Enhancing the Actuator Fault Tolerance in Autonomous Overactuated Vehicles via Adaptive Control Allocation

Alessandro Casavola and Emanuele Garone

Abstract— **This paper presents an adaptive actuator allocation scheme that is fault-tolerant with respect to actuator faults and loss of effectiveness. The main idea is to use an ad-hoc online parameter estimator coupled with an allocation algorithm to perform on-line control reconfiguration whenever necessary. A preliminary algorithm is proposed for nonlinear discrete-time systems. Its main properties are summarized in the disturbance-free case and its effectiveness shown by means of two numerical examples, the second one dealing with driving/braking loss of effectiveness due to tire/surface contact force variations in 4WS4WD terrestrial vehicles.**

I. Introduction

Actuator redundancy is an important issue to deal with in increasing the fault-tolerant properties of many real plants. For example, it is a very common matter in (autonomous) vehicle applications due to safety reasons. A traditional way to handle overactuated systems (i.e. systems with physical actuator redundancy) is to resort to optimal control design methods [1]. Such an approach achieves both regulation and control distribution amongst the actuators at the same time. A different approach consists of using a simpler control law that specifies only the total control effort that has to be produced and 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 the last decade and several methods have been proposed: Daisy Chaining [2], Direct Control Allocation [3]-[4], Convex Optimization Based algorithms [5]-[11] and PseudoInverse-Redistribution (PIR) methods [12]-[13].

In this paper the presence of redundant actuators is exploited to develop effective fault-tolerant reconfigurable control strategies. To this end, the so-called Reconfigurable Control Allocation (RCA) problem [14]-[16] is revisited. The key idea is depicted in Fig. 1 where supposedly the control law has been 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 total control effort $v(t)$ on the physical actuators $u(t)$ based on some meaningful criterion. Then, in the case of actuator fault or loss of effectiveness, control reconfiguration is possible in many cases 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 ways to distribute the control amongst actuators, all of which equivalent in terms of closed-loop properties. Unlike other works on the topic, here the algorithm is not assumed to know in advance 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 rules by solving a standard constrained QP problem. Two final examples are presented to show the effectiveness of the proposed strategies. Other examples have been reported in [18].

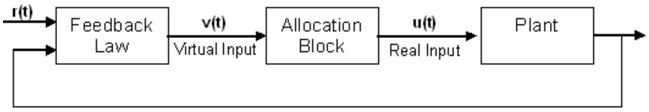

Fig. 1. Control structure with allocation and control performed separately

II. Problem statement

A. Control Allocation Problem

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

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

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: $Rank(B_u(x)) = k < m, \ \ \forall x$;
2) The input signal $u(t)$ lies into a compact set Ω, i.e.

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

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

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

$$\begin{aligned} x(t+1) &= a(x) + B_v(x)\,v(t), & (3)\\ B_v(x)\,v(t) &= B_u(x)\,u(t), & (4) \end{aligned}$$

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.

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

Alessandro Casavola, and Emanuele Garone are with the Dipartimento di Elettronica, Informatica e Sistemistica, Universitá degli Studi della Calabria, Via Pietro Bucci, Cubo 42-C, Rende (CS), 87036, ITALY

Hereafter, the system (3) will be referred to as the *virtual plant* while the equation (4) as the *parity equation* of the system, which defines the analytical relationships between the virtual and physical inputs. Note that in such a scheme, the virtual control input $v(t)$ represents the *desired total control effort* that 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 ([13]-[2]). Note that:

- Many previous works on the topic re-arrange the equation (4) as follows
$$\begin{array}{rcl} v(t) &=& B(x)\,u(t) \\ B_u(x) &=& B_v(x)\,B(x) \end{array} \tag{5}$$
where $B(x) \in \mathcal{R}^{k \times m}$ is a factorization of $B_u(x)$
$$B(x) = \left(B_v^T(x)B_v(x)\right)^{-1} B_v^T(x)B_u(x). \tag{6}$$
- **CAP** could not admit any solution due to the actuators saturation constraints (2). In such a case, **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 which are solutions for **CAP**. This fact can be exploited to comply with other specifications besides the **CAP** requirements.

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

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

where $Q_s = Q_s' > 0$ and $R_u = R_u' \geq 0$ are consistent weighting matrices. The slack-variable s is used to enlarge the set of solutions in the parity equation (4) and it allows the achievement of approximate allocations. When zero, a perfect allocation is achieved. On the contrary, the penalty on u is optional and it is used to minimize the actuator efforts 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 (see [11]-[13]). For the purposes of this paper, it is important to notice here that computational efficiency obtained through explicit approaches is paid in terms of a reduction of flexibility w.r.t. reconfiguration issues (see [11]).

B. 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 $\delta_i(t) \in \mathcal{R}, i = 1, ..., m$ are piecewise constant sequences representing the effectiveness of any single actuator. Notice that, in the absence of fault occurrences, $\Delta(t) = I$. Moreover, the parity equation (4) becomes

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

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.*

III. Two-step 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-step 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 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 parameter estimation and allocation problems. Many of the existing algorithms solving the two problems can be arranged in this general scheme.

A. A simple two-step algorithm

Hereafter, a very simple two-step 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 incremental matrix

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

defined as the diagonal matrix

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

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

We are especially interested in algorithms able to detect 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 $\hat{\Gamma}$. A workable strategy corresponds to the solution of the following weighted least-squares problem

$$\begin{array}{l} \hat{\Gamma}(t) \triangleq \arg\min\limits_{s_i,\Gamma} \sum\limits_{i=1}^{N} \|s_i\|^2_{Q_i} + \|\text{vect}(\Gamma)\|^2_R \\ x(t-i+1) - a(x(t-i)) \\ -B_u(x(t-i)) \left[\Gamma + \hat{\Delta}(t-1)\right] u(t-i) = s_i,\ i = 1, \dots, N \end{array} \tag{11}$$

with $Q_i >> R, i = 1, \dots, N$, where $\text{vect}(\Gamma) = [\gamma_1, \gamma_2, \dots, \gamma_m]^T \in \mathcal{R}^m$ and $s_i \in \mathcal{R}^n, i = 1, \dots, N$ are slack vectors and $R = R' > 0$ and $Q_i = Q_i' > 0, i = 1, \dots, N$ 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 $m/n \leq N \leq 2m$.

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

$$\begin{array}{l} u(t) \triangleq \arg\min\limits_{s,u} \|s\|^2_{Q_s} + \|u\|^2_{R_u} \\ B_v(x(t))v(t) = B_u(x(t))\hat{\Delta}(t)u + s \\ u \in \Omega \end{array} \tag{12}$$

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 - An analytical expression to approximately solve (11) for $Q_i >> R$ can be easily determined via pseudoinverse arguments. This would be beneficial for maintaining the on-line numerical burden of the algorithm low. See [17] for more details.

B. Properties of the two-step algorithm

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

$$\begin{array}{ll} \Delta(t) = I & t < t', i = 1, \dots, m \\ \Delta(t) = \Delta' & t \geq t', i = 1, \dots, m \end{array} \tag{13}$$

is assumed where $\Delta' = diag\{\delta_1', \dots \delta_m'\}$ is the 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

$$e_R(t) = ||\text{vect}(\hat{\Delta}(t) - \Delta')||_R \tag{14}$$

and the conditions for its convergence to zero. It is reasonable in fact to argue that, as many other parameter 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.

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) perform under (13). 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 [17]. □

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

$$M(t) = \begin{pmatrix} B_u(x(t-1))diag\{u_1(t-1), \dots, u_m(t-1)\} \\ \dots \\ B_u(x(t-N))diag\{u_1(t-N), \dots, u_m(t-N)\} \end{pmatrix} \tag{15}$$

Proof - See [17]. □

It is worth pointing out that the proposed two-step 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 persistently exciting the system. Those disturbances obviously cause unwanted side effects on the system behavior. In order to avoid them, more clever policies have been here implemented but, for space limitation, they are not discussed here.

IV. Numerical Examples

A. Linear unstable model

Consider the following linear model

$$x(t+1) = Ax(t) + B_u\Delta(t)u(t) \tag{16}$$

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 $-5 < u_i < 5 \;\; i = 1, ..., 3$. Matrices A and B_u are

$$A = 1.2, \quad B_u = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}, \tag{17}$$

and $\Delta(t)$ is assumed as

$$\begin{array}{lr} \Delta(t) = diag\{1, 1, 1\} & t < 50 \\ \Delta(t) = diag\{1, 1, 0\} & 50 \le t < 225 \\ \Delta(t) = diag\{1, 0.5, 0\} & t \ge 225. \end{array} \tag{18}$$

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. A version of the ($\mathcal{F}$-TCAP) strategy that ensures full rank for $M(t)$ was used [17] with parameters: $\epsilon_{thr} = 10^{-5}$, $Q = Q_i = 10^5, \;\; i = 1, .., 3$, 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 non-adaptive **CAP** problem is solved at each time instant without the updating of the effectiveness matrix.

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 allocation rules 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 the non-adaptive **CAP** 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 observe 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.

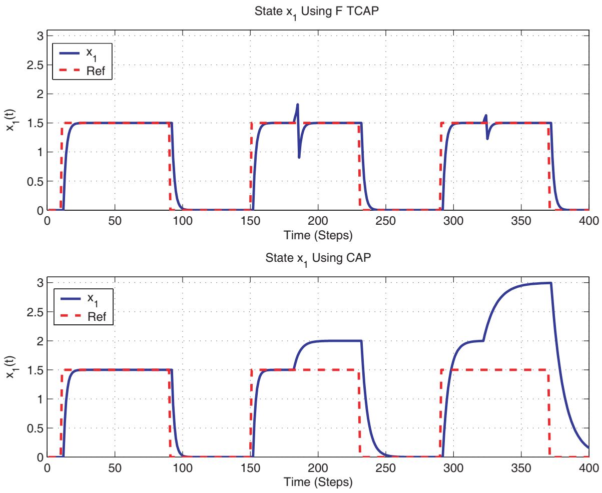

Fig. 2. Output and reference with (UP) and without (BOTTOM) **$\mathcal{F}$-TCAP**.

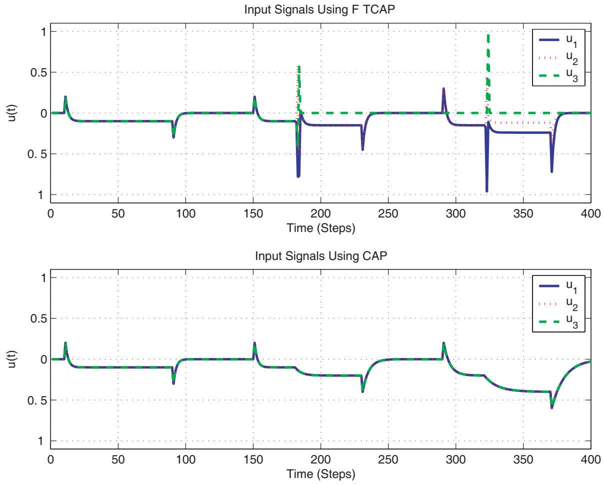

Fig. 3. Physical inputs $u(t)$ with (UP) and without (BOTTOM) **$\mathcal{F}$-TCAP**.

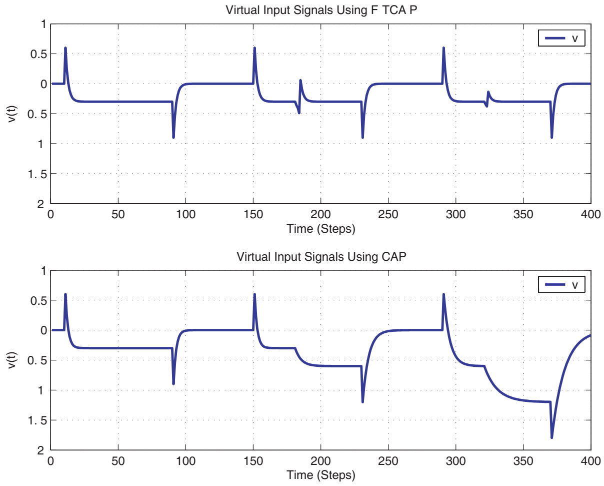

Fig. 4. Virtual inputs $v(t)$ with (UP) and without (BOTTOM) **$\mathcal{F}$-TCAP**.

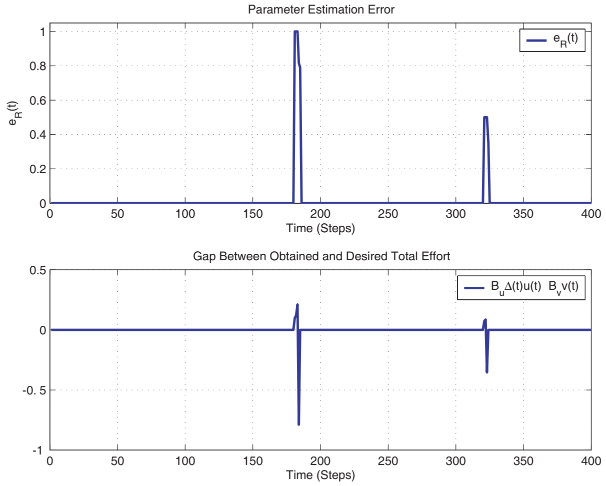

Fig. 5. (UP) Parameter estimation error $e_R(t))$ and (BOTTOM) parity equation, viz. the gap between obtained total effort $B_u\Delta(t)u(t)$ and desired total effort $B_vv(t)$

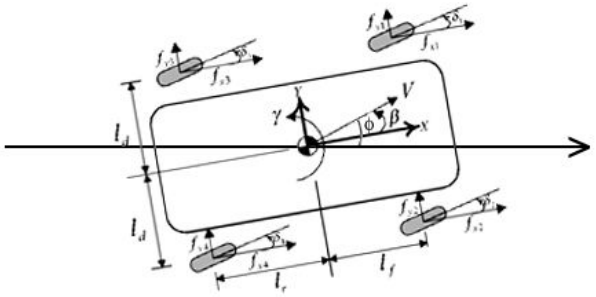

Fig. 6. A schematic of a 4WS4WD vehicle

B. Handling tire/surface loss of effectiveness in 4WS4WD cars

We consider the quasi-static model of a autonomous four-wheeled steering and four-wheeled driving (4WS4WD) vehicle presented in [19]. These vehicles are recognized to possess a highly integrated structure and actuator redundancy. They enjoy the advantage of 4WS vehicles in the superior performance of lateral dynamics and the benefits of the 4WD structure, allowing differential and driving wheel torques to be delivered at the same time for enhancing cornering performance and retaining cruising speed.

Each wheel ($i = 1, 2, 3, 4$) in Fig.6 can be independently steered by an angle δ_i and is provided by a driving/braking torque T_i by the propulsion system. Other important variables are the speed of the center of gravity (CG) V, its modulus $v = ||V||$, the sideslip angle β, the yaw angle ϕ and the yaw rate γ.

The goal is to distribute the total control effort, given in terms on the total x, y forces F_x and F_y and yaw momentum M at the CG, amongst the wheel driving torques and steering angles on the basis of estimated ground/wheel contact effectiveness. Under small sideslip angles $\beta \approx 0$, the following quasi-static dynamic model results

$$\begin{pmatrix} m & 0 & 0 \\ 0 & mv & 0 \\ 0 & 0 & J_Z \end{pmatrix} \frac{d}{dt} \begin{pmatrix} v \\ \beta \\ \gamma \end{pmatrix} = -\begin{pmatrix} 0 \\ mv\gamma \\ 0 \end{pmatrix} + \sum_{j=1}^{4} \begin{pmatrix} f_{xj} \\ f_{yj} \\ M_{zj} \end{pmatrix} \tag{19}$$

with

$$\sum_{j=1}^{4} M_{zj} = [-l_d \quad l_f]F_1 + [l_d \quad l_f]F_2 + [-l_d \quad -l_f]F_3 + [l_d \quad -l_f]F_4 \tag{20}$$

$$F_j = \begin{bmatrix} f_{xj} & f_{yj} \end{bmatrix}^T = \left[T_j/r_j \quad f_{zj}k_j \left(-\beta - \frac{l_f\gamma}{v} + \delta_j \right) \right], j = 1, 2 \tag{21}$$

$$F_j = \begin{bmatrix} f_{xj} & f_{yj} \end{bmatrix}^T = \left[T_j/r_j \quad f_{zj}k_j \left(-\beta + \frac{l_r\gamma}{v} + \delta_j \right) \right], j = 3, 4 \tag{22}$$

In (19)-(22), f_{xj} and f_{yj} are the wheel/surfance contact forces w.r.t. the X-Y body-fixed frame, T_j and δ_j the physical actuator signals, k_j the wheel/surface contact effectiveness (time-varying) and r_j and f_{zj} respectively the j-th effective wheel radius and the car weight acting on that wheel. The above quasi-static models will be used by the adaptive allocation algorithm to estimate the parameters k_i and to distribute the three virtual commands F_x, F_y and M into the physical eight signals T_i and δ_i.

The following loss of effectiveness scenario is considered for the effectiveness parameter vector $K(t) = [k_1, k_2, k_3, k_4]^T$:

$[5.5 \; 5.5 \; 5.5 \; 5.5]$ $t < 0.6$ s, dry asphalt on all wheels
$[4.5 \; 5.5 \; 4.5 \; 5.5]$ 0.6 $s \leq t < 1.5$ s, dry asphalt on one side and puddles on the other side
$[3.5 \; 3.5 \; 3.5 \; 3.5]$ $t \geq 1.5$ s, wet asphalt on all wheels

In Fig. 7, the continuous line depicts the prescribed trajectory whereas the dash-dotted and dashed lines represent the achieved trajectory respectively under the proposed adaptive $\mathcal{F}$-**TCAP** and the non-adaptive **CAP** strategies. Similar results are achieved for the other signals of interest depicted in next Fig. 8. It is worth commenting that no feedback loop is closed in these simulations to better show the ability of the allocation algorithm to reconfigure the control allocation rules. Nevertheless, in all above figures the reference curves and the ones achieved under $\mathcal{F}$-**TCAP** strategy are fully superimposed whereas large deviations occur under **CAP**. Finally, in Fig. 9 the estimated effectiveness k_i for the considered scenario are depicted. Also in this example,

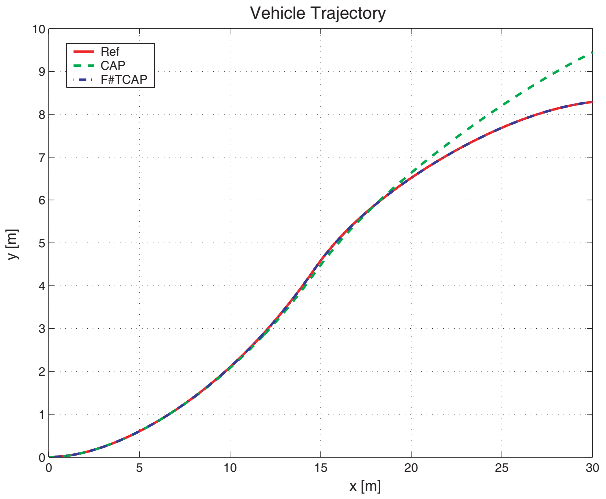

Fig. 7. (Trajectory of the car in the $x - y$ plane.

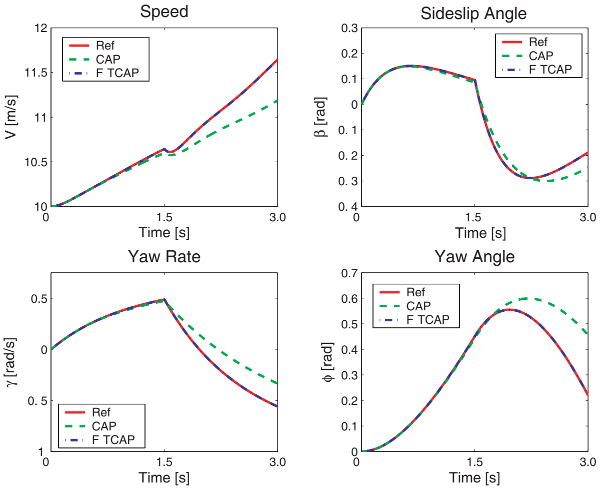

Fig. 8. Speed v, Sideslip angle β, Yaw angle ϕ and Yaw rate γ under the two allocation strategies.

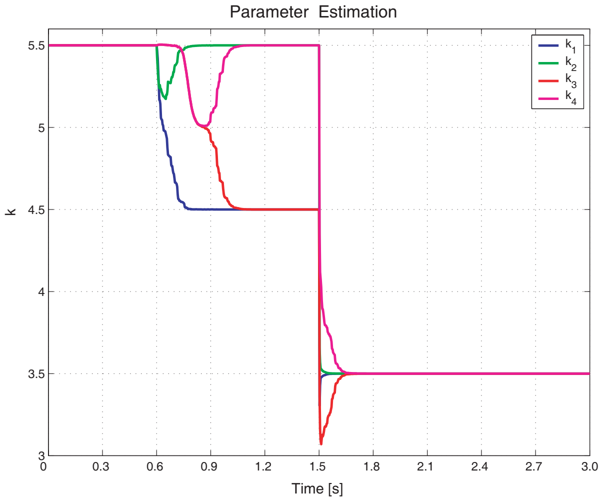

Fig. 9. Estimation of the weel/surface contact frictions k_i.

the estimates are quite accurate and a fast convergence is achieved.

V. Conclusions

A preliminary adaptive scheme to perform fault tolerant control allocation for nonlinear discrete-time system has been here proposed for disturbance free plants subject to loss of effectiveness. A workable algorithm has been proposed and its properties have been investigated. The effectiveness of the proposed method has been shown by means of two numerical examples.

References

[1] H. Kwakernaak, R. Silvan *Linear optimal control system*, Weley, 1972

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

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

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

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

[6] J.H. Oh, M.B. Jamoom, M.W. McConley, E.Feron, “Solving control allocation problems using semidefinite programming”, *Journal of Guidance, Control and Dynamics*, Vol 22, No.3, pp. 494-497, 1999.

[7] 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*, pp. 1711-1716, Las Vegas, USA, 2002.

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

[9] K.P. Lindegaard, T.I. Fossen, “Optimal Contrained Control Allocation in Marine Surface Vessels with Rudder”, *IFAC Conf. Menoeuvring and Control of Maine Craft*, Girona, Spain, 2003.

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

[11] 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, pp. 506-514, 2005.

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

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

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

[15] J.B. Davidson, F.J. Lallman, W.T. Bundick, “Real-Time Adaptive Control Allocation applied to a High Performance Aircraft”, *5th SIAM Conference on Control and Its Applications*, 2001.

[16] 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.

[17] A. Casavola, E. Garone, “Adaptive Schemes for Actuator Allocation”, *DEIS-University of Calabria, Technical Report*, DEIS-18/06, 2006.

[18] A. Casavola and E. Garone, “Adaptive fault toleant actuator allocation for overactuated plants”, *IEEE ACC 2007*, New-York, USA, July 11-13, 2007.

[19] S-T Peng,“On One Approach to Constraining the Combined Wheel Slip in the Autonomous Control of a 4WS4WD Vehicle”, *IEEE Transactions on Control Systems Technology*, Vol. 15, N. 1, pp. 168-175, Jan. 2007.