ADAPTIVE ACTUATORS ALLOCATION STRATEGIES FOR OVERACTUATED NETWORKED CONTROL SYSTEMS

Alessandro Casavola [*,1] **Emanuele Garone** [**,1]

[*] *casavola@deis.unical.it - DEIS, University of Calabria*
[**] *egarone@deis.unical.it - DEIS, University of Calabria*

Abstract: This paper presents a preliminary adaptive actuators allocation scheme that is fault-tolerant with respect to actuator faults or loss of effectiveness for overactuated networked systems. The main idea here is to use an ad-hoc online parameter estimator coupled with an allocation algorithm to perform on-line control reconfiguration. Two simple algorithms are proposed for nonlinear discrete-time systems. The main properties of the algorithms are summarized in the disturbance-free case and their effectiveness shown by means of two numerical examples.

1. INTRODUCTION

Actuators and sensors redundancy is an important issue to deal with for increasing the fault-tolerance and control properties of many real networked systems. It is a very common matter in vehicle applications where, due to safety and performance reasons, a bank of actuators and sensors operating on the vehicle dynamic is often used. Moreover, it is also common in gas, water and electrical distribution and production networks, where such a physical redundancy may be exploited to avoid interruption of services.

In this paper we focus on the problem of actuators allocation for overactuated systems (i.e. systems with physical actuator redundancy). One simple way to solve it is the use of optimal control design (Kwakernaak, 1972). Such an approach achieves both regulation and control effort distribution amongst actuators at the same time. A different approach consists in using a simpler control law that specifies only the total control effort that has to be produced and in separately solving the so-called Control Allocation Problem (CAP) i.e. the one of optimally distributing the desired total control effort over the available actuators. Due to its relevance, especially in flight control systems, CAP has been deeply investigated in last decade and several methods have been proposed: Daisy Chaining (Buffington, Enn, 1996), Direct Control Allocation (Durham, 1993), Convex Optimization Based algorithms (Durahm, 1998; Bošković et al., 2002; Härkegård, 2002; Petersen, Fossen, 2005; Johansen et al., 2005) and PseudoInverse-Redistribution (PIR) methods (Bodson, 2002; Jin, 2005).

One of most actual research interest in overactuated systems is how to exploit their physical redundancy to develop effective reconfigurable control strategy so as to avoid or at least mitigate the effects of actuator failures. A popular way to ensure some level of control reconfiguration is the use of adaptive control laws (Bodson, Groszkiewicz, 1997; Tao et al., 2002). An alternative approach that will be investigated here is the so-called Reconfigurable Control Allocation (RCA) problem (Buffington et al., 1998; Bolender, Doman, 2005). The key idea is depicted in Fig. 1 where supposedly the control law has been

[1] This work has been supported by MIUR Project *Fault Detection and Diagnosis, Control Reconfiguration and Performance Monitoring in Industrial Process.*

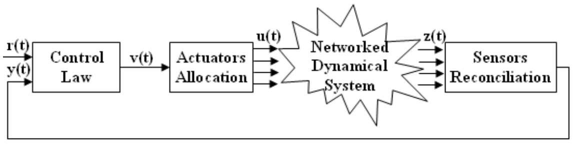

Fig. 1. Control structure with actuators allocation, sensors reconciliation and control computation performed separately

designed on the basis of a virtual system with a minimal number of inputs $v(t)$, fully equivalent to the physical inputs $u(t)$ in generating a desired total control effort. Then, an allocation unit distributes at each time t the control $v(t)$ on the physical actuators $u(t)$, with a law that is allowed to be time-variant. Then, in the case of actuator's faults, control reconfiguration is possible in many cases without altering the control law by simply modifying the distribution of the total control effort $v(t)$ to the remaining no-faulty actuators in $u(t)$. This does not perturb in principle the closed-loop system dynamics because there are several way to distribute the control amongst actuators, all of which making the system behaving in the same way.

In this paper a preliminary adaptive control allocation scheme is proposed able at solving RCA problems for non-linear discrete-time systems without delay. Such systems are subject to actuator's faults or loss of effectiveness. Unlike other works on the topic, here the algorithm is not assumed to know the occurrence of a fault. On the contrary, an adaptive mechanism is used to estimate possible loss of effectiveness and make possible the on-line computation of the allocation's rules by solving a standard constrained QP problem. The main properties of the scheme are summarized and some indications on how to ensure persistence of input excitation are given in order to ensure good estimation properties. For simplicity all development are done in a disturbance-free scenario without considering model uncertainty. All issues related to the robustness properties of the algorithm, data loss and delays are demanded to future studies.

The paper is organized as follows: the problem is stated in Section II. In Section III an adaptive allocation scheme is presented, two different algorithms are proposed and their properties summarized. Finally, two numerical experiments are reported in Section IV and some conclusions end the paper.

2. PROBLEM STATEMENT

2.1 Control Allocation Problem

Let us consider plants whose dynamics is described by the following nonlinear discrete-time state space equation

$$x(t+1) = a(x) + B_u(x)\,u(t), \qquad (1)$$

where $x \in \mathcal{R}^n$ is the state vector and $u(t) \in \mathcal{R}^m$ the control input; $a(x) \in \mathcal{R}^n$ and $B_u(x) \in \mathcal{R}^{n\times m}$ are nonlinear state-dependent functions. The following assumptions are considered

1) The matrix $B_u(x)$ is column-rank deficient: $Rank(B_u(x)) = k < m, \quad \forall x$;
2) The input signal $u(t)$ lies into a compact set Ω, i.e.

$$u(t) \in \Omega := \left\{u \in \mathcal{R}^m \,|\, u^- \le u \le u^+\right\}, \quad (2)$$

where $u^- := [u_1^-, u_2^-, \dots, u_m^-]^T \in \mathcal{R}^m$ and $u^+ := [u_1^+, u_2^+, \dots, u_m^+]^T \in \mathcal{R}^m$.

The assumption 1) allows one to define an equivalent representation of the plant (1)

$$x(t+1) = a(x) + B_v(x)\,v(t), \qquad (3)$$

$$B_v(x)\,v(t) = B_u(x)\,u(t), \qquad (4)$$

where $B_v(x) \in \mathcal{R}^{n\times k}$ is a full column-rank matrix such that its columns are a basis for the subspace defined by the columns of $B_u(x)$ and $v(x) \in \mathcal{R}^k$ is the virtual control input. Hereafter, the state space equation (3) will be referred to as the *virtual plant* while (4) the *parity equation* of the system, which defines the analytical relationship between the virtual and applied commands. Note that, in such a scheme, the virtual control input $v(t)$ represents the *desired total control effort* we want to apply to the plant. In the sequel, we will assume that such a signal $v(t)$ is provided at each time instant by the control law. On the basis of the overall system description (3)-(4) under the actuator constraints (2), the following problem can be stated:

Control allocation problem (CAP) - *Given a virtual input $v(t) \in \mathcal{R}^k$ compute a command input $u(t) \in \mathcal{R}^m$ such that (2) and (4) are satisfied.*

Such a problem has been extensively studied in recent years and several numerical procedures for its solution have been proposed (Bodson, 2002; Johansen et al., 2005). Note that:

- Many previous works on the topic re-arrange the equation (4) as follows

$$\begin{aligned} v(t) &= B(x)\,u(t) \\ B_u(x) &= B_v(x)\,B(x) \end{aligned} \qquad (5)$$

where $B(x) \in \mathcal{R}^{k\times m}$ is a factorization of $B_u(x)$

$$B(x) = \left(B_v^T(x)B_v(x)\right)^{-1} B_v^T(x)B_u(x). \quad (6)$$

- The **CAP** could not admit any solution due to the actuators saturation constraints (2). In such a case, the **CAP** can be relaxed by requiring to compute a command $u(t)$ such that $B_u(x)u(t)$ is somehow close to $B_v(x)v(t)$ (e.g. by evaluating at each time instant the numerical value of $\|B_u(x)u(t) - B_v(x)v(t)\|$);
- The analytical redundancy, i.e. $Rank(B_u(x)) = k < m$, implies that in principle there exists a set of admissible commands u solution for the **CAP**. This fact can be exploited to comply with other specifications besides the **CAP** requirements.

A common way to solve **CAP** at time t is that of minimizing the quadratic optimization problem

$$\begin{aligned} &u(t) \triangleq \arg\min_{s,u} \|s\|^2_{Q_s} + \|u\|^2_{R_u}, \\ &B_v(x(t))v(t) = B_u(x(t))u + s, \\ &u \in \Omega. \end{aligned} \tag{7}$$

The slack-variable s is used to enlarge the set of solutions in the parity equation (4) and it allows the achievement of approximating allocations. When zero, a perfect allocation is achieved. On the contrary, the penalty on u is optional and it is used to minimize the actuators' effort when many solutions are possible. It is well-known that an explicit solution to this optimization problem can be found in the unconstrained case while it does not exist in the general case. However, in order to reduce computational burdens, several efficient algorithms based on the semi-explicit solution have been proposed in the last years (Johansen et al., 2005; Jin, 2005). For the purposes of this paper, it is important to notice here that computational efficiency obtained through explicit approaches is paid in term of a reduction of flexibility w.r.t. reconfiguration issues (Johansen et al., 2005).

2.2 Fault Modeling

Here we desire to take into account possible actuator faults, therefore we suppose that the plant dynamics is corrupted by unpredictable events which alter the nominal behavior of the system. The aim is to make use of the input analytical redundancy to reconfigure the actuator allocation in such a way that the fault become ineffective.

In this paper we will focus only on the class of faults describing effectiveness variations of the actuators. The effect of a fault event is then to change in percentage the nominal gain of some actuator signal. Such a kind of fault can be naturally formalized in a multiplicative fashion

$$x(t+1) = a(x) + B_u(x)\Delta(t)\, u(t), \tag{8}$$

where $\Delta(t) = diag\,\{\delta_1(t), \delta_2(t), \ldots, \delta_m(t)\}$ is the so-called Effectiveness Matrix and $\delta_i(t) \in \mathcal{R}, i = 1, \ldots, m$ are piecewise constant sequences representing the effectiveness of any single actuator. Notice that, in absence of fault occurrences, $\Delta(t) = I$. Moreover, the parity equation (4) becomes

$$B_v(x)\, v(t) = B_u(x)\Delta(t)\, u(t). \tag{9}$$

Then, the problem we want to solve can be stated as follows

$\mathcal{F}$ - Tolerant Control Allocation Problem ($\mathcal{F}$-TCAP) - *Given the virtual plant (8) and a virtual input $v(t) \in \mathcal{R}^k$, find a command input $u(t) \in \mathcal{R}^m$ such that (2) and (9) hold true.*

3. TWO-STEPS PROCEDURE

It may simply be observed that the knowledge of the Effectiveness Matrix $\Delta(t)$ makes **$\mathcal{F}$-TCAP** be reduced to a more simple **CAP**. This allows us to propose the following adaptive **two steps method** to solve **$\mathcal{F}$-TCAP** at each time t:

Step 1: Compute the diagonal matrix $\hat{\Delta}(t)$, the best estimate of $\Delta(t)$ at time t, based on records of N past system's measures.

Step 2: Solve the **CAP** defined by (2) and (9) by assuming (certainty equivalence hypothesis) $\Delta(t) = \hat{\Delta}(t)$.

There is an huge literature both on online parameters estimation and allocation problems. Many of the existing algorithms solving the two problems can be arranged in this general scheme.

3.1 A simple two-steps algorithm

Hereafter, a very simple two-steps algorithm is proposed by using quadratic programming arguments.

Step 1: - Estimate of $\hat{\Delta}(t)$

In order to estimate the Effectiveness Matrix it is convenient to rewrite things in terms of the increment matrix

$$\hat{\Gamma}(t) \triangleq \hat{\Delta}(t) - \hat{\Delta}(t-1) \tag{10}$$

defined as the diagonal matrix

$$\hat{\Gamma} \triangleq diag\{\hat{\gamma}_1, \hat{\gamma}_2, \ldots, \hat{\gamma}_m\} \in \mathcal{R}^m$$

of loss-of-effectiveness actuator increments $\hat{\gamma}_i(t) = \hat{\delta}_i(t) - \hat{\delta}_i(t-1),\ i = 1, \ldots, m$.

We are especially interested in algorithms able at detecting constant or slow-varying actuator faults or loss of effectiveness, that is in determining matrices $\hat{\Delta}(t)$ that "matches as much as possible" the measured signals of the plant in the last N time instants, with N arbitrarily chosen. This corresponds to solutions which minimize the entries of $\hat{\Gamma}$. In principle, such a strategy corresponds to the sequential solution of the following two least-squares problems:

$$\begin{array}{l} \hat{s}_i(t) \triangleq \arg\min_{s_i,\Gamma} \sum_{i=1}^{N} \|s_i\|^2_{Q_i} \\ x(t-i+1) - a(x(t-i)) \\ -B_u(x(t-i)) \left[\Gamma + \hat{\Delta}(t-1)\right] u(t-i) = s_i \\ \qquad\qquad\qquad\qquad i = 1,\ldots,N \end{array} \quad (11)$$

and, once $\hat{s}_i(t), i = 1,\ldots,N$ are obtained,

$$\begin{array}{l} \hat{\Gamma}(t) \triangleq \arg\min_{\Gamma} \|\text{vect}(\Gamma)\|^2_R \\ x(t-i+1) - a(x(t-i)) \\ -B_u(x(t-i)) \left[\Gamma + \hat{\Delta}(t-1)\right] u(t-i) = \hat{s}_i(t), \\ \qquad\qquad\qquad\qquad i = 1,\ldots,N \end{array} \quad (12)$$

where $\text{vect}(\Gamma) = [\gamma_1, \gamma_2, \ldots, \gamma_m]^T \in \mathcal{R}^m$ and $s_i \in \mathcal{R}^n, i = 1,\ldots,N$ slack vectors and $R = R' > 0$ and $Q_i = Q_i' > 0, i = 1,\ldots,N$ consistent weighting matrices.

The choice of N has an important role in such a computation: picking a small value of N means having less or no information and in turn bad parameters estimation results. On the contrary, a large value of N yields to long computation and reconfiguration times. A reasonable choice is $m/n \leq N \leq 2m$. From a practical point of view, an approximate numerical solution to (11) and (12) can be obtained by combining those two optimization problems into a unique mixed weighted least-squares problem

$$\begin{array}{l} \hat{\Gamma}(t) \triangleq \arg\min_{s_i,\Gamma} \sum_{i=1}^{N} \|s_i\|^2_{Q_i} + \|\text{vect}(\Gamma)\|^2_R \\ x(t-i+1) - a(x(t-i)) \\ -B_u(x(t-i)) \left[\Gamma + \hat{\Delta}(t-1)\right] u(t-i) = s_i, \\ \qquad\qquad\qquad\qquad i = 1,\ldots,N \end{array} \quad (13)$$

with $Q_i >> R, i = 1,\ldots,N$.

Step 2: - Given $\hat{\Gamma}(t)$, compute $\hat{\Delta}(t) = \hat{\Delta}(t-1) + \hat{\Gamma}(t)$ and solve the following CAP

$$\begin{array}{l} u(t) \triangleq \arg\min_{s,u} \|s\|^2_{Q_s} + \|u\|^2_{R_u} \\ B_v(x(t))v(t) = B_u(x(t))\hat{\Delta}(t)u + s \\ u \in \Omega \end{array} \quad (14)$$

where $s \in \mathcal{R}^n$ is the parity slack vector and $Q_s = Q_s' > 0$ and $R_u = R_u' \geq 0$ consistent weighting matrices. In order to force slack vector to be as small as possible usually $Q_s >> R_u$ is chosen.

Remark 1 - In all cases in which a unique solution exists for problems (11), (12) and (13) an analytical expression can easily be determined. This would be beneficial for maintaining the on-line numerical burden of the algorithm low. However, in general one cannot ensure that a unique solution exists unless special care in generating the inputs $u(t)$ is taken. See e.g. subsection 3.2.

3.2 Properties of the two-steps algorithm

In this section we will investigate the properties of the proposed algorithm with a particular regard to constant actuator's faults or loss of effectiveness. To this end, the following fault at time t'

$$\begin{array}{ll} \Delta(t) = I & t < t', i = 1,\ldots,m \\ \Delta(t) = \Delta' & t \geq t', i = 1,\ldots,m \end{array} \quad (15)$$

is assumed where $\Delta' = diag\{\delta_1', \ldots \delta_m'\}$ is the a constant diagonal matrix corresponding to the true loss of effectiveness.

In particular, we are interested to study the asymptotical properties of the R-weighted estimation error

$$e_R(t) = ||\text{vect}(\hat{\Delta}(t) - \Delta')||_R \quad (16)$$

and the conditions of its convergence to zero. It is reasonable in fact to argue that, as many other parameters estimators, the convergence of the proposed one strongly depends on the nature of the input signals. Such a dependence, especially in a closed loop embedding, can yield to partially uncorrected estimations (Åström, Wittenmark, 1989). The following result on the monotonicity of the estimation error can be stated.

Proposition 1 - Given the overactuated physical plant (8) and the corresponding virtual plant (3)-(9), let the algorithm (11)-(12) perform under (15). Then, the weighted estimation error $e_R(t) = ||\text{vect}(\hat{\Delta}(t) - \Delta')||_R$ is a monotonically non-increasing sequence, i.e. $e_R(t+1) \leq e_R(t), \forall t > t' + N$.

Proof - See (Casavola, Garone, 2006).

Finally, by Proposition 1 and exploiting some arguments of its proof, under a constant fault it is possible to conclude that:

Main results

1 - As it was expected, in the general case the algorithm does not ensure that $e_R(t)$ converges to zero. In fact, the convergence strictly depends on the nature of the $u(t)$ history;

2 - Because $e(t)$ is monotonically non-increasing, if $\exists t^* > t' + N$ such that $e_R(t^*) = 0$ then $e_R(t) = 0$, $\forall t \geq t^*$;

3 - A sufficient condition for $e(t)$ to have zero value at some finite time $t^* > t' + N$ is that $rank\{(M(t^*)\} = m$, where

$$M(t) = \begin{pmatrix} B_u(x(t-1))diag\{u_1(t-1),\ldots,u_m(t-1)\} \\ \ldots \\ B_u(x(t-N))diag\{u_1(t-N),\ldots,u_m(t-N)\} \end{pmatrix} \quad (17)$$

Proof - See (Casavola, Garone, 2006).

3.3 A threshold two-steps algorithm for linear systems

It has been shown that the proposed two-steps algorithm to the $\mathcal{F}$**-TCAP** does not guarantee the convergence of the estimation error to zero in general because of possible rank deficiency of the $M(t)$ matrix. A popular way to move around this obstacle is by the introduction of artificial disturbances able to force the input signals to be persistently exciting the system. Those disturbances obviously cause unwanted side effects on the system behavior. Here we will perform a different policy and we will exploit both parameter estimator properties and actuator redundancy to reduce side effects.

This is made by using the following two key ideas:

1- Because of the monotonicity of the estimation error, in order to have an exact estimate of $\Delta(t)$ is enough that matrix $M(t)$ have full rank at least for a single time instant t^*
2- It is possible to exploit actuator redundancy in order to reduce the side effects of artificial disturbances.

In order to easily perform the objective of having a full rank $M(t)$, the following sufficient condition for linear systems is proved.

Proposition 2 - Let $B_u(x) = B_u$ be a constant matrix and $B_{u,j} \neq 0, j = 1, \dots, m$, denote the j-th column of B_u. Then $rank\{M(t)\} = m$ for $N = m$ provided that

$$rank\left\{\begin{pmatrix} u_1(t-1) & \dots & u_m(t-1) \\ \dots & \dots & \dots \\ u_1(t-m) & \dots & u_m(t-m) \end{pmatrix}\right\} = m, \qquad (18)$$

Proof It is enough to notice that, for N=m and $B_u(x(t)) = B_u$, one has that

$$M(t) = M_U(t)\ M_{B_u}, \quad \forall t \qquad (19)$$

where:

$$M_U(t) = \begin{pmatrix} u_1(t-1)I_{n\times n} & \dots & u_m(t-1)I_{n\times n} \\ \dots & \dots & \dots \\ u_1(t-m)I_{n\times n} & \dots & u_m(t-m)I_{n\times n} \end{pmatrix} \qquad (20)$$

is full rank whenever (18) holds true and

$$M_{B_u} = \begin{pmatrix} B_{u,1} & 0 & \dots & 0 \\ 0 & B_{u,2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & B_{u,m} \end{pmatrix} \qquad (21)$$

is full rank if $B_{u,j} \neq 0, j = 1, \dots, m$. □

3.4 An on-line algorithm

It is possible then to present the following algorithm in which condition (18) is ensured in (22) by applying an appropriate perturbation to the allocated inputs $u(t)$.

Init
  $Flag = 0; N = 0; r = 0;$
Step 1 - `Residual generator and threshold`
  `if(`$N > 0$`)`
    $r = x(t) - Ax(t-1) - B_u(\hat{\Delta}(t-1))u(t-1)$
  `if (`$r^T Q r > \epsilon_{thr}$`) AND (`$Flag == 0$`)`
    $Flag = 1; N = 0;$
  `if (`$Flag == 1$`) AND (`$N == m$`)`
    $Flag = 0;$
Step 2 - `Parameter Estimation`
  `Solve (13)`
Step 3 - `Allocation`
  `Solve (14)`
Step 4 - `Adding Artificial Disturbances`
  `if(`$Flag == 1$`)`
    $U_{old} = [u(t-1), \dots, u(t-N)]^T$
  `if(`$Ker(U_{old}) * u(t) == 0$`)`
    `Solve`

$$\begin{array}{l} \min\limits_{s,u} \|s\|^2_{Q_s} + \|u(t)\|^2_{R_u} \\ B_v(x)v = B_u(x)\hat{\Delta}(t)u(t) + s \\ u(t) = \sum\limits_{i=1}^{N} \beta_i u(t-i) + \sum\limits_{i=1}^{m-N} \alpha_i Ker_i\{U_{old}\}, \\ [\alpha_1, \dots, \alpha_{m-N}]^T \neq 0 \\ u \in \Omega \end{array} \qquad (22)$$

  `if(`$N < m$`)`
    $N = N + 1$
  `goto Step 1`

where $Q \in \mathcal{R}^{n\times n}$ is an appropriate weighting matrix and ϵ_{thr} is an appropriate scalar threshold.

4. NUMERICAL EXAMPLES

4.1 Linear unstable model

Consider the following linear model

$$x(t+1) = Ax(t) + B_u\Delta(t)u(t) \qquad (23)$$

where $x \in \mathcal{R}^3$ is the state vector and $u = [u_1, u_2, u_3]$ the physical input vector subject to the constraints $-5 < u_i < 5 \;\; i = 1, \dots, 3$. Matrices A and B_u are

$$A = 1.2, \quad B_u = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}, \qquad (24)$$

and $\Delta(t)$ is assumed as

$$\begin{array}{ll} \Delta(t) = diag\{1,1,1\} & t < 50 \\ \Delta(t) = diag\{1,1,0\} & 50 \leq t < 225 \\ \Delta(t) = diag\{1,0.5,0\} & t \geq 225. \end{array} \qquad (25)$$

consisting of a sequence of two faults. The first occurring at time $t = 50$, when the effectiveness of the third actuator becomes zero. This is followed, at time $t = 225$, by a 50% reduction of the effectiveness of the second actuator. The virtual input matrix $B_v = 1$ and the virtual control law $K = -0.6$ have been chosen. The threshold two-steps version of the ($\mathcal{F}$-TCAP) strategy was used with parameters: $\epsilon_{thr} = 10^{-5}$, $Q = Q_i = 10^5$, $i = 1, \ldots, 3$, and $R = R_u = I_{3\times 3}$. Simulation results on plant evolutions, actuators' allocation and control reconfiguration are reported in next figures for a tracking problem from an initial state $x(0) = 0$ and a square wave reference signal. In order to show the the effectiveness of the adaptive strategy, two sets of simulations have been accomplished: with and without the use of the adaptive **($\mathcal{F}$-TCAP)** strategy. When **($\mathcal{F}$-TCAP)** is not used, a CAP problem is solved at time $t = 0$ and the corresponding allocation rule frozen afterwards. Figs. 2-4 report respectively the output, the physical and virtual input closed-loop evolutions achieved with (UP) or without (BOTTOM) the use of the (**$\mathcal{F}$-TCAP**) strategy. In particular, in Fig. 2 it is easy to note the effectiveness of **$\mathcal{F}$-TCAP** in reconfiguring the actuators' allocation after a fault occurrence. Correspondingly, Figs. 3 and 4 report the physical and virtual input evolutions. It is worth noticing how signals related to failed actuators smartly change, coherently with the new estimated actuators effectiveness. On the contrary, under a frozen allocation, all input signals change uniformly and the tracking performance is lost. This behavior can be better explained in Fig. 4, where the virtual inputs v are shown in both cases. In the (UP) part is possible in fact to remark how, unlike in the (BOTTOM) part, the control law behavior is not influenced by the faulty events, apart around the time instants of fault occurrences, and the steady-state values of the control action remain unchanged. Finally, in Fig. 5 , (UP) the estimation error $e_R(t) = ||Vect(\hat{\Delta}(t) - \Delta(t))||_R$ and (BOTTOM) the difference between the desired total control effort and the actual one, i.e. $B_u\Delta(t)u(t) - B_v v(t)$ are reported, both achieved under the proposed **$\mathcal{F}$-TCAP** algorithm. Those are two important indexes to evaluate the estimation and reconfiguration performances, the lower the better (at zero one has exact estimation and allocation). Notice, in particular, their finite-time convergence to zero.

4.2 Networked Pipeline

Consider the pipeline network of Figure 6 providing water to a tank of section S. The water level x_1 has to be maintained to a suitable value in order to ensure that enough water exits from the outgoing pipe of section a on the bottom of the tank. The level regulation is performed by proportional linear valves with opening ratio $u_i \in [0, 1]$. Then, the i-th incoming flow to the tank is given by

$$\phi_i(t) = q_i(t)u_i(t), \tag{26}$$

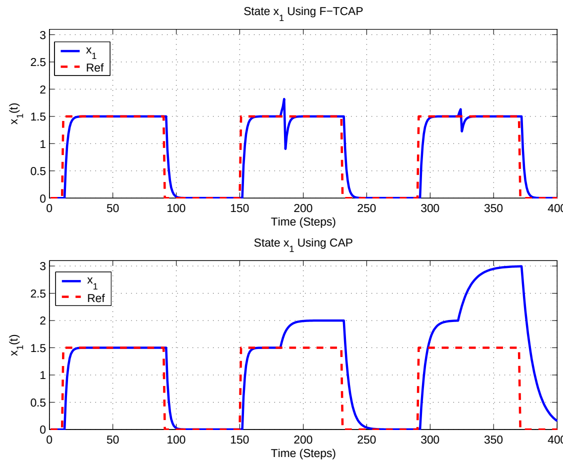

Fig. 2. Output and reference with (UP) and without (BOTTOM) **$\mathcal{F}$-TCAP**.

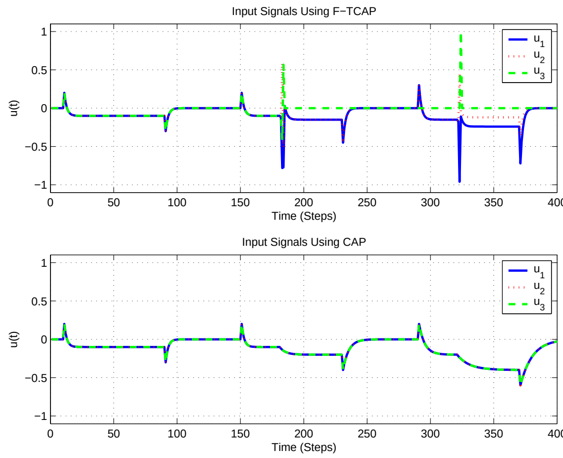

Fig. 3. Physical inputs $u(t)$ with (UP) and without (BOTTOM) **$\mathcal{F}$-TCAP**.

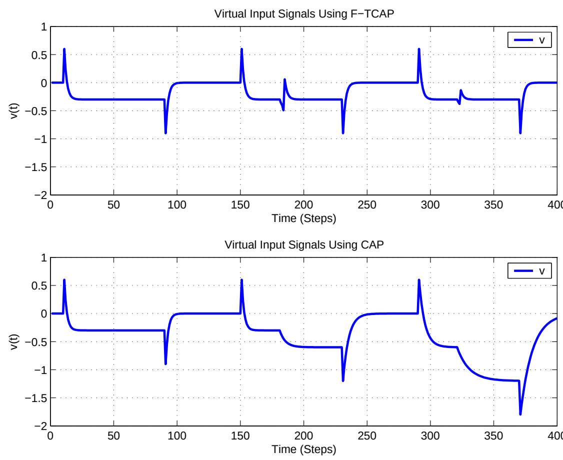

Fig. 4. Virtual inputs $v(t)$ with (UP) and without (BOTTOM) **$\mathcal{F}$-TCAP**.

where $q_i(t) \in [0, \bar{q}_i]$ is the incoming flows to each valve which is supposed to be bounded, unknown and time-varying. An alternative way to rewrite (26) is

$$\phi_i(t) = \bar{q}_i\delta_i(t)u_i(t) \tag{27}$$

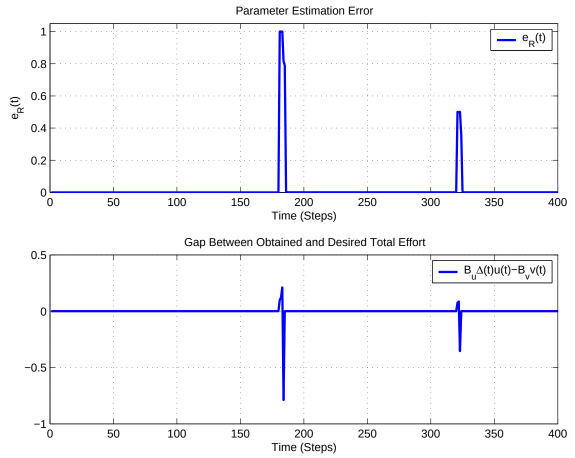

Fig. 5. (UP) Parameter estimation error $e_R(t)$) and (BOTTOM) parity equation, viz. the gap between obtained total effort $B_u\Delta(t)u(t)$ and desired total effort $B_v v(t)$

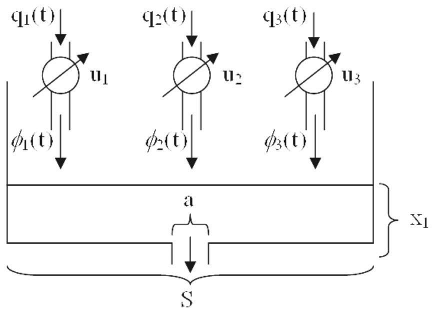

Fig. 6. Pipeline Network.

with $\delta_i(t) \in [0,1]$ denoting the fraction of the maximum flow $\bar{q}_i$ corresponding to the actual flow. Each $\delta_i(t)$ can be interpreted as the effectiveness of the i-th pipeline in carrying out its maximum flow. Therefore, by denoting with g and, respectively, ρ the gravity constant and the water density, the overall network model is given by

$$\rho S\dot{x}_1 = -\rho a\sqrt{2gx_1} + B_u\Delta(t)u, \qquad (28)$$

where $\Delta = diag\{\delta_1, \delta_2, \delta_3\}$ is the usual effectiveness matrix, $B_\alpha = [\bar{q}_1, \bar{q}_2, \bar{q}_3]$ is the input matrix and $u = [u_1, u_2, u_3]^T$ is the input vector.

In order to perform the tracking of the reference signal $r(t)$ shown in Fig. 7, the $[0-1]$-saturated state-feedback law

$$v(t) = \sigma(K(r(t) - x_1(t))) \qquad (29)$$

is used where $\sigma(\cdot)$ is the $[0-1]$ saturation function and $v(t) \in [0,1]$ the virtual input. It represents the opening ratio of a single virtual valve acting on an unique pipeline with $\bar{q}_1 + \bar{q}_2 + \bar{q}_3$ maximum incoming flow. Consequently, the input virtual matrix for this signal is set to $B_v = [\bar{q}_1 + \bar{q}_2 + \bar{q}_3]$. It is also assumed that the water provided by each pipeline has a different cost and one wants to minimize the total cost in maintaining the water in the tank at a prescribed level. In order to minimize the total cost, instead of (14), it is

Tank	Value
S_1	2500 cm^2
A_1	9 cm^2
Parameters	**Value**
g	980 cm/(sec^2)
ρ	0.001 Kg/(cm^3)
$\bar{q}_1 = \bar{q}_2 = \bar{q}_3$	2.5
T_c	0.1 sec
K	5

Table 1. Parameter values of the plant

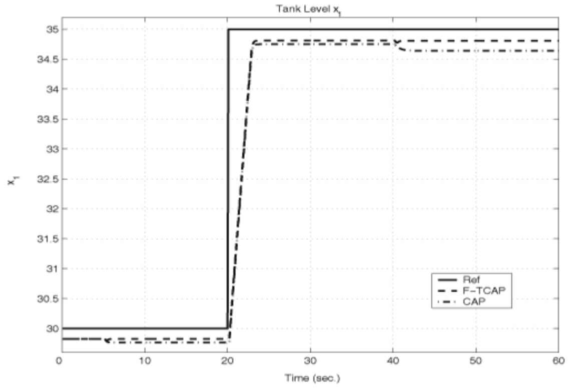

Fig. 7. Tank level $x_1(t)$ and reference with and without $\mathcal{F}$**-TCAP**.

convenient to consider the following optimization problem

$$\begin{aligned} &u(t) \triangleq \arg\min_{s,u} \|s\|^2_{Q_s} + R_u u \\ &B_v v(t) = B_u\hat{\Delta}(t)u + s, \\ &u \in \Omega, \end{aligned} \qquad (30)$$

where $R_u = [1, 10, 100]$, that is the water flow from pipes 1, 2 and 3 cost respectively 1, 10 and 100 and $Q_s = 10^5$.

The plant, the control law and the control allocator have been simulated with parameters shown in Table 1. The simulations have been carried out for an initial state $x_1(0) = 29.826$ and for the following $\Delta(t)$ sequence

$$\begin{aligned} \Delta(t) &= diag\{1,1,1\}, && t < 5, \\ \Delta(t) &= diag\{0.7,1,1\}, && 5 \le t \le 35, \\ \Delta(t) &= diag\{0.7,0.3,1\}, && t \ge 35. \end{aligned} \qquad (31)$$

In order to show the effectiveness of the proposed strategy, the experiment has been simulated both with the proposed $\mathcal{F}$**-TCAP** scheme and with a simple **CAP** allocation strategy. In Figure 7 the water level x_1 is reported. It is worth noting that $\mathcal{F}$**-TCAP** is able of reconfiguring the input signals after the fault occurrences without affecting the closed-loop tracking performances. On the contrary, a certain amount of tracking performance degradation is visible for **CAP**. Figures 8 and 9 report respectively the physical input $u(t)$ and the virtual input $v(t)$ for $\mathcal{F}$**-TCAP** (UP) and in the **CAP** (DOWN) case.

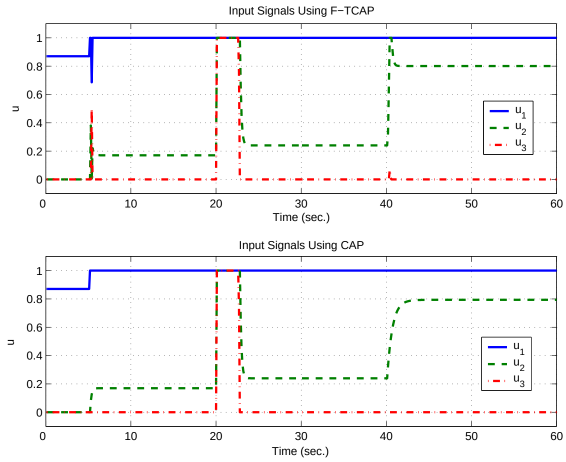

Fig. 8. Physical inputs $u(t)$ to the valves with (UP) and without (BOTTOM) $\mathcal{F}$**-TCAP**.

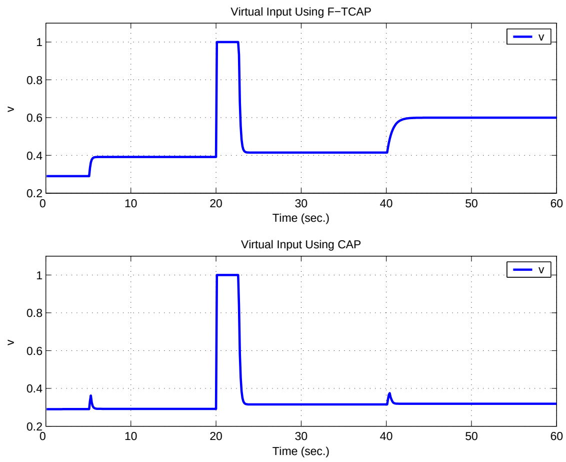

Fig. 9. Virtual inputs $v(t)$ to the pipeline network with (UP) and without (BOTTOM) $\mathcal{F}$-**TCAP**.

5. CONCLUSIONS

A preliminary adaptive scheme to perform fault tolerant control allocation for nonlinear discrete-time networked dynamical systems has been here proposed for disturbance-free systems subject to actuator faults or loss of effectiveness. Two algorithms have been proposed and their properties investigated. The effectiveness of the proposed method has been shown by means of two numerical experiments.

REFERENCES

H. Kwakernaak, R. Silvan (1972). Linear optimal control system. *Weley.*

W.C. Durham, Contrained Control Allocation, *Journal of Guidance, Control and Dynamics*, Vol. 16, No. 4, 1993 pp.717-725

K.A. Bordignon, W.C. Durham, Closed-Form Solutions to Contrained Control Allocation Problem, *Journal of Guidance, Control and Dynamics*, Vol. 18, No.5, 1995, pp.1000-1007.

J. Buffington, D. Enn, Lyapunov Stability Analysis of Daisy Chain Control Allocation, *Journal of Guidance, Control and Dynamics*, Vol. 19, No. 6, 1996 pp.1226-1230

W.C. Durham, Efficient, Near-Optimal Control Allocation, *Journal of Guidance, Control and Dynamics*, Vol. 22, No.2, 1998, pp.369-372.

J.D. Bošković, B. Ling, R.Prasanth, R.K. Mehra, Design of Control Allocation Algorithms for Overactuated Aircraft Under Constraints Using LMIs, *Proceedings of the 41st IEEE Conference on Decision and Control*, Las Vegas, 2002, pp. 1711-1716.

O. Härkegård, Efficien Active Set Algorithms for Solving Contrained Least Squares Problem in Aircaft Control Allocation,, *Proceedings of the 41st IEEE Conference on Decision and Control*, Las Vegas, 2002, pp. 1295-1300.

J.A.M. Petersen, M. Bodson, Interior-Point Algorithms for Control Allocation *Journal of Guidance, Control and Dynamics*, Vol. 28, No.3, 2005, pp. 471-480.

T.A. Johansen, T.I. Fossen, P.Tøndel, Efficient Optimal Contrained Control Allocation via Multiparametric Programming, *Journal of Guidance, Control and Dynamics*, Vol. 28 No.3, 2005, pp. 506-514.

M.Bodson, Evaluation of Optimization Methods for Control Allocation, *Journal of Guidance, Control and Dynamics*, Vol. 25, No. 4, 2002, pp. 703-711.

J.Jin, Modified Peudoinverse Redistribution Methods for Redundant Controls Allocation, *Journal of Guidance, Control and Dynamics*, Vol.28, No.5, 2005, pp. 1076-1079.

K.J. Åström, B. Wittenmark *Adaptive Control*, Addison-Wesley Publishing Company, 1989

M. Bodson, J.E. Groszkiewicz, Multivariable Adaptive Algorithms for Reconfigurable Flight Control, *IEEE Transactions on Control Systems Technology*, Vol. 5, No. 2, 1997, pp.217-229.

G.Tao, S. Chen, S.M. Joghi, An adaptive control scheme for sistems with unknown actuator failures, *Automatica*, Vol. 38, 2002, pp. 1027-1034.

J. Buffington, P. Chandler, M. Pachter, Integration of on-line System Identification and optimization-based Control Allocation. *AIAA Guidance, Navigation, and Control Conference* , Boston, 1998, AIAA-98-4487.

M.A. Bolender, D.B. Doman, Nonlinear Control Allocation Using Piecewise Linear Functions: A Linear Programming Approach, *Journal of Guidance, Control and Dynamics*, Vol. 28, No.3, 2005, pp.558-562.

A. Casavola, E. Garone, Adaptive Scheme for Actuator Allocation, *DEIS-University of Calabria, Technical report*, DEIS-18/06, 2006.