

Methodology for Actuator Fault Tolerance Evaluation of Linear Constrained MPC: Application to the Barcelona Water Network

Deneb Robles, Vicenç Puig, Carlos Ocampo-Martinez and Luis E. Garza

Abstract—This paper presents a methodology to evaluate the admissibility of actuator fault configurations (AFC) when Linear Constrained Model Predictive Control (LCMPC) is used. The methodology combines the use of structural and feasibility analysis. Structural analysis allows to evaluate the loss of reachability after a fault occurrence. The results of the structural analysis can be complemented with the feasibility analysis of the MPC problem taking into account the effect of actuator constraints after the fault occurrence. Additionally, a degradation analysis of the system performance can also be included. The proposed methodology is tested in the Barcelona water network.

I. INTRODUCTION

The problem of evaluating the admissibility of actuator fault configurations considering a linear predictive/optimal control law with constraints is studied in this paper. This problem has been already treated in the literature for the case of LQR problem without constraints [1][2], due to the existence of analytical solution. However, constraints (on states and control signals) are always present in real industrial control problems and could be easily handled using Linear Constrained Model Predictive Control (LCMPC). But in general, an analytical solution for these kind of control laws does not exist, which makes difficult to do this type of analysis. The method proposed in this paper is not of analytical but of computational nature. It follows the idea proposed by [3] in which the calculation of the control law for a predictive/optimal controller with constraints can be divided in two steps: first, the calculation of solutions set that satisfies the constraints (feasible solutions) and second, the optimal solution determination. Following this approach, the authors have already proposed a method based on zonotope-based set computations [4].

Faults in actuators cause changes in the set of feasible solutions since constraints on the control signals vary. As a consequence, this causes that the set of admissible solutions for the control objective could be empty. Therefore, the admissibility of the control law facing the actuator faults can be determined knowing the feasible solutions set. One of the aims of this paper is to provide a methodology to evaluate

This work has been partially funded by the Spanish Ministry of Science and Technology through the following project SHERECS (Ref. DPI2011-26243), WATMAN (Ref. DPI2009-13744), and by European Commission through contract i-Sense (Ref. FP7-ICT-2009-6-270428).

D. Robles, V. Puig and C. Ocampo-Martinez are with Advanced Control Systems Group (SAC) at Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Llorens i Artigas, 4-6, 08028 Barcelona, Spain. e-mail: {drobles, vpuig, cocampo}@iri.upc.edu

D. Robles and L.E. Garza are with the Tecnológico de Monterrey, Campus Monterrey, Av. Eugenio Garza Sada Sur No. 2501 Monterrey, N.L. 64489 México e-mail: legarza@itesm.mx

the admissibility of the control law without evaluating the feasible solutions set.

This paper presents a methodology to evaluate the admissibility of actuator fault configurations when Linear Constrained Model Predictive Control (LCMPC) is used. For a general framework of this type of analysis see [5]. The methodology combines the use of structural and feasibility analysis. In the structural approach, a reachability analysis after a fault occurrence is made. The results of the structural analysis can be complemented with the feasibility analysis of the MPC problem taking into account the effect of actuator constraints after a fault occurrence. Additionally, a degradation analysis of the system performance can also be included. The proposed methodology is tested in the Barcelona water network in order to show its effectiveness and promising results.

The organization of the paper is as follows: Section II establishes the proposal of the first stage in developing a fault tolerant control, which is the evaluation of the network configuration. Section III explains both of the analysis made to the configuration and describes the algorithms used. The case study and results obtained through the proposed approach are shown in Section IV as well. Finally, Section V concludes the paper.

II. PROBLEM STATEMENT

A. Statement of the Control Problem

Consider a linear system in discrete-time with constraints in states and actuators and that is controlled by an LCMPC law. According to [6], the solution of a control problem consists in finding a control law from a given set of *control laws* \mathcal{U} , such that the controlled system achieves the *control objectives* \mathcal{O} while its behaviour satisfies a set of *constraints* \mathcal{C} . Thus, the solution of the problem is completely defined by the triple $\langle \mathcal{O}, \mathcal{C}, \mathcal{U} \rangle$. In the case of an LCMPC, the triple $\langle \mathcal{O}, \mathcal{C}, \mathcal{U} \rangle$ is defined by

$$\mathcal{O} : \min_{\tilde{u}} J(\tilde{x}, \tilde{u}), \quad (1)$$

subject to

$$\mathcal{C} : \begin{cases} x_{k+1} = Ax_k + Bu_k, \\ u_k \in \mathcal{U} & k \in [0, N-1] \subset \mathbb{N}, \\ x_k \in \mathcal{X} & k \in [0, N] \subset \mathbb{N}, \end{cases} \quad (2)$$

where

$$\mathcal{U} \stackrel{\Delta}{=} \{u \in \mathbb{R}^m \mid u_{\min} \leq u \leq u_{\max}\}, \quad (3)$$

$$\mathcal{X} \stackrel{\Delta}{=} \{x \in \mathbb{R}^n \mid x_{\min} \leq x \leq x_{\max}\}, \quad (4)$$

A and B are the system matrices and N denotes the time horizon. The control law belongs to the set \mathcal{U} and it is obtained using the *receding horizon philosophy* [7], [8]. This technique consists in taking only the first value from the sequence \tilde{u} computed at each time instant by solving the previous optimization problem. Initial states x_0 are updated from measurements or state estimation. The objective function J is defined, in general form, as

$$J(\tilde{x}, \tilde{u}) = \phi(x_N) + \sum_{i=0}^{N-1} \Phi(x_i, u_i), \quad (5)$$

where ϕ is a function that constraints the final state value over N and Φ is a function of state and input sequences over a given time horizon N denoted by

$$\begin{aligned} \tilde{x} &= (x_k)_0^N = (x_0, x_1, \dots, x_N), \\ \tilde{u} &= (u_k)_0^{N-1} = (u_0, u_1, \dots, u_{N-1}). \end{aligned} \quad (6)$$

B. Statement of the Fault-tolerant Control Problem

Fault-Tolerant Control (FTC) is concerned with the control of a faulty system. There exists two types of FTC approaches: the active and the passive. The active FTC uses a Fault Detection and Isolation (FDI) module to know the state of each component of the network and based in the diagnosis of the fault, the controller adapts the control law to achieved the established set point. On the other hand, in the passive FTC, the FDI module is not used and the controller is designed to cope with the fault without adaptation.

In the active FTC, there exist two main strategies to adapt the control loop in order to introduce fault tolerance:

- 1) *Fault Accommodation*: Consists in solving the control problem $\langle \mathcal{C}, \hat{\mathcal{C}}_f(\hat{\Theta}_f), \hat{\mathcal{U}}_f \rangle$, being $\hat{\mathcal{C}}_f(\hat{\Theta}_f)$ an estimation of actual system constraints and parameters provided by the FDI module.
- 2) *System Reconfiguration*: Consists in finding a new set of constraints $\mathcal{C}_f(\Theta_f)$ such that the control problem $\langle \mathcal{C}, \mathcal{C}_f(\Theta_f), \mathcal{U}_f \rangle$ can be solved.

In the previous strategies, the subscript f denotes the fault presence.

III. ADMISSIBILITY ANALYSIS METHODOLOGY

A. Introduction

The admissibility analysis methodology proposed in this paper consists of two main algorithms based on *structural analysis* and *constraint satisfaction techniques* that evaluate the impact of a fault occurrence in the control loop. The first algorithm relies on graph theory to analyze the connectivity properties of the system, while the second one evaluates the feasibility of the control law taking into account the system constraints.

Both algorithms consider the set of all subsets of system actuators I , denoted by 2^I . For each subset $J \subseteq I$, the algorithms evaluate whether a given system property, denoted by $P(J)$, is satisfied or not [9]. Thus,

$$P(J) = \begin{cases} 1 & \text{if property is satisfied,} \\ 0 & \text{if property is not satisfied.} \end{cases} \quad (7)$$

This evaluation induces the set of all subsets of I , 2^I , a two-class partition as follows:

$$2^{I+} = \{J \subseteq I; P(J) = 1\}, \quad (8)$$

$$2^{I-} = \{J \subseteq I; P(J) = 0\}. \quad (9)$$

The class 2^{I+} contains all the subsets of the actuators in which a specific property is satisfied.

The admissibility analysis will permit, among other things, to find:

- *Critical components*: i.e., the set of system components that are indispensable to satisfy a determined property.
- *Redundant components*: i.e., the system components which are not critical for the correct functionality of the system, therefore they could be subtracted and the satisfaction of the property will still be guaranteed.
- *Redundancy degree*: consists of the number of extra non-critical components through which the property could be guaranteed. There are two types of redundancy: *weak* (corresponding to larger number of sequential faults that can be tolerated in the best case) and *strong* (that corresponds the smaller number of sequential faults that can be tolerated in the worst-case).

B. Structural Analysis Algorithm

The structural analysis algorithm analyzes the connectivity properties of the system without considering the actual value of the parameters of the components or the limitations of the actuators¹. This algorithm can be used to evaluate the admissibility of a given AFC when the reconfiguration FTC is used, i.e., when an actuator is removed after the fault occurrence and the system should be controlled with the remaining actuators.

The main steps of the structural analysis algorithm implemented in this paper are summarized in Algorithm 1. The algorithm starts building a digraph $G(\mathcal{V}, \mathcal{E})$ of the system S using system model (2). See [10] for details on how to obtain a digraph from the system model. Using the digraph, the *input reachability* of the system for a given AFC will be evaluated. If this property is preserved after the actuator fault occurrence, the AFC is admissible, that is able to tolerate the fault; otherwise is not admissible.

To define the concept of input reachability, first the concept of path in a digraph will be recalled [10].

Definition 3.1 (Path): Given a digraph $G = (\mathcal{V}, \mathcal{E})$, a collection of vertices v_1, v_2, \dots, v_k together with the edges $(v_1, v_2), (v_2, v_3), \dots, (v_{k-1}, v_k)$ placed in sequence, then the ordered set $(v_1, v_2), (v_2, v_3), \dots, (v_{k-1}, v_k)$ is a path from v_1 to v_k . \diamond

Then, it is said that v_i is reachable from v_j if there is a path from v_j to v_i .

Definition 3.2 (Reachable set of a vertex): A reachable set $\mathcal{V}_i(v_j)$ of a vertex v_j is a set \mathcal{V}_i of vertices v_i reachable from the vertex $v_j \in \mathcal{V}$. \diamond

¹Important definitions related to the topic can be found in [6].

Definition 3.3 (Reachable set of a vertices set): A reachable set $\mathcal{V}_i(\mathcal{V}_j)$ as the set of vertices v_i reachable from at least one vertex $v_j \in \mathcal{V}_j$. \diamond

Notice that $\mathcal{V}_i(\mathcal{V}_j)$ is the union of the sets $\mathcal{V}_i(v_j)$ for all $v_j \in \mathcal{V}_j$.

Definition 3.4 (Input reachable system): A system S with a digraph $G = (\mathcal{X} \cup \mathcal{U}, \mathcal{E})$ is input reachable if \mathcal{X} is a reachable set of \mathcal{U} . \diamond

From previous definition follows that if no input u_j can reach a state x_i , either directly or through other states/inputs of the system, then there is no way of controlling the system to satisfy the desired control objective, i.e., the system is uncontrollable.

In order to establish the input reachability of a system S with digraph $G = (\mathcal{V}, \mathcal{E})$, the reachability matrix $R = (r_{ij})$, defined as

$$r_{ij} = \begin{cases} 1 & \text{if } v_i \text{ is reachable from } v_j, \\ 0 & \text{otherwise,} \end{cases} \quad (10)$$

should be computed. According to [10], to determine the reachability matrix there are several algorithms. One of the straightforward way to do so is through the following expression:

$$r_{ij} = \begin{cases} 0 & \text{if } q_{ij} = 0, \\ 1 & \text{if } q_{ij} \neq 0, \end{cases} \quad (11)$$

where $Q = (q_{ij}) = E + E^2 + \dots + E^s$ and E is the interconnection matrix, defined as

$$E = \begin{bmatrix} \bar{A} & \bar{B} & 0 \\ 0 & 0 & 0 \\ \bar{C} & 0 & 0 \end{bmatrix}, \quad (12)$$

where the submatrices $\bar{A}, \bar{B}, \bar{C}$ are boolean representations of the original system matrices A, B, C in space state.

Algorithm 1 Reachability Analysis using Structural Approach

- 1: Obtain the digraph $G = (\mathcal{X} \cup \mathcal{U}, \mathcal{E})$ of the system S using system model for a given AFC
 - 2: From the system digraph $G = (\mathcal{X} \cup \mathcal{U}, \mathcal{E})$, find the reachability matrix R
 - 3: **for** each $x_i \in X, i = 1, \dots, n_x$ **do**
 - 4: **for** each $u_j \in U, j = 1, \dots, n_u$ **do**
 - 5: **if** $r_{ij} = 0$ **then**
 - 6: AFC is *non reachable*
 - 7: **else**
 - 8: AFC is *reachable*
 - 9: **end if**
 - 10: **end for**
 - 11: **end for**
-

In order to find critical and redundant actuators, recall that I is the whole set of the system sensors, which are supposed to be all available when the system is operating. Each subset $J \subseteq I$ represents the actuator network state in which some sensors have been lost. Thus, the set of all possible situations is identified with 2^I (the collection of the subsets of I) and

it can be graphically represented by a multi-level automaton (see [1]) in which

- a node is associated with each subset of I , and is interpreted as a set of available actuators;
- the initial node is associated with the whole set of sensors I ; and
- a transition from one node to another one is associated with changes in the set of the available actuators.

The system is actuator fault tolerant as long as the remaining actuators guarantee the input reachability property. Evaluating the actuator fault tolerance capability of the system consists in computing the *length* of the paths between the initial (all actuators are healthy) and the terminal configuration (i.e., the remaining actuators do not satisfy the input reachability property). The length of the longest path in this automaton and a terminal configuration corresponds to the *weak redundancy degree*, while the length of the shortest path corresponds to the *strong redundancy degree*. A *critical* actuator in a given actuator configuration corresponds to the one that, if it is lost, the remaining actuators do not satisfy the input reachability property; otherwise the actuator is *redundant*.

C. Constraint Satisfaction Algorithm

When an accommodation FTC strategy is used, the actuator is not removed after the fault but is operated under the remaining operating range estimated by the FDI module. To evaluate the admissibility of the control, it is not possible to use the structural analysis algorithm. In this case, the admissibility evaluation problem for a given AFC can be naturally handled as constraint satisfaction problem. Thus, the definition of the *constraint satisfaction problem* (CSP) is presented and some particular details related to this approach are presented and discussed. Then, this approach is applied to the AFC admissibility evaluation.

A CSP on sets can be formulated as a 3-tuple $\mathcal{H} = (\mathcal{W}, \mathcal{D}, \mathcal{Z})$ [11], where

- $\mathcal{W} = \{w_1, \dots, w_n\}$ is a finite set of variables;
- $\mathcal{D} = \{\mathcal{D}_1, \dots, \mathcal{D}_n\}$ is the set of their domains (where, in this paper, $\mathcal{D}_i \subseteq \mathbb{R}, i = 1, \dots, n$);
- $\mathcal{Z} = \{z_1, \dots, z_n\}$ is a finite set of constraints relating variables of \mathcal{W} .

A point solution of \mathcal{H} is a n-tuple $(\tilde{w}_1, \dots, \tilde{w}_n) \in \mathcal{D}$ such that all constraints \mathcal{Z} are satisfied. The set of all point solutions of \mathcal{H} is denoted by $\mathcal{S}(\mathcal{H})$. This set is called the *global solution set*. The variable $w_i \in \mathcal{W}_i$ is *consistent* in \mathcal{H} if and only if

$$\forall w_i \in \mathcal{W}_i \exists (\tilde{w}_1 \in \mathcal{D}_1, \dots, \tilde{w}_n \in \mathcal{D}_n) | (\tilde{w}_1, \dots, \tilde{w}_n) \in \mathcal{S}(\mathcal{H}),$$

with $i = 1 \dots n$. The solution of a CSP is said to be *globally consistent*, if and only if every variable is consistent considering the whole set of constraints. A variable is *locally consistent* if and only if it is consistent with respect to a group of constraints. Thus, the solution of a CSP is said to be locally consistent if all variables are locally consistent.

The admissibility evaluation of an AFC corresponds naturally to a CSP on sets. Algorithm 2 allows the admissibility

evaluation of a given AFC by solving the associated CSP defined by the system equations, the operative limits, inputs and states over N and the initial state.

It is well known that the solution of this kind of problem has a high computational complexity since in order to represent the solutions sets accurately they should be decomposed using subpavings, which implies exponential computational times [11].

However, in order to evaluate the fault tolerance of LCMPC controller after a fault occurrence, it is not required to find the solution of the CSP, but whether the CSP problem has or not solution. In particular, the fact that a CSP does not have solution means that the LCMPC controller is not fault tolerant with respect to the considered fault. Otherwise, it would be fault tolerant.

The existence of the solution of the CSP associated to the tolerance evaluation of LCMPC for a given faulty situation can be proved by solving the following optimization problem:

$$\mathcal{O} : \min 0, \quad (13)$$

subject to

$$\mathcal{C} : \begin{cases} \phi(x_N) + \sum_{i=0}^{N-1} \Phi(x_i, u_i) \leq J_f, \\ x_{k+1} = Ax_k + Bu_k, \\ u_k \in \mathcal{U} \quad k \in [0, N-1] \subset \mathbb{N}, \\ x_k \in \mathcal{X} \quad k \in [0, N] \subset \mathbb{N}, \end{cases} \quad (14)$$

where

$$\mathcal{U} \triangleq \{u \in \mathbb{R}^m \mid u_{\min} \leq u \leq u_{\max}\}, \quad (15)$$

$$\mathcal{X} \triangleq \{x \in \mathbb{R}^n \mid x_{\min} \leq x \leq x_{\max}\}. \quad (16)$$

In the case that this problem is feasible, the CSP has solution and LCMPC controller is fault tolerant with respect to the control objective (5) with a control objective degradation less or equal to J_f , which establishes the admissibility threshold.

Algorithm 2 Feasibility evaluation using CSP

```

1: for  $k = 1$  to  $N$  do
2:    $\mathcal{U}_{k-1} \leftarrow \mathcal{U}$ 
3:    $\mathcal{X}_k \leftarrow \mathcal{X}$ 
4:   end for
5:    $\mathcal{W} \leftarrow \{x_1, x_2, \dots, x_N, u_1, u_2, \dots, u_{N-1}, J\}$ 
6:    $\mathcal{D} \leftarrow \{\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_N, \mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_{N-1}, \mathcal{J}_{\mathcal{A}}\}$ 
7:    $\mathcal{L} \leftarrow \left\{ (x_{k+1} = Ax_k + Bu_k)_0^{N-1}, J(\tilde{x}, \tilde{u}) = \phi(x_N) + \sum_{i=0}^{N-1} \Phi(x_i, u_i) \right\}$ 
8:    $\mathcal{H}_{\mathcal{A}} = (\mathcal{W}, \mathcal{D}, \mathcal{L})$ 
9:    $\mathcal{A} = \text{solve}(\mathcal{H}_{\mathcal{A}})$ 
10:  if  $\mathcal{A} = \emptyset$  then
11:    AFC is non-admissible
12:  else
13:    AFC is admissible
14:  end if

```

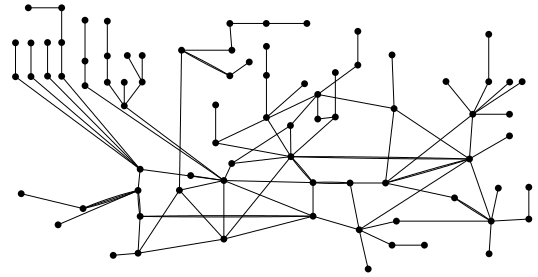


Fig. 2. Graph of the Barcelona DWN

IV. RESULTS

A. Case Study Description

The drinking water network (DWN) of Barcelona is used as the case study of this paper. This network is managed by Aguas de Barcelona S.A (AGBAR), which not only supplies drinking water to Barcelona city but also to the metropolitan area. The sources of water are the rivers Ter and Llobregat. Currently, there are four potabilisation plants: Abrera and Sant Joan Despí plants, which extract water from river Llobregat; Cardedeu plant, which extracts water from river Ter; and Besòs plant, which treats the underground flows from the aquifer of Besòs river. There are also several underground sources (wells) that may provide water through pumping procedures. These sources should currently provide a flow of around $12 \text{ m}^3/\text{s}$. The Barcelona drinking water network is currently comprised of 67 tanks and 121 actuators, which are divided in 46 pumps and 75 valves (see Figure 1). Figure 2 shows the graph of the network. Among of the pumps, five of them draw water from the underground sources, whereas the others are used to carry the water to those places with different elevation (higher city sectors). Moreover, the network has 88 main sectors of consume and 16 water nodes. Both the demand episode and the calibration setup of the network are provided by AGBAR. The current AGBAR control centre has a tele-control system for the network management. The Barcelona water network is also comprised of more than 98 remote stations, which manages in real time about 450 elements: flow meters, pumping stations, valves, chloride dosing instruments, etc.

According to the system model shown in (2), states related to the case study are represented by tanks and water nodes. The former are expressed by constraints regarding the maximal and minimal volume that tanks can store. The latter obey the mass balance law (amount of water entering the node equals to amount of water leaving the node). The control vector corresponds to water flows through actuators, which are constrained by their own physical limitations. Matrices A and B are defined by the physical topology of the network. In order to fulfill the control objectives of the system, the objective function defined in (5) should be minimized based on the cost of the proper operation of the entire network, satisfying the demands of water in each sector of consume. This operational cost of the network is composed by the electric (tariffs of electricity related to the operation of pumps

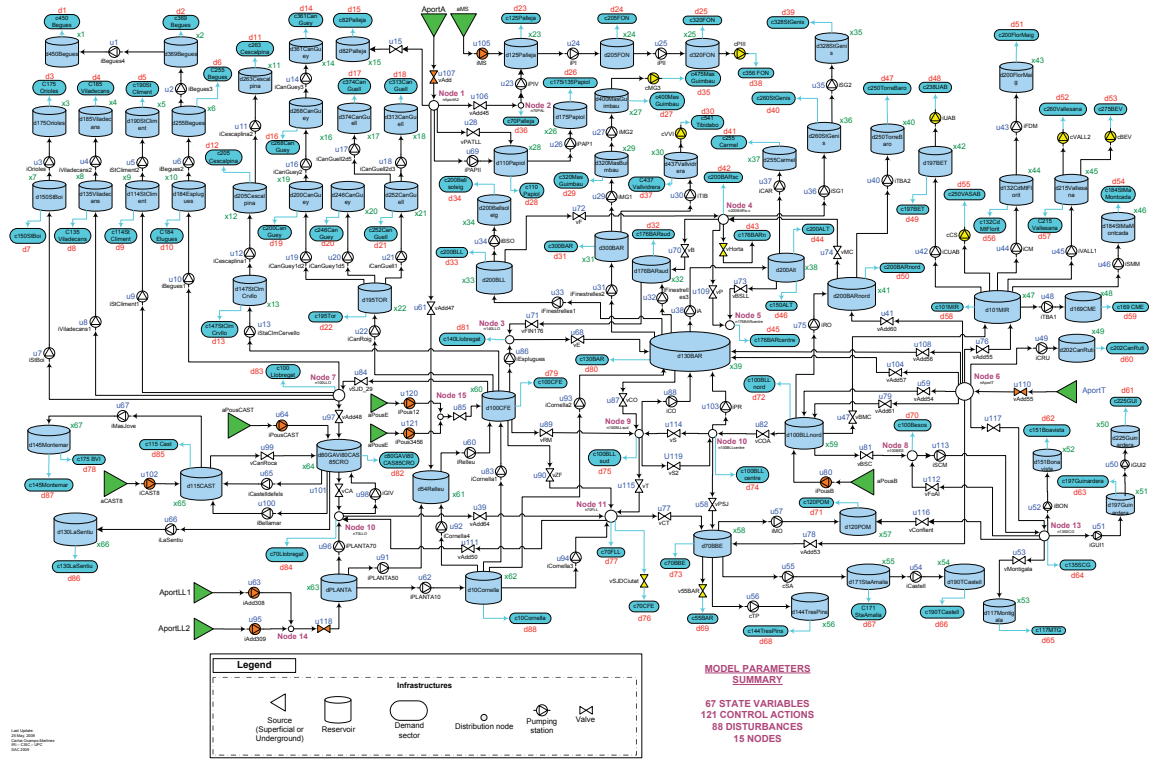


Fig. 1. Barcelona Drinking Water Network

and valves) and water (price of the water at its source) costs.

B. Simulation Results

Five tests/analyses have been made over the case study. Figure 3 shows the sequence of test performed. The first test consists in finding the critical network actuators by means of its structural analysis. These critical actuators are those which, without them (outage), the connectivity of a path is lost. Results of this test, collected in Table I, evidence an important number of critical actuators within the network. This fact is caused by the topology and the way of connecting the network elements: most actuators (either valves or pumps) are just connected between tanks and demands. Therefore, if an actuator fault occurs, the corresponding demand will not be satisfied.

The second analysis done to the Barcelona DWTN consists in the determination of those actuators whose physical constraints limit the capacity of water transport through a certain path. Notice that this analysis do not consider any fault in those actuators. The analysis, performed by using an optimization algorithm, results also in the statement of several alternative paths through which the water transport is possible (or even mandatory) given the constraints of the paths for supplying demands.

Results of this latter analysis yield the determination of another critical actuator: actuator 52 (namely *vBesos-MontCerd*). Notice that the increment in the amount of

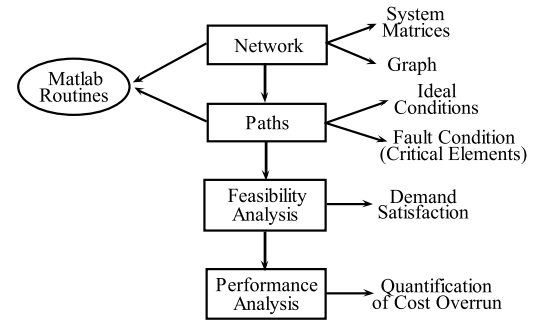


Fig. 3. Flow diagram of the proposed methodology

critical actuators taking in account the actuator physical constraints is not significant.

The third analysis consists in the determination of the optimal paths to reach a selected destination node without considering the constraints of the system, i.e., the *structural optimal paths*. This analysis is performed by using structural algorithm in the way already explained in Section III-B. Results of this test are not provided in this paper due to space constraints.

Fourth analysis naturally consists in the statement of the set of optimal paths including the *objective function* in (5) and the system constraints (in states and actuators). Paths details are not provided here, but the total economic cost of

TABLE I
STRUCTURAL CRITICAL ACTUATORS

No.	Name	No.	Name	No.	Name	No.	Name	No.	Name
122	iAltures	15	iCanGuey2	62	iGuinardera1	30	iPapiol1	44	iVallensana1
10	iBegues1	14	iCanGuey3	60	iGuinardera2	88	iSJD10	8	iViladecans1
6	iBegues2	21	iCanRoig	101	iLaSentiu	7	iStBoi	4	iViladecans2
2	iBegues3	57	iCanRuti	34	iMasGuimbau1	9	iStCliment1	25	vAbrera
1	iBegues4	37	iCarmel	31	iMasGuimbau2	5	iStCliment2	54	vCerdanyola90
32	iBellsolieg	43	iCerdMontflorit	100	iMasJove	40	iStGenis1	63	vMontigala
61	iBonavista	42	iCerdUAB	68	iMntjcStaAmalia	38	iStGenis2	27	vPalleja70
20	iCanGuell1	12	iCesalpina1	69	iMntjcTresPins	13	iStaCImCervello	90	vSJD
17	iCanGuell2d3	11	iCesalpina2	3	iOrioles	45	iStaMaMontcada	104	vSJDtot
16	iCanGuell2d5	82	iCornella100	23	iPalleja1	35	iTibidabo	58	vTer
18	iCanGuey1d2	39	iFlorMaig	24	iPalleja2	56	iTorreBaro1	59	vTerStaColoma
19	iCanGuey1d5	109	iFnestrelles300	26	iPalleja4	65	iTorreoCastell		

TABLE II
NETWORK PERFORMANCE ANALYSIS FOR TEST 5

Actuator No.	Faulty price [e.u.]	Cost overrun [%]
41	514.44	2.43
47	515.94	2.73
74	528.05	5.14
78	557.62	11.03
86	515.08	2.55
89	556.22	10.74
97	510.49	1.64
102	539.87	7.49
103	552.21	9.95

maintaining a functional network with satisfaction of all the demands is 502.25 e.u.²

Finally, performance analysis has been done as fifth test. The test has been performed using the objective function in (5), where the b array is formed with the real values for the actuator constraints. Several simulations using *linprog* have been performed, including a faulty component in each one by setting $b(i) = 0$; if a feasible solution is found, the cost of maintaining an operational network is computed and compared with costs obtained in through previous analyses. The difference in prices shows the impact that a single malfunctioning actuator has in the entire network. Results from this analysis are collected in Table II. Notice that all comparisons are done taking into account an optimal functioning price (under faultless conditions) of 502.25 e.u. Moreover, faulty price denotes the functioning price under faulty conditions.

According to the analysis made to the system, some actuators do not have significant impact in the total performing cost (overrun over 1%, e.g., actuators 28, 29, 33, 64, 71, 80, 81, 85, 87, 94, 107, 108, 113), but there are some others (such as 78 or 89) that induce an important increment in the price, taking in account the daily estimation. These latter actuators are shown in Table II. Degradation in prices obtained with this analysis can be the foundation for the introduction of redundant actuators in the network or an alternative way of fault tolerance strategy.

²In this paper, economic amounts are given in *economic units* (e.u.) rather than real units (€) for confidentiality reasons.

V. CONCLUDING REMARKS

This paper have presented a methodology to evaluate the admissibility of actuator fault configurations when Linear Constrained Model Predictive Control (LCMPC) is used. The methodology combines the use of structural and feasibility analysis. Structural analysis allows to evaluate the loss of connectivity analysis in the presence of a fault. The results of the structural analysis can be complemented with the feasibility analysis of the MPC problem taking into account the effect of actuator constraints after the fault occurrence. Additionally, a degradation analysis in the system performance can also be included. The proposed methodology has been successfully tested in the Barcelona water network. As a further research, additional criteria as actuator reliability will be included in the proposed methodology.

REFERENCES

- [1] M. Staroswiecki, G. Hoblos, and A. Aitouche, "Sensor network design for fault tolerant estimation," *International Journal of Adaptive Control and Signal Processing*, vol. 18, no. 1, pp. 55–72, 2004.
- [2] M. Staroswiecki and D. Berdjag, "A general fault tolerant linear quadratic control strategy under actuator outages," *International Journal of Systems Science*, vol. 41, no. 8, pp. 971–985, 2010.
- [3] F. Lydoire and Ph. Poignet, "Commande non linéaire à horizon fuyant via l'arithmétique d'intervalles," *Conférence Internationale Francophone d'Automatique*, no. 66, 2004.
- [4] C. Ocampo-Martinez, P. Guerra, V. Puig, and J. Quevedo, "Actuator fault tolerance evaluation of linear constrained MPC using zonotope-based set computations," *Journal of Systems & Control Engineering*, vol. 221, no. 6, pp. 915–926, 2007.
- [5] M. Staroswiecki, C. C., and J. Dion, "Fault tolerance evaluation based on the lattice of system configurations," *International Journal of Adaptive Control and Signal Processing*, vol. 26, pp. 54–72, 2012.
- [6] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*. Springer-Verlag Berlin Heidelberg, 2006.
- [7] J. Maciejowski, *Predictive Control with Constraints*. Great Britain: Prentice Hall, 2002.
- [8] J. Rawlings and D. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI (USA): Nob Hill Publishing, 2009.
- [9] M. Staroswiecki, G. Hoblos, and A. Aitouche, "Sensor network design for fault tolerant estimation," *International Journal of Adaptive Control and Signal Processing*, p. in press, 2004.
- [10] D. Šiljak, *Decentralized control of complex systems*. Academic Press, 1991.
- [11] L. Jaulin, M. Kieffer, I. Braems, and E. Walter, "Guaranteed nonlinear estimation using constraint propagation on sets," *International Journal of Control*, vol. 74, no. 18, pp. 1772–1782, 2001.