PAPER

Reconfiguration-Based Fault Tolerant Control of Dynamical Systems: A Control Reallocation Approach

Ali MORADI AMANI†a), *Student Member*, Ahmad AFSHAR†b), *Nonmember*, *and* Mohammad Bagher MENHAJ†c), *Member*

SUMMARY In this paper, the problem of control reconfiguration in the presence of actuator failure preserving the nominal controller is addressed. In the actuator failure condition, the processing algorithm of the control signal should be adapted in order to re-achieve the desired performance of the control loop. To do so, the so-called reconfiguration block, is inserted into the control loop to reallocate nominal control signals among the remaining healthy actuators. This block can be either a constant mapping or a dynamical system. In both cases, it should be designed so that the states or output of the system are fully recovered. All these situations are completely analysed in this paper using a novel structural approach leading to some theorems which are supported in each section by appropriate simulations.

key words: *fault tolerant control, control reconfiguration, actuator failure, system structure, control reallocation*

1. Introduction

Stability and performance of control systems are highly affected by system faults. The main goal of Fault Tolerant Control (FTC) methods is to overcome these undesired effects when a fault occurs in sensors, actuators or the system internally. There are two main approaches in fault tolerant control systems [1]: Passive Fault Tolerant Control (PFTC) and Active Fault Tolerant Control (AFTC). In PFTC, the controller should be designed such that the performance of the control system is acceptable in both normal and fault conditions. This should be achieved without any online modification to the controller. Here, some major faults are modelled and the controller is designed to compensate for their consequences.

In AFTC, the controller is designed to provide online self-adaptation under possible fault conditions. Here, initially a Fault Detection and Diagnosis (FDD) block should detect, isolate and possibly estimate a model for the fault. Then, the controller adapts itself to the fault condition. This approach is further subdivided to fault accommodation and control reconfiguration [1]. In the first one, controller parameters are re-tuned in fault conditions to recover the performance of the control system. This approach can basically be placed in the adaptive control scheme; however, its performance might not be acceptable when there is a severe fault (e.g. actuator failure) in the control loop. Here, the control system should be reconfigured based on the remaining set of sensors and actuators. This approach is called control reconfiguration which is not an easy task (if not impossible), especially in the presence of real-time constraints.

For control reconfiguration in the case of actuator failure, which is the main topic of this article, the “virtual actuator” technique is proposed in [1], [3] and [7]. In this technique, a static or dynamic block is inserted inside the control loop between the controller and the faulty plant. To preserve the nominal controller in the fault condition, this block should present the fault hiding effect from the controller point of view, i.e. the faulty plant in combination with the virtual actuator should present the same I/O behaviour as the fault-free plant. Using this assumption as well as stability conditions of the reconfigured control loop, the sufficient condition for the dynamic virtual actuator is derived and design conditions are fully discussed.

In addition to classic control methods (like robust control techniques), structural and energy-based approaches are also considered by the FTC researchers. A general structural approach based on the graph theory is presented in [1]. In [8] and [9], fault effects on the controllability Gramian is considered as a sense of the energy necessary for recovery of the faulty plant. [10] also focuses on reducing the energy spent for this performance recovery using reconfiguration of the reference input. One of the most recent activities in this field is reported in [11] in which, considering the case of actuator outages, a general framework for several controllers is designed. The recoverability of a given fault is defined by two structural properties, stability and possibility to re-achieve the performance, of the faulty system considering an acceptable pre-defined performance degradation level.

In this paper, pursuing the previous activities on structural and energy-based FTC, the reconfiguration problem is solved using a more conceptual approach which is based on the following principle:

When an actuator fails, its role should be reallocated among the others.

Based on this principle, a new approach for designing the reconfiguration block is proposed. This approach is a more conceptual, straightforward and yet easily understandable version of what was presented as virtual actuator in [1] and [3].

The paper is organized as follows: Sect. 2 presents some preliminaries necessary for the the rest of the paper.

Manuscript received September 20, 2011.
Manuscript revised November 21, 2011.

†The authors are with the Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran.
a) E-mail: a.m.amani@aut.ac.ir
b) E-mail: aafshar@aut.ac.ir (Corresponding author)
c) E-mail: tmenhaj@ieee.org

DOI: 10.1587/transinf.E95.D.1074

Static reconfiguration block is designed in Sect. 3 for both state and output recovery based on the control reallocation approach. The dynamic reconfiguration block and its design conditions are presented in Sect. 4. Also, the performance of the reconfiguration process in the presence of both static and dynamic reconfiguration blocks are compared in this section. In Sect. 5, the problem is solved in the presence of constraints on input signals. Finally, Sect. 6 concludes the paper.

2. Preliminaries

Consider the following dynamical equations for a Multi Input Single Output (MISO) LTI system:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

in which, $x \in R^n$ is the state vector, $u \in R^m$ is the input vector and $y \in R$ represents system output. Assume that the system is successfully controlled by the following controller G_c as presented in Fig. 1 A:

$$\begin{aligned} \dot{x}_c(t) &= A_c x_c(t) + B_c u_c(t) \\ y_c(t) &= C_c x_c(t) \end{aligned} \tag{2}$$

Also, assume that Λ represents the set of eigenvalues of the system matrix $A_{n\times n}$ and $e_1, e_2, \ldots, e_n$ is an orthonormal basis for R^n, in which the i^{th} element of the vector e_i is "1" and the others are zero. The system states can be represented as [Appendix I]:

$$x(t) = \sum_{i=1}^{n}\sum_{k=1}^{m} \zeta_{ki} e_i, \quad \zeta_{ki} = b_{ki} \int_0^t e^{\Lambda(t-q)} u_k(q) dq \tag{3}$$

where ζ_{ki} shows the effect of k^{th} input (u_k) on the system states in the direction of e_i. The coefficients b_{ki} are derived form the projection of k^{th} columns of the matrix B on the space spanned by the e_i's [see (A· 3)]. Also, the output vector can be represented in this way as:

$$y(t) = \sum_{i=1}^{n}\sum_{k=1}^{m} \zeta_{ki} c_i \tag{4}$$

where c_i is derived form the projection of the vector C on the space spanned by the e_i^T's [see (A· 8)].

A method for presenting some actuator faults is to present them as a deviation in the corresponding columns of the matrix B. By this method, actuator failure can be shown as a zero column in B. Therefore, the faulty plant G_p^f affected by actuator failure can be represented as:

$$\begin{aligned} \dot{x}^f(t) &= Ax^f(t) + B^f u^f(t) \\ y^f(t) &= Cx^f(t) \end{aligned} \tag{5}$$

In the fault condition, the state and output of this faulty system are transferred to:

$$\begin{aligned} x^f(t) &= \sum_{i=1}^{n}\sum_{k=1}^{m} \zeta_{ki}^f e_i \\ y^f(t) &= \sum_{i=1}^{n}\sum_{k=1}^{m} \zeta_{ki}^f c_i \end{aligned} \tag{6}$$

in which,

$$\zeta_{ki}^f = b_{ki}^f \int_0^t e^{\Lambda(t-q)} u_k^f(q) dq \tag{7}$$

where u_k^f represents the k^{th} control signal of the faulty system. Now, the main duty of FTC algorithm is to re-transfer the output or state of the faulty system to the nominal condition. In this paper, it has been done by inserting a reconfiguration block G_R inside the control loop (Fig. 1 B). Assume G_R as follows:

$$\begin{aligned} \dot{x}_R(t) &= A_R x_R(t) + B_R u_R(t) \\ y_R(t) &= C x_R(t) + C x^f(t) \\ u^f(t) &= C_R x_R(t) + D_R u_R(t) \end{aligned} \tag{8}$$

in which, x_R is the state of the reconfiguration block. A_R, B_R, C_R and D_R should be designed to achieve a successful reconfiguration. In Sects. 3 and 4, this block will be designed to achieve state recovery (i.e. $x^f \to x$ as $t \to \infty$) as well as output recovery (i.e. $y^f \to y$ as $t \to \infty$).

The following theorem is useful in analysing the stability of the reconfigured control loop (Fig. 1 B):

Theorem 1: Assume that in Fig. 1 A, the system (1) is successfully controlled by the controller (2), i.e. stability and the desired performance are both achieved. An actuator failure converts the plant G_p to G_p^f presented in (5). If a reconfiguration block G_R (Fig. 1 B) in the form of (8) exists which can preserve the input/output behaviour of the faulty plant (i.e. $y_R/u_R = y_p/u_p$ in Fig. 1) then the resulted control loop still remains stable in the fault condition if the following matrix is Hurwitz:

$$\begin{bmatrix} A_R & 0 & B_R C_c \\ A_R + (B^f C_R - A) & A & (B_R + B^f D_R) C_c \\ 0 & B_c C & A_c \end{bmatrix} \tag{9}$$

Proof: See appendix II.

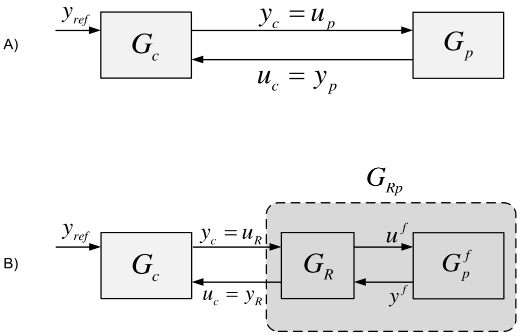

Fig. 1 FTC using reconfiguration block.

3. Static Reconfiguration Block Design

The FTC architecture considered in this paper is presented in Fig. 2 in which, a reconfiguration block is inserted into the faulty control loop while the controller of the fault-free condition is preserved. The simplest case is to consider the reconfiguration block as a constant matrix, i.e. $u^f = Nu$. N should be designed to recover state/output of the faulty system to the fault-free condition. This case is called reconfiguration using static reconfiguration block in this paper.

3.1 State Recovery

To recover the states of the faulty plant, the conditions under which the faulty state x^f in (6) is re-transferred to x in (3) should be obtained.

$$\begin{aligned} & x^f(t) = x(t) \\ & \rightarrow \sum_{i=1}^{n}\sum_{k=1}^{m} \zeta^f_{ki} e_i = \sum_{i=1}^{n}\sum_{k=1}^{m} \zeta_{ki} e_i \\ & \rightarrow \sum_{i=1}^{n}\left(\sum_{k=1}^{m} \zeta^f_{ki} - \sum_{k=1}^{m} \zeta_{ki}\right) e_i = 0 \end{aligned} \tag{10}$$

The vectors e_i are orthogonal, hence:

$$\sum_{k=1}^{m} \zeta^f_{ki} - \sum_{k=1}^{m} \zeta_{ki} = 0, \qquad \forall i \tag{11}$$

Regarding (3), the above equation can be rewritten as:

$$\sum_{k=1}^{m} b^f_{ki} u^f_k - \sum_{k=1}^{m} b_{ki} u_k = 0, \qquad \forall i \tag{12}$$

Considering $u^f = Nu$, (12) can be presented as the following matrix equation:

$$\underbrace{\begin{bmatrix} b^f_{11} & b^f_{21} & \cdots & b^f_{m1} \\ b^f_{12} & b^f_{22} & \cdots & b^f_{m2} \\ \vdots & \vdots & \vdots & \vdots \\ b^f_{1n} & b^f_{2n} & \cdots & b^f_{mn} \end{bmatrix}}_{B^f} N = \underbrace{\begin{bmatrix} b_{11} & b_{21} & \cdots & b_{m1} \\ b_{12} & b_{22} & \cdots & b_{m2} \\ \vdots & \vdots & \vdots & \vdots \\ b_{1n} & b_{2n} & \cdots & b_{mn} \end{bmatrix}}_{B} \tag{13}$$

Therefore, the static reconfiguration block N which satisfies (13) can successfully recover the states of the faulty plant. The stability of reconfigured control loop using the static reconfiguration block can be easily proved using theorem 1.

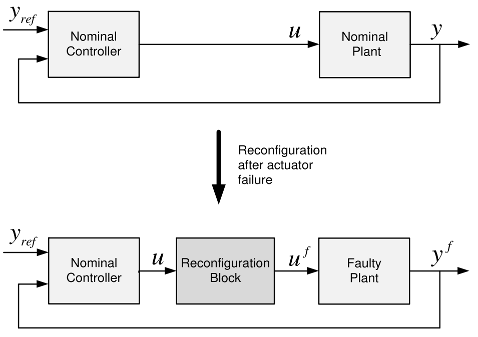

Fig. 2 FTC using static reconfiguration block.

3.2 Output Recovery

Considering the goal $y^f \rightarrow y$ and using some algebraic manipulations like what was led to (13), the following matrix equality will be driven as a condition for output recovery using static reconfiguration block:

$$\underbrace{\begin{bmatrix} b^f_{11}c_1 & b^f_{21}c_1 & \cdots & b^f_{m1}c_1 \\ b^f_{12}c_2 & b^f_{22}c_2 & \cdots & b^f_{m2}c_2 \\ \vdots & \vdots & \vdots & \vdots \\ b^f_{1n}c_n & b^f_{2n}c_n & \cdots & b^f_{mn}c_n \end{bmatrix}}_{B^f_c} N = \underbrace{\begin{bmatrix} b_{11}c_1 & b_{21}c_1 & \cdots & b_{m1}c_1 \\ b_{12}c_2 & b_{22}c_2 & \cdots & b_{m2}c_2 \\ \vdots & \vdots & \vdots & \vdots \\ b_{1n}c_n & b_{2n}c_n & \cdots & b_{mn}c_n \end{bmatrix}}_{B_c} \tag{14}$$

in which, c_i is defined in (A· 8). The following theorem concludes the achievements about reconfiguration using static reconfiguration block.

Theorem 2 (Control reconfiguration using static reconfiguration Block): Assume that in Fig. 1 A, the system (1) is successfully controlled by the controller (2), i.e. stability and the desired performance are both achieved. An actuator failure degrades the performance of the control loop. The static reconfiguration block ($u^f = Nu$ in Fig. 2) can recover
i) the state of the faulty system if $B^f N = B$
ii) the output of the faulty system if $B^f_c N = B_c$

For further analysis, it is clear from (13) and (14) that increasing in the number of failed actuators will cause more equations than free parameters which will make the problem unsolvable. The solution for (14) is more expectable since some of the coefficients c_i may be zero which will reduces the number of equations. The following lemma presents sufficient condition for the existence of static reconfiguration block.

Lemma1 (existence of the static reconfiguration block): The system (1) in the case of actuator fault can be:
a) Recovered respect to its state, only if $B \in im(B^f)$
b) Recovered respect to its output, only if $B_c \in im(B^f_c)$
where "$im(.)$" represents the image of the matrix. On the other hand, an extreme point can be considered which is very important in reconfigurability of the control system. Assume that in the decomposition of columns of the input matrix B (A· 6), there is an actuator u_k for which $\exists l; b_{kl} \neq 0$, noting that equivalent coefficients in other actuators are all zero, i.e. $b_{nl} = 0, \forall n \neq k$. This means that among all actuators, just u_k can move the system state in the direction e_l. Therefore, if it fails, no other input can play its role in this direction, i.e. the system states can not be recovered to the fault-free case. Hence, the category of critical actuators can be defined as follows.

Definition 1 (Critical Actuator): The actuator u_k is called "critical" in the sense of state recovery if there is a direction e_l for which:

$$b_{kl} \neq 0 \quad but \quad b_{nl} = 0, \ \forall n \neq k \tag{15}$$

Now, using this definition, the following lemma is easily established.

Lemma 2: The system (1) in the case of actuator failure is

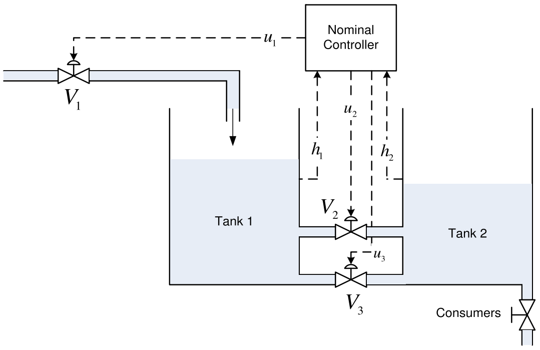

Fig. 3 Two coupled tanks problem [1].

unrecoverable respect to its state if one of the failed actuators is critical.

Example 1: To illustrate the reconfiguration by this technique, a two coupled tank problem is considered as depicted in Fig. 3. The main goal of the system is to control the water level of tank 2 for some consumers. The system is generally described by a nonlinear state space model which can be linearized around the operation point to the following model [1]:

$$\dot{x} = \begin{bmatrix} -0.0478 & -0.0004 & 0 \\ 1 & 0 & 0 \\ 0.0058 & 0 & -0.0058 \end{bmatrix} x + \begin{bmatrix} 0.0406 & -0.0058 & -0.0092 \\ -1 & 0 & 0 \\ 0 & 0.0046 & 0.0073 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

$$y = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x$$

According to (A· 3), the columns of the input matrix can be represented as:

$$b_1 = \begin{bmatrix} 0.0406 \\ -1 \\ 0 \end{bmatrix} = 0.0406 e_1 - e_2,$$

$$b_2 = \begin{bmatrix} -0.0058 \\ 0 \\ 0.0046 \end{bmatrix} = -0.0058 e_1 + 0.0046 e_3,$$

$$b_3 = \begin{bmatrix} -0.0092 \\ 0 \\ 0.0073 \end{bmatrix} = -0.0092 e_1 + 0.0073 e_3$$

First of all, it is clear that the actuator V_1 is a critical one since it is the only actuator which can force the system in the direction of e_2. Therefore, the system is unrecoverable respect to its states if V_1 fails.

Now assume that the actuator V_2 totally fails, i.e. it is completely closed. In this case, dynamical equations of the faulty system can be achieved by making all the entries of the second column (the column related to V_2) of the input matrix zero as:

$$\dot{x} = \begin{bmatrix} -0.0478 & -0.0004 & 0 \\ 1 & 0 & 0 \\ 0.0058 & 0 & -0.0058 \end{bmatrix} x + \begin{bmatrix} 0.0406 & 0 & -0.0092 \\ -1 & 0 & 0 \\ 0 & 0 & 0.0073 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

$$y = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x$$

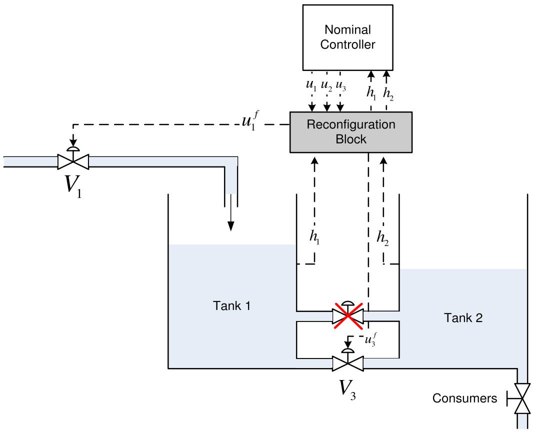

Fig. 4 Reconfiguration structure for two coupled tanks.

The control reconfiguration is proposed as depicted in Fig. 4. The control specification in this situation is to preserve stability as well as setpoint tracking for level of tank 2. According to (13),

$$\begin{bmatrix} 0.0406 & 0 & -0.0092 \\ -1 & 0 & 0 \\ 0 & 0 & 0.0073 \end{bmatrix} N = \begin{bmatrix} 0.0406 & -0.0058 & -0.0092 \\ -1 & 0 & 0 \\ 0 & 0.0046 & 0.0073 \end{bmatrix}$$

which has clearly no solution. Therefore, the complete state recovery in the case of failure in V_2 is impossible. To check the possibility of output recovery, the vector C can be decomposed to:

$$C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} = e_3^T$$

Hence, (14) can be written as:

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0.0073 \end{bmatrix} N = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0.0046 & 0.0073 \end{bmatrix}$$

$$\rightarrow N = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 46/73 & 1 \end{bmatrix}$$

Using this value for static reconfiguration block N, the output of the system will be recovered in the case of failure in V_2. Figures 5 and 6 show the performance of this reconfiguration block in the output and states recovery, respectively. It is assumed that the actuator V_2 is failed at T=500 sec. The static reconfiguration block is activated as soon as this failure is detected at T=1200 sec. As depicted in Fig. 5, the output of the system deviates from its desired value as the failure occurs at T=500 sec. However, it is recovered when the reconfiguration process starts at T=1200 sec. In spite of successful output recovery, Fig. 6 shows that the reconfiguration process is failed to recover the states of the faulty system. This means that a output recovery is successfully

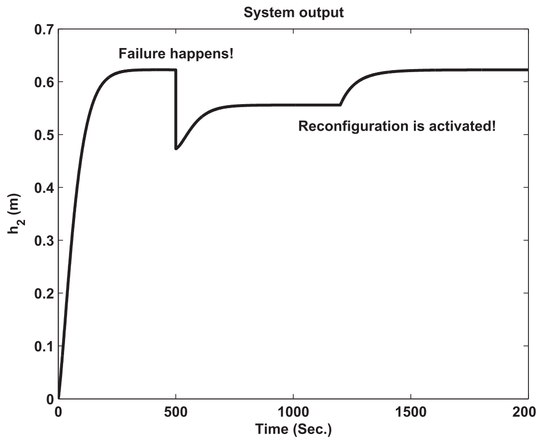

Fig. 5 The system output is recovered using a static reconfiguration block.

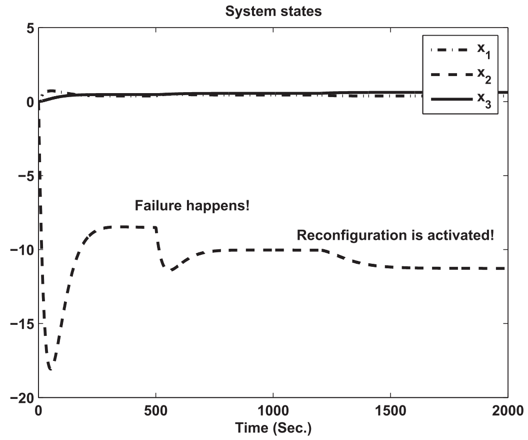

Fig. 6 The system states are not recovered using a static reconfiguration block.

achieved inserting just a constant mapping into the control loop.

4. Dynamical Reconfiguration Block

We start this section with an illustrative example. Again consider the system of example 1 whose output was finally reconfigured using a static reconfiguration block. Figure 7 shows the control signal for that example. It is seen that the control signal changes suddenly when the reconfiguration block is activated at T=1200 sec. Of course, this shock to the actuators is not acceptable in a practical control system. In order to avoid it in the time of activation of the reconfiguration block, a dynamic reconfiguration block is proposed in this section. The following dynamical reconfiguration block can be proposed for this problem:

$$\begin{aligned} \dot{x}_R(t) &= A_R x_R(t) + B_R u_R(t), \\ u^f(t) &= C_R x_R(t) + D_R u_R(t) \end{aligned} \tag{16}$$

in which,

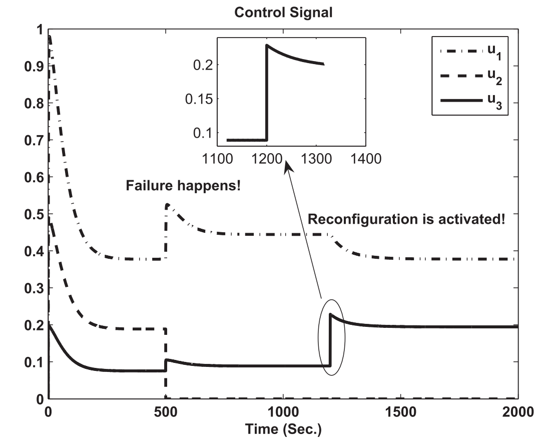

Fig. 7 Variation of control signal when static reconfiguration block activates.

$$\begin{aligned} A_R &= A - B^f M \\ B_R &= B - B^f \\ C_R &= M \\ D_R &= I \end{aligned} \tag{17}$$

and M is a design parameter. The question of why this kind of reconfiguration block is considered can be easily answered by looking at the stability condition of the faulty reconfigured control loop presented in theorem 1. By this selection of A_R and B_R, the set of eigenvalues of (9) can be decomposed to:

$$\sigma(A_R) \cup \sigma\left(\begin{bmatrix} A & BC_c \\ B_cC & A_c \end{bmatrix}\right) \tag{18}$$

The second part of this decomposition is stable as the fault-free control system has been assumed to be stable. About the first part, if (A, B^f) is stabilizable, M can be found to make A_R stable. Therefore, in this condition, the reconfigured control system in presence of dynamical reconfiguration block (16) remains stable. On the other hand, D_R is selected as a unity matrix to avoid a sudden shock in control signals when the reconfiguration block activates. Another important criteria to design M is the I/O behaviour of the dynamic reconfiguration block in the steady state condition. Comparing dynamic reconfiguration block with its static counterpart in the architecture of Fig. 2, it can be easily seen that the dynamic reconfiguration block can recover the state (output) of the faulty system if its I/O behaviour in the steady state condition is the same as N in (13) ((14)). It means that:

$$-M(A - B^f M)^{-1}(B - B^f) + I = N \tag{19}$$

Now, we return to the two tank problem which was previously presented in example 1. To design a dynamic reconfiguration block for it, M should be selected to make $A - B^f M$ stable and satisfy (19) with N achieved in example 1 for output recovery as well.The matrix M can be appropriately chosen as:

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -92 & 100 \end{bmatrix} \tag{20}$$

Figure 8 shows that the system output is successfully recovered like the the case of reconfiguration using static reconfiguration block in Fig. 5. The main advantage of the dynamic reconfiguration block is clarified comparing Fig. 9 with Fig. 7. The sudden change in the control signal of Fig. 7 is softened in Fig. 9. Therefore, using the dynamic reconfiguration block, the output recovery is achieved without any shock to the actuators. Of course, this achievement is a result of the extra degree of freedom, M, we have in the case of dynamic reconfiguration block.

Recent achievements are concluded in the following lemma:

Lemma 3: Assume that a system with dynamical equations (1) is successfully controlled, i.e. the stability and desired performance are both achieved. An actuator failure converts the plant G_p to G_p^f presented in (5). If the set (A, B^f) is still stabilizable, the following dynamic reconfiguration block in the architecture of Fig. 2 can recover the state/output of the faulty plant:

$$\begin{aligned}
\dot{x}_R(t) &= A_R x_R(t) + B_R u_R(t)\\
u^f(t) &= C_R x_R(t) + D_R u_R(t)\\
A_R &= A - B^f M\\
B_R &= B - B^f\\
C_R &= M\\
D_R &= I
\end{aligned} \tag{21}$$

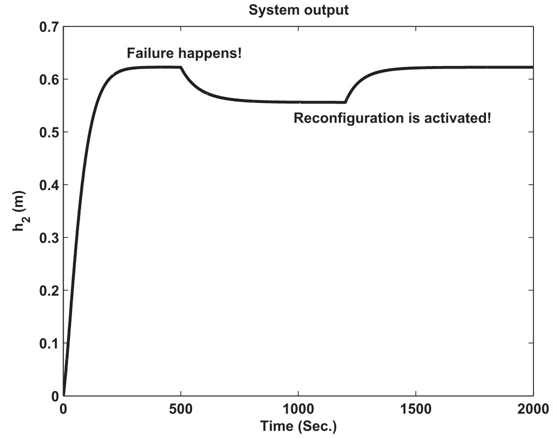

Fig. 8 The system output is recovered using a dynamic reconfiguration block.

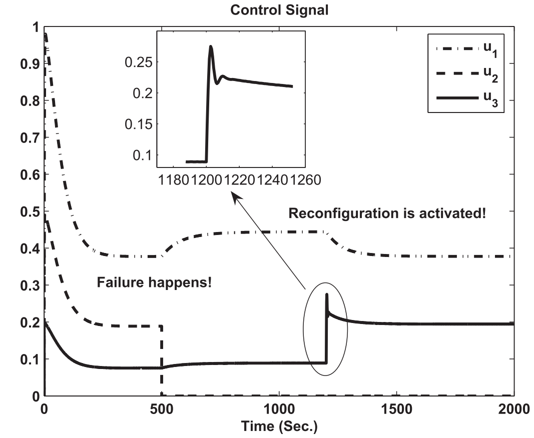

Fig. 9 Variation of control signal when dynamic reconfiguration is used.

in which M should be designed such that A_R is Hurwitz, and $-M(A - B^f M)^{-1}B + I = N$ where N is the solution of (13) for state recovery or the solution of (14) for output type.

5. Reconfiguration in the Presence of Constraints on Control Signals

In the previous part, the reconfiguration-based FTC was addressed and the reconfiguration block was designed based on a novel structural method. In this part, the design of the reconfiguration block in the presence of constraint on actuators is considered.

5.1 Energy Constraint

Assume that the reconfiguration block (13)/(14) should be designed subject to the constraint in the total energy delivered to the system, i.e. the recovery should be done using minimum energy. It means that $\|u^f\|$ should be minimized. Since $u^f = Nu$ and the previously designed control signal u is acceptable, the constraint can be converted to minimization of the norm of N; i.e. the largest singular value of N should be minimized. Therefore, the acceptable static reconfiguration block for state recovery is derived from the following problem:

$$\begin{aligned}
&Minimize && \left\|B^f N - B\right\|^2\\
&subject\,to: && \bar{\sigma}(N) < \gamma^2
\end{aligned} \tag{22}$$

This problem can be transformed to a standard LMI problem as:

$$\begin{aligned}
&\left\|B^f N - B\right\|^2 < \alpha^2\\
&\to (B^f N - B)^T (B^f N - B) < \alpha^2 I\\
&\to \begin{pmatrix} \alpha I & (B^f N - B)\\ (B^f N - B)^T & \alpha I \end{pmatrix} > 0
\end{aligned} \tag{23}$$

On the other hand, the singular value minimization problem can be represented as:

$$\bar{\sigma}(N) < \gamma^2 \to \begin{pmatrix} \gamma I & N\\ N^T & \gamma I \end{pmatrix} > 0 \tag{24}$$

Therefore, the problem (22) can be converted to the following standard LMI:

$$\begin{aligned}
&Minimize \quad w_1\alpha^2 + w_2\gamma^2\\
&subject\,to:\\
&\begin{pmatrix} \alpha I & (B^f N - B)\\ (B^f N - B)^T & \alpha I \end{pmatrix} > 0\\
&\begin{pmatrix} \gamma I & N\\ N^T & \gamma I \end{pmatrix} > 0
\end{aligned} \tag{25}$$

in which w_1 and w_2 are weighting factors which are degree of freedom for the designer. The similar problem for output recover can be easily defined and solved according to (14) as:

$$\begin{array}{ll} \mathit{Minimize} & \left\| B_c^f N - B_c \right\|^2 \\ \mathit{subject\,to}: & \bar{\sigma}(N) < \gamma^2 \end{array} \tag{26}$$

5.2 Saturation

The more realistic constraint on the control signals is the saturation. The process of reallocating the control signals may cause the actuators to get saturated. In this section, the control reconfiguration problem in the presence of limit on the control signals is considered. Assume the following constraint on the control signals:

$$\delta_i^{\min} \leq u_i \leq \delta_i^{\max}, \quad i = 1, 2, \ldots, m \tag{27}$$

It is assumed that the constraints are satisfied in the fault-free control loop. Now in the faulty one, the reconfiguration block should be designed such that (27) is still satisfied, i.e.:

$$\delta_i^{\min} \leq u_i^f \leq \delta_i^{\max}, \quad i = 1, 2, \ldots, m \tag{28}$$

Definition 2 (Free capacity factor for an actuator): Assume that a system is successfully controlled and the acceptable performance is achieved. The amount of increase in a control signals of an actuator before its saturation is called “Free Capacity Factor” and can be calculated as:

$$k_i^{ca} = \frac{\delta_i^{\max} - u_i^{ss}}{\delta_i^{\max}} \tag{29}$$

in which u_i^{ss} is the value of the control signal u_i in the steady state condition.

Therefore, the control signal of the actuator i in the fault free control loop can be presented as:

$$u_i^{ss} = (1 - k_i^{ca})\delta_i^{\max} \tag{30}$$

On the other hand, the equation $u^f = Nu$ means that each control signal of the faulty plant is achieved by linear combination of control signals of the previous healthy system as:

$$\begin{aligned} u_i^f &= n_{i1}u_1 + n_{i2}u_2 + \cdots + n_{im}u_m \rightarrow \\ u_i^{fss} &= n_{i1}(1 - k_1^{ca})\delta_1^{\max} + n_{i2}(1 - k_2^{ca})\delta_2^{\max} + \\ &\quad + \cdots + n_{im}(1 - k_m^{ca})\delta_m^{\max} \\ &= \left[\, n_{i1}(1-k_1^{ca}) \;\; n_{i2}(1-k_2^{ca}) \;\; \cdots \;\; n_{im}(1-k_m^{ca}) \,\right] \begin{bmatrix} \delta_1^{\max} \\ \delta_2^{\max} \\ \vdots \\ \delta_m^{\max} \end{bmatrix} \end{aligned} \tag{31}$$

Considering the constraint in which $u_i^{fss} \leq \delta_i^{max}, i = 1, 2, \ldots, m$, (31) can be written as:

$$\begin{gathered} [NK^{ca} - \delta^{\max}] < 0 \\ N = \begin{bmatrix} n_{11} & n_{12} & \cdots & n_{1m} \\ n_{21} & n_{22} & \cdots & n_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ n_{m1} & n_{m2} & \cdots & n_{mm} \end{bmatrix} \\ K^{ca} = \begin{bmatrix} 1 - k_1^{ca} \\ 1 - k_2^{ca} \\ \vdots \\ 1 - k_m^{ca} \end{bmatrix}, \; \delta^{\max} = \begin{bmatrix} \delta_1^{\max} \\ \delta_2^{\max} \\ \vdots \\ \delta_m^{\max} \end{bmatrix} \end{gathered} \tag{32}$$

Therefore, the reconfiguration block for state recovery will be the solution of the following LMI problem:

$$\begin{array}{ll} \mathit{Minimize} & \left\| B^f N - B \right\|^2 \\ \mathit{subject\,to}: & [NK^{ca} - \delta^{\max}] < 0 \end{array} \tag{33}$$

Similarly, the output recovery problem using the static reconfiguration block in the presence of saturation constraint can be formulated as follows:

$$\begin{array}{ll} \mathit{Minimize} & \left\| B_c^f N - B_c \right\|^2 \\ \mathit{subject\,to}: & [NK^{ca} - \delta^{\max}] < 0 \end{array} \tag{34}$$

Example 2: Assume that in example 1, the output recovery should be achieved under a saturation limit of 0.17 on the control signal u_3. Solving (34), the following static reconfiguration block N is achieved for this constrained optimization problem:

$$N = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0.6301 & 0.4619 \end{bmatrix}$$

Figures 10 and 11 show the output and control signal of the reconfigured faulty control loop in presence of this value of N as the reconfiguration block, respectively. Figure 11 indicates that the control signal u_3 meets the constraint. However, it is clear from Fig. 10 that the system output is not exactly recovered to its value before the failure. Though it is still acceptable.

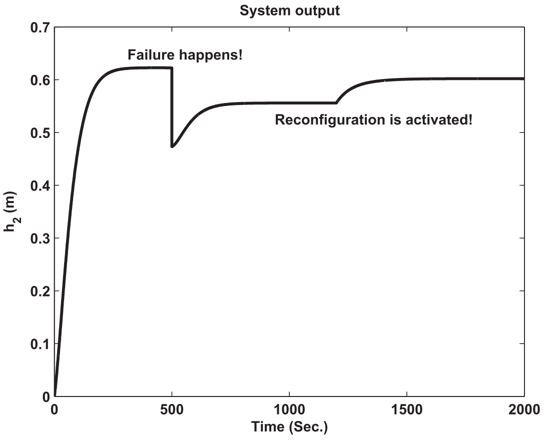

Fig. 10 Recovery of the system output in presence of saturation constraint on control signal u_1.

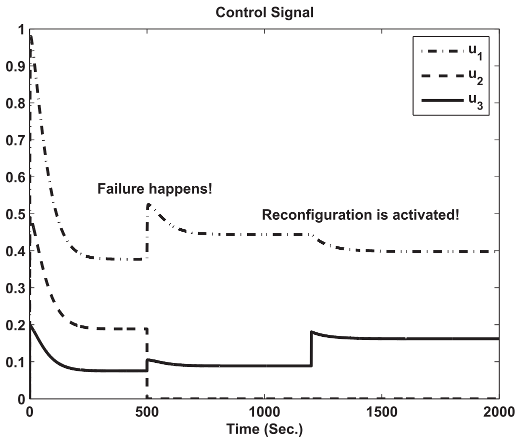

Fig. 11 Control signals for weak recovery of Fig. 10.

6. Conclusions

In this paper, a novel approach for control reconfiguration in the actuator failure condition based on the system structure was proposed. The method was based on reallocating the role of the failed actuator among the remaining healthy ones. Using this, the static reconfiguration block was designed to implement the state or output recovery of the faulty plant. Also, to improve the flexibility of the reconfiguration block respecting the constraints in the control loop, a dynamic reconfiguration block was proposed, but its performance was at best the same as the static one in the steady state condition. It is shown that the dynamical one has an extra degree of freedom to satisfy the constraints of the control loop. Finally, the reconfiguration problem was solved in the presence of constraints on the energy delivered to the actuators and also in the presence of saturation. All the achievements were concluded in theorems and were fully supported by the simulations.

References

[1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, Diagnosis and fault tolerant control, 2nd ed., Springer, 2006.

[2] M.A. Wicks and R.A. DeCarlo, "An energy approach to controllability," 27th Conference on Decision and Control, Texas, USA, 1988.

[3] J.H. Richter, T. Schlage, and J. Lunze, "Control reconfiguration of a thermofluid process by means of virtual actuator," IET Control Theory Appl., vol.1, no.6, pp.1606–1620, 2007.

[4] M. Staroswiecki, "On reconfigurability with respect to actuator failures," 15th Triennial IFAC World Congress, Barcelona, Spain, 2002.

[5] A.M. Amani, A. Afshar, and M.B. Menhaj, "Fault tolerant networked control systems subject to actuator failure using virtual actuator technique," to be presented in 15th IFAC World Congress, Milan, Italy, 2011.

[6] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," Annual Reviews in Control, vol.32, pp.229–252, 2008.

[7] T. Steffen, Control Reconfiguration of Dynamical Systems, Springer, 2005.

[8] B.M. Gonzalez-Contreras, D. Theilliol, and D. Sauter, "Actuator fault tolerant controller synthesis based on second order information," Proc. ECC 2007, Greece, 2007.

[9] B.M. Gonzalez-Contreras, D. Theilliol, and D. Sauter, "Performance evaluation of networked control systems based on the gramian controllability," 3rd International Workshop on Networked Control Systems: Tolerant to Faults, Nancy, France, 2007.

[10] D. Theilliol, C. Join, and Y. Zhang, "Actuator fault tolerant control design based on a reconfigurable reference input," Int. J. Appl. Math. Comput. Sci., vol.18, no.4, 2008.

[11] 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, Aug. 2010.

[12] Y. Guo, Y. Zhang, and B. Jiang, "Multi-model-based flight control system reconfiguration control in presence of input constraints," 8th World Congress on Intelligent Control and Automation, China 2010.

[13] R. Nazari, A. Yetendje, and M.M. Seron, "Fault-tolerant control of a magnetic levitation system using virtual-sensor-based reconfiguration," 18th Mediterranean Conference on Control and Automation, Morocco, 2010.

Appendix A

Define the following dynamical equations for a linear time-invariant system:

$$\dot{x}(t) = Ax(t) + bu(t) \tag{A· 1}$$

in which $x \in R^n$ is the state vector of the system and $u \in R^n$ is the input signal. Assume that Λ represents the set of eigenvalue of the system matrix $A_{n\times n}$ and $e_1, e_2, \ldots, e_n$ is an orthonormal basis for R^n in which the i^{th} element of the vector e_i is "1" and the others are zero. It is proved in [2] that the effect of the input signal on the system state can be represented as:

$$x(t) = \sum_{i=1}^{n} \zeta_i e_i, \qquad \zeta_i = b_i \int_0^t e^{\Lambda(t-q)} u(q)dq \tag{A· 2}$$

in which, the coefficients b_i are resulted form projection of b onto the space spanned by e_i's as follows:

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} = e_1 b_1 + e_2 b_2 + \ldots + e_n b_n \tag{A· 3}$$

In other words, ζ_i represents the distance the state has travelled in the direction of the eigenvector e_i [2]. It is clear from (A· 2) that if $\exists i; b_i = 0$ then $\zeta_i = 0$, which means that the input signal can not push the system states in the direction of e_i, i.e. the system is not state controllable in this direction.

Lemma A1: The system (A· 1) is not state controllable in the direction of e_k if b has not any element in this direction i.e. $b_k = 0$ in (A· 3).

The following example clarifies this lemma.

Example A1: Assume the following systems:

$$\dot{x}(t) = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

It can be easily seen that the system is not full state controllable. The eigenvectors of the system matrix are:

$$e_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad e_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Decomposition of b in this coordination will result: $b = e_1$, i.e., the element of b in the direction of e_2 is zero. Therefore, the system is not state controllable in this direction.

For the case of multi-input system,

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{A·4}$$

The input matrix can be presented by its columns as:

$$B_{n\times m} = \begin{bmatrix} b_1 & b_2 & \cdots & b_m \end{bmatrix} \tag{A·5}$$

The effect of each input on the system state can be represented according to the previous part:

$$\begin{aligned} & b_k = e_1 b_{k1} + e_2 b_{k2} + \ldots + e_n b_{kn} \\ & \rightarrow x_k(t) = \sum_{i=1}^{n} \zeta_{ki} e_i \end{aligned} \tag{A·6}$$

By superposition, the effect of all system inputs on the system state can be concluded as:

$$x(t) = \sum_{i=1}^{n} \sum_{k=1}^{m} \zeta_{ki} e_i \tag{A·7}$$

It means that each of the inputs can move the system state somewhat in each of the e_i directions. Hence, the total movement of the system state is resulted by combination of these parts. It is important to note that each input signal would not necessarily affect the system state in all the directions.

On the other hand, considering the single output case for simplicity, the vector C can be projected onto the space spanned by e_i^T's as follows:

$$C = c_1 e_1^T + c_2 e_2^T + \cdots + c_n e_n^T \tag{A·8}$$

Considering (A· 7) for system state, the system output can be represented as follows:

$$\begin{aligned} & y(t) = Cx(t) \rightarrow y(t) = \sum_{i=1}^{n} \sum_{k=1}^{m} \sum_{j=1}^{n} (c_j e_j^T) \zeta_{ki} e_i \\ & = \sum_{i=1}^{n} \sum_{k=1}^{m} \sum_{j=1}^{n} \zeta_{ki} c_j e_j^T e_i \end{aligned} \tag{A·9}$$

From definition of the set of e_i's,

$$e_j^T e_i = \begin{cases} 1, & if \;\; j \neq i \\ 0, & if \;\; j = i \end{cases} \tag{A·10}$$

Therefore, (A· 9) can be simplified to:

$$y(t) = \sum_{i=1}^{n} \sum_{k=1}^{m} \zeta_{ki} c_i \tag{A·11}$$

Table A· 1 List of symbols and abbreviations.

symbol	meaning
$u/x/y$	input/state/output of healthy system
$u^f/x^f/y^f$	input/state/output of faulty system
$u_c/x_c/y_c$	input/state/output of the controller
x_R	state of reconfiguration block
G_p	Healthy plant
G_p^f	Faulty plant
G_c	Controller
G_R	Reconfiguration block
MISO	Multi Input Single Output system
MIMO	Multi Input Multi Output system
$x < 0 (x \in R^n)$	All elements of the vector x are negative.

Appendix B

Augmenting (5) and (8), the reconfigured plant G_{Rp} in Fig. 1 B can be represented as:

$$\begin{aligned} \begin{bmatrix} \dot{x}_R \\ \dot{x}^f \end{bmatrix} &= \begin{bmatrix} A_R & 0 \\ B^f C_R & A \end{bmatrix} \begin{bmatrix} x_R \\ x^f \end{bmatrix} \\ &\quad + \begin{bmatrix} B_R \\ B^f D_R \end{bmatrix} u_R \\ y_R(t) &= \begin{bmatrix} C & C \end{bmatrix} \begin{bmatrix} x_R \\ x^f \end{bmatrix} \end{aligned} \tag{A·12}$$

Applying the following similarity transformation:

$$\begin{bmatrix} x_R \\ v \end{bmatrix} = \begin{bmatrix} I & 0 \\ I & I \end{bmatrix} \begin{bmatrix} x_R \\ x^f \end{bmatrix} \tag{A·13}$$

(A· 12) will be transformed to:

$$\begin{aligned} \begin{bmatrix} \dot{x}_R \\ \dot{v} \end{bmatrix} &= \begin{bmatrix} A_R & 0 \\ A_R + (B^f C_R - A) & A \end{bmatrix} \begin{bmatrix} x_R \\ v \end{bmatrix} + \begin{bmatrix} B_R \\ B_R + B^f D_R \end{bmatrix} u_R \\ y_R(t) &= \begin{bmatrix} 0 & C \end{bmatrix} \begin{bmatrix} x_R \\ v \end{bmatrix} \end{aligned} \tag{A·14}$$

Considering (2), the dynamical equation of the closed loop reconfigured control system can be presented as:

$$\begin{aligned} \begin{bmatrix} \dot{x}_R \\ \dot{v} \\ \dot{x}_c \end{bmatrix} &= \begin{bmatrix} A_R & 0 & B_R C_c \\ A_R + (B^f C_R - A) & A & (B_R + B^f D_R) C_c \\ 0 & B_c C & A_c \end{bmatrix} \begin{bmatrix} x_R \\ v \\ x_c \end{bmatrix} \\ y^f(t) &= Cx^f = \begin{bmatrix} -C & C & 0 \end{bmatrix} \begin{bmatrix} x_R \\ v \\ x_c \end{bmatrix} \end{aligned} \tag{A·15}$$

which is stable if (9) is Hurwitz.

Appendix C

Table A.1 represents a list of some of symbols and abbreviations used in this paper.

Ali Moradi Amani received the B.Sc. degree from Amirkabir University of Technology, Tehran, Iran, in 2001, and the M.Sc. degree from University of Tehran, Tehran, Iran, in 2003, both in electrical engineering. He is currently working toward the Ph.D. degree in electrical engineering at Amirkabir University of Technology. His main research interests includes, Fault tolerant control, Large scale systems, Networked control systems, Fault Detection and Diagnosis and industrial control and automation.

Ahmad Afshar received the Ph.D. degree in electrical engineering from Manchester university in 1991 and subsequently started his career as a member of staff with university of Petroleum in the department of electrical engineering. He joined in the electrical engineering department of Amirkabir university of technology in 1998. He is currently an Assistant Professor in the control engineering group of this department and is the head of the computational intelligence and large scale systems research laboratory and also the head of automation and IT research laboratory. His research fields include large scale systems, intelligent crisis control, multi-agent systems, network based control systems, sensor networks and decision support systems.

Mohammad Bagher Menhaj received his Ph.D. degree from school of electrical and computer engineering at OSU in 1992. After completing one year with OSU as postdoctoral fellow, in 1993, he joined Amirkabir University of Technology, Tehran, Iran. December 2000 to Aug. 2003, he was with school of Electrical and Computer Engineering and Department of Computer Science at OSU as visiting faculty member and research scholar. He is author and coauthor of more than 400 technical papers, and six books: Fundamentals of Neural Networks, 1998, Application of Computational Intelligence in Control, 1998, Neural Networks, 2000, and Fuzzy Computations, 2007, Evolutionary Computation and Fuzzy Control (under print) all in Persian. He has also been project director for many industrial projects in the areas such as crisis control management, communication traffic control, real time simulator design, flight control and navigation systems, and satellite attitude determination and control systems, sponsored by private and government institutions. His main research interests are: theory of computational intelligence, learning automata, adaptive filtering and their applications in control, power systems, image processing, pattern recognition, and communications, and other areas of interests are: theory of rough set and knowledge discovery.