[i]}, \mathcal{L}_T^{[i]})$ retrieved from solving the robust planner in Section 4.4, are detailed.

The MPC time horizon set is to $N = T = 24$h. For an arbitrary configuration C_i, matrix $M_v^{[i]}$ is designed as an orthonormal basis to the null space $E\Sigma^{[i]}$. In addition, the controller gain $K^{[i]}$ is computed using an LQR design for matrices $(A, \hat{B}^{[i]}, Q, R^{[i]})$, with

$$\hat{B}^{[i]} = B\Sigma^{[i]}M_v^{[i]}, \qquad Q = I_{17},$$
$$R^{[i]} = M_v^{[i]T} diag(1/u_{max}) M_v^{[i]},$$

where u_{max} denotes the maximum flow handling capacity of the actuators. Matrix $R^{[i]}$ has been selected to handle the big differences in the network actuator limits: a_{50} has a maximum value of $15\text{m}^3/\text{s}$, whereas the maximum value of a_7 is of $10^{-5}\text{m}^3/\text{s}$.

7.4. Admissibility criterion

The solution of the robust planner problem in Section 4.4 for the nominal configuration C_n yields the average stage cost $\mathcal{L}_T^{[n]} = 2.2518 \cdot 10^3$. Below, the parameter β that rules the satisfaction of the performance admissibility condition (cf. Section 2.4) is set to $\beta = 1.25$.

7.5. Search for minimal configurations

An offline analysis of the minimal configurations C_l^m that satisfy the performance property after a fault in any of the nominal actuators has been performed. In particular, Algorithm 1 presented in Section 5.1 is used in order to look for the minimal configurations that satisfy $\mathcal{A}^{pf}(C_l^m)$.

The offline tests yielded the following results[1]:

- A fault in $C_{out} = \{a_i\}$ for $i \in \{1, 2, 5, 12, 13, 15, 18, 21, 22, 23, 28, 29, 31, 35, 36, 40, 41, 42, 44, 49, 54, 56, 59, 61\}$ have been identified as critical faults, that is, either there are no combinations of back-up elements for which (17) generates a feasible trajectory, or the attained trajectory has associated cost lower than the threshold. The total time for the offline identification of the above set of critical faults is $t_{crt} = 578.64$ s.
- For a fault in $C_{out} = \{a_i\}$ with $i \in \{7, 8, 9, 10, 11, 33, 38, 39, 43, 45, 46, 48, 52, 60\}$, the solution of (17) is able to generate an admissible trajectory without the need for back-up actuators, i.e. $C_l^m = C_f$. The total time required for the offline identification of the above set of admissible faults is $t_{adm} = 184.94$ s.
- For a fault in $C_{out} = \{a_i\}$ with $i \in \{3, 20, 24, 34, 47, 51, 53, 57\}$, Algorithm 1 must perform a non trivial search for minimal configurations. In this regard, Table 2 shows the number of minimal configurations found, as well as the number (and percentage) of configurations explored by Algorithm 1 out of the possible $2^{15} - 1 = 32.767$ candidate configurations. The total time for the offline search of minimal configurations for the above set of faults is $t_{rec} = 1.332 \cdot 10^4$ s.

The total time for the offline analysis of all possible faults C_{out} in the nominal components is $t_{off} = 1.4086 \cdot 10^4\ \text{s} \approx 3$ h 55 min.

[1] Laptop (Intel i7 1.8 GHz, 16 GB RAM) running Windows 10; optimizations using Yalmip parser and Cplex solver.

Table 2
Minimal configurations.

C_{out}	$\lvert\mathcal{M}^{[f]}\rvert$	Explored confs.	Percentage
a_3	1	16	0.049
a_{20}	17	200	0.610
a_{24}	11	144	0.439
a_{34}	1	17	0.052
a_{47}	2	34	0.104
a_{51}	1	17	0.052
a_{53}	1	17	0.052
a_{57}	1	16	0.049

Table 3
Set of minimal configurations — $C_{out} = \{a_{24}\}$; $\beta = 1.25$.

l	C_l^m
1	$C_f \cup \{a_6, a_{27}\}$
2	$C_f \cup \{a_{17}, a_{27}\}$
3	$C_f \cup \{a_{27}, a_{37}\}$
4	$C_f \cup \{a_{27}, a_{50}\}$
5	$C_f \cup \{a_{27}, a_{55}\}$
6	$C_f \cup \{a_{27}, a_{58}\}$
7	$C_f \cup \{a_6, a_{17}, a_{50}\}$
8	$C_f \cup \{a_{14}, a_{25}, a_{27}\}$
9	$C_f \cup \{a_{25}, a_{26}, a_{27}\}$
10	$C_f \cup \{a_6, a_{14}, a_{17}, a_{25}\}$
11	$C_f \cup \{a_6, a_{17}, a_{25}, a_{26}\}$

7.6. Fault scenario — fault in actuator 24

Here, a fault in actuator $C_{out} = \{a_{24}\}$ is simulated. For this scenario, the set of minimal configurations $\mathcal{M}^{[f]}$ is shown in Table 3. Notice that in Table 3 the configurations have been sorted by taking into account its cardinality. The fault scenario simulated is the following: a fault in actuator 24 appears at $k_f = 39$ h. This fault causes a performance loss of 25% of the actuator capabilities. Moreover, it is assumed that an FDI block detects the fault at $k_d = 43$ h and that the actuator 24 is turned-off.

For the above scenario, Algorithm 2 in Section 6.2 is used for the online computation of the new configuration C_{new}. In this case, the sequential addition of constraints in Algorithm 2 is only imposed for the first optimization objective J_1. The algorithm behaves as follows:

- **Iteration $i = 1$:** the first optimization is launched for the set $\mathbb{P}(C_1^m|a_{24})$, yielding the solution:

  $$\tilde{\mathrm{P}}_1: \; C_{new} = C_1^m, \; (J_1^* = 2 + |C_f|, \; J_2^* = 2.2641 \cdot 10^3).$$

  Hence, further optimizations are subject to the constraint $J_1 \leq 2 + |C_f|$. Notice that, because $|C_l^m| > 2 + |C_f|$ for $l > 6$ (cf. Table 3), these optimizations are guaranteed to be infeasible, and thus the search must only continue in the sets $\mathbb{P}(C_l^m|a_{24})$, $l \in \{2, \ldots, 6\}$.
- **Iterations $i \in \{2, \ldots, 6\}$:** The results obtained in the successive optimizations are

  $$\tilde{\mathrm{P}}_2: \; C_{new} = C_2^m \quad (J_1^* = 2 + |C_f|, \; J_2^* = 2.2528 \cdot 10^3),$$
  $$\tilde{\mathrm{P}}_3: \; C_{new} = C_2^m \quad (J_1^* = 2 + |C_f|, \; J_2^* = 2.2953 \cdot 10^3),$$
  $$\tilde{\mathrm{P}}_4: \; C_{new} = C_4^m \quad (J_1^* = 2 + |C_f|, \; J_2^* = 2.2419 \cdot 10^3),$$
  $$\tilde{\mathrm{P}}_5: \; C_{new} = C_4^m \quad (J_1^* = 2 + |C_f|, \; J_2^* = 2.2658 \cdot 10^3),$$
  $$\tilde{\mathrm{P}}_6: \; C_{new} = C_4^m \quad (J_1^* = 2 + |C_f|, \; J_2^* = 2.2589 \cdot 10^3).$$

Accordingly, $C_{new} = C_4^m$, since, for the same number of back-up actuators ($J_1^* = 2$), generates the best expected average cost

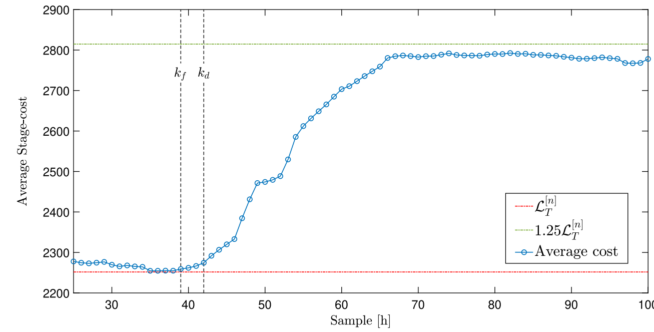

Fig. 3. Average stage cost — Fault in actuator 24.

during the transient. Previous optimizations are run online in a total time $t_{on} = t_{J_1} + t_{J_2} = 4.812$ s (well below the 1 h sampling time), where the total times for the computation of J_1 and J_2 are $t_{J_1} = 1.912$ s; $t_{J_2} = 2.90$ s.

Fig. 3 depicts the evolution of the average stage cost during the fault scenario. This cost has been computed at time k by averaging the cost obtained for the time interval $[k-T+1, k]$. In this figure, it can be seen how, firstly, the cost stabilizes at $\mathcal{L}_T^{[n]}$ and how, after the transient induced by the fault plus reconfiguration, the stage cost stabilizes below the admissibility threshold $\beta\mathcal{L}_T^{[n]}$ ($\beta = 1.25$). It must be remarked that there may be some discrepancies between the cost values obtained by the real system trajectory and the values generated by the robust planner, since the planner is computed for the non-disturbed system.

Besides, Figs. 4(a) to 4(c) show the time evolution of the volume of tanks 4, 7 and 11 during the previous fault scenario, respectively. In these figures, it can be appreciated how: before the fault, the system stabilizes over the nominal configuration reference (red dashed line); and after the fault detection, over the new planner trajectory corresponding with the new configuration C_{new} (green dashed line). Additionally, Fig. 5 displays the evolution of several actuators of the network. In particular, Fig. 5(a) shows how pump 23 increases its power with the new configuration, yielding a worse economic performance. Besides, in Fig. 5(b) it can be seen how the back-up actuator 27 is turned-on after the detection time, whereas Fig. 5(c) shows the modification in the periodic operation of valve 51 before and after the configuration change.

8. Conclusions

This paper presents a methodology for the robust system reconfiguration with back-up components problem in FNs, which combines an offline analysis on the configurations that yield and admissible steady-state operation with an online search for the optimal configuration required to cope with the transient induced by the fault. One of the main difficulties relates with the tuning of the parameters that constitute the control scheme, since small modifications can significantly affect the existence of an admissible control law for a given system configuration. In this regard, the proposed solution seeks for a good trade-off between the optimality in the new configuration selection and the computational complexity of the approach.

Finally, the next natural step in the development of these reconfiguration techniques would be to consider possible non-linearities in the system model. Additionally, partitioning approaches can be used to split the network into physically redundant areas, so the search for hardware-redundant components can be conducted in each area independently.

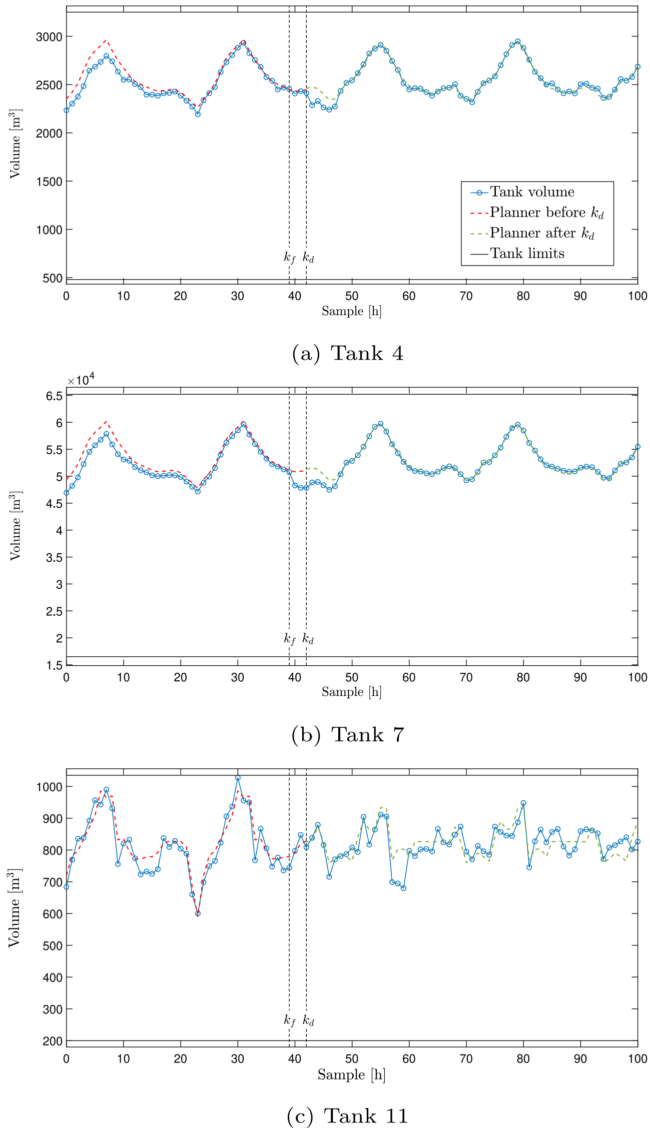

Fig. 4. Tank volumes evolution — Fault in actuator 24.

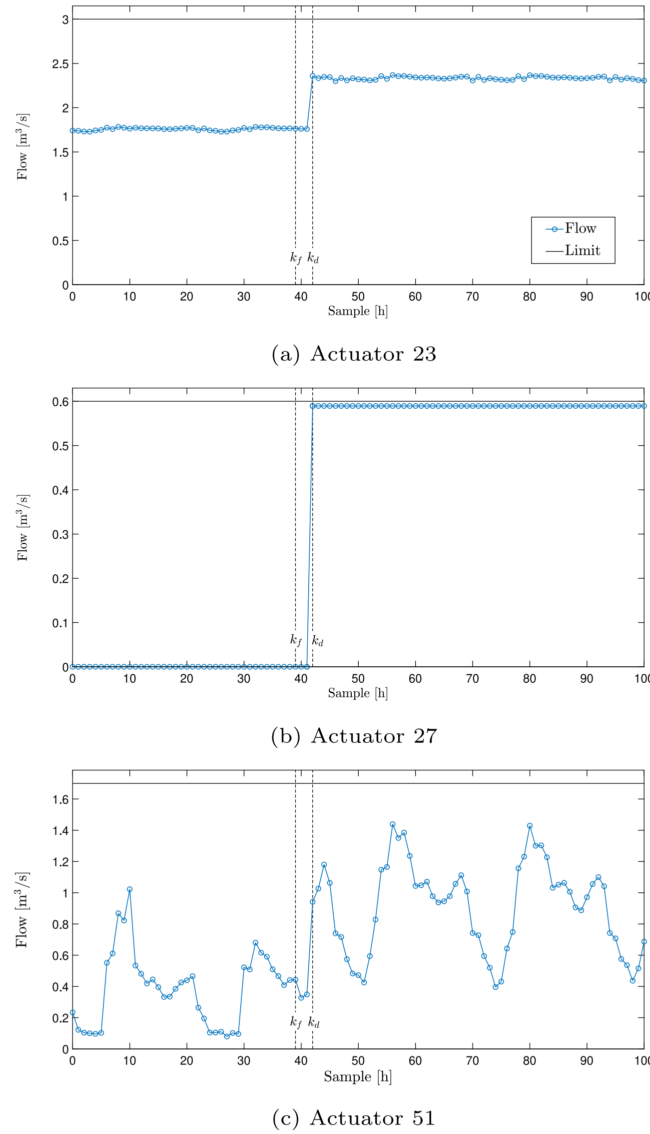

Fig. 5. Actuators evolution — Fault in actuator 24.

CRediT authorship contribution statement

Carlos Trapiello: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing. **Vicenç Puig:** Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing. **Gabriela Cembrano:** Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing.

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.

Data availability

The data that has been used is confidential.

Appendix. Proof of Corollary 1

Lemma 2 (*P-difference [46]*). *Given the zonotope $\mathcal{Z} = \langle c, H\rangle \subset \mathbb{R}^n$, with $c \in \mathbb{R}^n$ and $H \in \mathbb{R}^{n\times z}$, and the polyhedron $\mathcal{S} = \{x \in \mathbb{R}^n : Lx \leq l\} \subseteq \mathbb{R}^n$, with $l \in \mathbb{R}^m$ and $L \in \mathbb{R}^{m\times n}$, then $\mathcal{S} \ominus \mathcal{Z} = \{x \in \mathbb{R}^n : Lx \leq l - Lc - |LH|\mathbf{1}_z\}$.*

Proof of Corollary 1. Starting from the robust planner (7) for the configuration C_i. Matrix $M_w^{[i]}$ affects the sets

$$\mathcal{X}^{[i]}(N) = \mathcal{X} \ominus \mathcal{R}^{[i]}(N), \qquad \mathcal{U}^{[i]}(N) = \mathcal{U} \ominus \mathcal{R}_u^{[i]}(N), \tag{A.1}$$

with $\mathcal{R}^{[i]}(N)$ and $\mathcal{R}_u^{[i]}(N)$ the Nth iteration of

$$\mathcal{R}^{[i]}(j) = \bigoplus_{0}^{j-1} \mathcal{Q}^{[i]}(l), \quad \mathcal{Q}^{[i]}(l) = (A + \hat{B}^{[i]}K^{[i]})^l \hat{B}_w^{[i]}\mathcal{W},$$
$$\mathcal{R}_u^{[i]}(j) = M_w^{[i]}\mathcal{W} \oplus M_v^{[i]}K^{[i]}\mathcal{R}^{[i]}(j).$$

By recalling that $\mathcal{W}$ is a unitary zonotope and that $\hat{B}_w^{[i]} = B_w + B\Sigma^{[i]}M_w^{[i]}$, it follows that $\mathcal{R}^{[i]}(j)$ and $\mathcal{R}_u^{[i]}(j)$ are also zonotopic sets which can be rewritten as

$$\begin{aligned}
&\mathcal{R}^{[i]}(j)=\langle 0, G_a(j)+G_b(j)(I_j\otimes M_w^{[i]})\rangle,\\
&\mathcal{R}_u^{[i]}(j)=\langle 0, [M_w^{[i]},\ \ M_v^{[i]}K^{[i]}\big(G_a(j)+G_b(j)(I_j\otimes M_w^{[i]})\big)]\rangle,
\end{aligned}\tag{A.2}$$

with $G_a(j)$ and $G_b(j)$ the jth elements of the recursion

$$G_a(j+1)=[(A+\hat{B}^{[i]}K^{[i]})G_a(j),\ B_w],\qquad G_a(0)=0,$$

$$G_b(j+1)=[(A+\hat{B}^{[i]}K^{[i]})G_b(j),\ B],\qquad G_b(0)=0.$$

Therefore, from $\mathcal{X}$ and $\mathcal{U}$ in (2) and Lemma 2, the sets in (A.1) are rewritten as

$$\begin{aligned}
&\mathcal{X}^{[i]}(N)=\{x(k)\ :\ H_x x(k)\le h_x^{[i]}\},\\
&\mathcal{U}^{[i]}(N)=\{u(k)\ :\ H_u u(k)\le h_u^{[i]}\},
\end{aligned}\tag{A.3}$$

where

$$\begin{aligned}
h_x^{[i]}&=h_x-|H_xG_a(N)|\mathbf{1}_{Nn_w}\\
&\quad-|H_xG_b(N)(I_N\otimes M_w^{[i]})|\mathbf{1}_{Nn_w},\\
h_u^{[i]}&=h_u-|H_uM_v^{[i]}K^{[i]}G_a(N)|\mathbf{1}_{Nn_w}\\
&\quad-|H_u(\tilde{I}+M_v^{[i]}K^{[i]}G_b(N))(I_N\otimes M_w^{[i]})|\mathbf{1}_{Nn_w},
\end{aligned}\tag{A.4}$$

and $\tilde{I}=[I_{n_u}\ 0_{n_u\times n_u(N-1)}]$.

In addition, (A.4) can be reformulated as linear constraints in $M_w^{[i]}$ by bounding the absolute value of the matrices from above through the introduction of the variable matrices Ω_x and Ω_u as

$$\begin{aligned}
h_x^{[i]}&=h_x-|H_xG_a(N)|\mathbf{1}_{Nn_w}-\Omega_x\mathbf{1}_{Nn_w},\\
h_u^{[i]}&=h_u-|H_uM_v^{[i]}K^{[i]}G_a(N)|\mathbf{1}_{Nn_w}-\Omega_u\mathbf{1}_{Nn_w},
\end{aligned}\tag{A.5}$$

altogether with the set of constraints

$$\begin{aligned}
H_xG_b(N)(I_N\otimes M_w^{[i]})&\le\Omega_x,\\
-H_xG_b(N)(I_N\otimes M_w^{[i]})&\le\Omega_x,\\
H_u(\tilde{I}+M_v^{[i]}K^{[i]}G_b(N))(I_N\otimes M_w^{[i]})&\le\Omega_u,\\
-H_u(\tilde{I}+M_v^{[i]}K^{[i]}G_b(N))(I_N\otimes M_w^{[i]})&\le\Omega_u.
\end{aligned}\tag{A.6}$$

Hence, by means of (A.5)–(A.6) and imposing the satisfaction of (10b), then $M_w^{[i]}$ can be set as an optimization variable in the robust-planner while preserving the convexity of the optimization problem. □

References

[1] J.M. Grosso Pérez, On Model Predictive Control for Economic and Robust Operation of Generalised Flow-Based Networks (Ph.D. thesis), Universitat Politècnica de Catalunya, 2015.
[2] F. Kennel, D. Görges, S. Liu, Energy management for smart grids with electric vehicles based on hierarchical MPC, IEEE Trans. Ind. Inform. 9 (3) (2012) 1528–1537.
[3] C. Ocampo-Martinez, V. Puig, G. Cembrano, J. Quevedo, Application of predictive control strategies to the management of complex networks in the urban water cycle [applications of control], IEEE Control Syst. Mag. 33 (1) (2013) 15–41.
[4] D. Angeli, M.A. Müller, Economic model predictive control: Some design tools and analysis techniques, in: Handbook of Model Predictive Control, Springer, 2019, pp. 145–167.
[5] M. Pereira, D.M. De La Pena, D. Limón, Robust economic model predictive control of a community micro-grid, Renew. Energy 100 (2017) 3–17.
[6] Y. Wang, V. Puig, G. Cembrano, Non-linear economic model predictive control of water distribution networks, J. Process Control 56 (2017) 23–34.
[7] L. Würth, R. Hannemann, W. Marquardt, A two-layer architecture for economically optimal process control and operation, J. Process Control 21 (3) (2011) 311–321.
[8] D. Angeli, R. Amrit, J.B. Rawlings, On average performance and stability of economic model predictive control, IEEE Trans. Automat. Control 57 (7) (2011) 1615–1626.
[9] R. Huang, E. Harinath, L.T. Biegler, Lyapunov stability of economically oriented NMPC for cyclic processes, J. Process Control 21 (4) (2011) 501–509.
[10] J.H. Lee, S. Natarajan, K.S. Lee, A model-based predictive control approach to repetitive control of continuous processes with periodic operations, J. Process Control 11 (2) (2001) 195–207.
[11] M. Staroswiecki, On reconfigurability with respect to actuator failures, IFAC Proc. Vol. 35 (1) (2002) 257–262.
[12] M. Staroswiecki, G. Hoblos, A. Aitouche, Sensor network design for fault tolerant estimation, Internat. J. Adapt. Control Signal Process. 18 (1) (2004) 55–72.
[13] M. Staroswiecki, On reconfiguration-based fault tolerance, in: 18th Mediterranean Conference on Control and Automation, MED'10, IEEE, 2010, pp. 1681–1691.
[14] B. Heiming, J. Lunze, Definition of the three-tank benchmark problem for controller reconfiguration, in: 1999 European Control Conference, ECC, IEEE, 1999, pp. 4030–4034.
[15] D. Mignone, Control and Estimation of Hybrid Systems with Mathematical Optimization (Ph.D. thesis), ETH Zurich, 2002.
[16] C. Trapiello, V. Puig, G. Cembrano, Reconfiguration of large-scale systems using back-up components, Comput. Chem. Eng. 149 (2021) 107288.
[17] K. Balzereit, O. Niggemann, Automated reconfiguration of cyber-physical production systems using satisfiability modulo theories, in: 2020 IEEE Conference on Industrial Cyberphysical Systems, Vol. 1, ICPS, IEEE, 2020, pp. 461–468.
[18] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, Diagnosis and Fault-Tolerant Control, Vol. 2, Springer.
[19] D.Q. Mayne, J.B. Rawlings, C.V. Rao, P.O. Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (6) (2000) 789–814.
[20] L. Chisci, J.A. Rossiter, G. Zappa, Systems with persistent disturbances: Predictive control with restricted constraints, Automatica 37 (7) (2001) 1019–1028.
[21] D. Limón, I. Alvarado, T. Alamo, E.F. Camacho, MPC for tracking piecewise constant references for constrained linear systems, Automatica 44 (9) (2008) 2382–2387.
[22] D. Limon, M. Pereira, D.M. De La Peña, T. Alamo, J.M. Grosso, Single-layer economic model predictive control for periodic operation, J. Process Control 24 (8) (2014) 1207–1224.
[23] J. Quevedo, V. Puig, G. Cembrano, J. Blanch, J. Aguilar, D. Saporta, G. Benito, M. Hedo, A. Molina, Validation and reconstruction of flow meter data in the Barcelona water distribution network, Control Eng. Pract. 18 (6) (2010) 640–651.
[24] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, J. Schröder, Diagnosis and Fault-Tolerant Control, vol. third ed., Springer, 2016.
[25] S.X. Ding, Advanced Methods for Fault Diagnosis and Fault-Tolerant Control, Springer, 2021.
[26] J. Quevedo, H. Chen, M. Cugueró, P. Tino, V. Puig, D. García, R. Sarrate, X. Yao, Combining learning in model space fault diagnosis with data validation/reconstruction: Application to the Barcelona water network, Eng. Appl. Artif. Intell. 30 (2014) 18–29.
[27] M.A. Cugueró-Escofet, J. Quevedo, C. Alippi, M. Roveri, V. Puig, D. García, F. Trovò, Model-vs. Data-based approaches applied to fault diagnosis in potable water supply networks, J. Hydroinform. 18 (5) (2016) 831–850.
[28] T. Escobet, R. Sarrate, R. Comasolivas, Fault diagnosis, in: V. Puig, C. Ocampo-Martinez, R. Perez, G. Cembrano, J. Quevedo, T. Escobet (Eds.), Real-Time Monitoring and Operational Control of Drinking-Water Systems, Springer International Publishing, 2017, pp. 195–224.
[29] F.A. Bayer, M. Lorenzen, M.A. Müller, F. Allgöwer, Robust economic model predictive control using stochastic information, Automatica 74 (2016) 151–161.
[30] F.A. Bayer, M.A. Müller, F. Allgöwer, Min-max economic model predictive control approaches with guaranteed performance, in: 2016 IEEE 55th Conference on Decision and Control, CDC, IEEE, 2016, pp. 3210–3215.
[31] Y. Wang, D. Muñoz De la Peña, V. Puig, G. Cembrano, Robust economic model predictive control based on a periodicity constraint, Internat. J. Robust Nonlinear Control 29 (11) (2019) 3296–3310.
[32] M. Pereira, D.M. de la Pena, D. Limón, I. Alvarado, T. Alamo, Robust model predictive controller for tracking changing periodic signals, IEEE Trans. Automat. Control 62 (10) (2016) 5343–5350.
[33] D.Q. Mayne, M.M. Seron, S. Raković, Robust model predictive control of constrained linear systems with bounded disturbances, Automatica 41 (2) (2005) 219–224.
[34] I. Alvarado, D. Limon, D.M. de la Pena, T. Alamo, E. Camacho, Enhanced ISS Nominal MPC Based on Constraint Tightening for Constrained Linear Systems, IET, 2010.
[35] Y. Wang, D.M. de la Peña, V. Puig, G. Cembrano, Robust periodic economic model predictive control using probabilistic set invariance for descriptor systems, IFAC-PapersOnLine 51 (20) (2018) 436–441.
[36] J. Grosso, C. Ocampo-Martínez, V. Puig, B. Joseph, Chance-constrained model predictive control for drinking water networks, J. Process Control 24 (5) (2014) 504–516.
[37] M. Nassourou, J. Blesa, V. Puig, Robust economic model predictive control based on a zonotope and local feedback controller for energy dispatch in smart-grids considering demand uncertainty, Energies 13 (3) (2020) 696.

[38] A. Bemporad, M. Morari, Control of systems integrating logic, dynamics, and constraints, Automatica 35 (3) (1999) 407–427.
[39] M.V. Kothare, V. Balakrishnan, M. Morari, Robust constrained model predictive control using linear matrix inequalities, Automatica 32 (10) (1996) 1361–1379.
[40] C.V. Rao, S.J. Wright, J.B. Rawlings, Application of interior-point methods to model predictive control, J. Optim. Theory Appl. 99 (3) (1998) 723–757.
[41] T.L. Santos, J.E. Normey-Rico, D. Limón, Explicit input-delay compensation for robustness improvement in MPC, IFAC Proc. Vol. 43 (2) (2010) 384–389.
[42] V. Fambrini, C. Ocampo-Martinez, Modelling and Decentralized Model Predictive Control of Drinking Water Networks, Tech. Rep. IRI-TR-04-09, Institut de Robotica I Informatica Industrial (CSIC-UPC), 2009.
[43] M. Pereira, D.M. de la Peña, D. Limon, I. Alvarado, T. Alamo, Application to a drinking water network of robust periodic MPC, Control Eng. Pract. 57 (2016) 50–60.
[44] C. Ocampo-Martinez, V. Puig, G. Cembrano, R. Creus, M. Minoves, Improving water management efficiency by using optimization-based control strategies: The Barcelona case study, Water Sci. Technol.: Water Supply 9 (5) (2009) 565–575.
[45] C. Ocampo-Martinez, D. Barcelli, V. Puig, A. Bemporad, Hierarchical and decentralised model predictive control of drinking water networks: Application to Barcelona case study, IET Control Theory Appl. 6 (1) (2012) 62–71.
[46] J.K. Scott, R. Findeisen, R.D. Braatz, D.M. Raimondo, Input design for guaranteed fault diagnosis using zonotopes, Automatica 50 (6) (2014) 1580–1589.