

Implementation of Reconfiguration Management in Fault-Adaptive Control Systems

Gyula Simon^{*,#}, Tamás Kovács házy^{*}, Gábor Péceli^{*}, Tivadar Szemethy[#], Gábor Karsai[#], Ákos Lédeczi[#]

^{*}Department of Measurement and Information Systems,
Budapest University of Technology and Economics, H-1521 Budapest, Hungary
Phone +36-1-463-2057, Fax +36-1-4634112, Email: {simon,khazy,peceli}@mit.bme.hu

[#]Institute of Software-Integrated Systems, Vanderbilt University, Nashville, TN 37235, USA
Phone +1-615-343-7472, Fax +1-615-343-7440, Email: {simon,tiv,gabor,akos}@isis-server.vuse.vanderbilt.edu

Abstract – Fault adaptive systems must adapt and reconfigure themselves to the changes in the environment or the system itself, and have to maintain operation even in case of system failures. In order to avoid performance degradation due to system reconfigurations, adequate reconfiguration management is necessary. This paper describes a fault-adaptive control system with multi-layer control and a reconfiguration management system.

Keywords – reconfigurable systems, hybrid systems, transient reduction, fault-adaptive control

I. INTRODUCTION

Fault tolerant and fault adaptive systems play an important role in today's safety critical applications such as fly-by-wire and drive-by-wire vehicles, and the environmental control of spacecrafts. In case of failures, such systems must remain operational, and the basic and necessary capabilities must be maintained. To achieve this goal fault adaptive control systems [1] are designed to diagnose and isolate faults, make decisions, and reconfigure themselves in order to maintain the operation in a satisfactory level.

The operation of such complex applications can often be modeled and described as hybrid dynamical systems [2], [3], which have discrete operational modes. Each mode corresponds to a dedicated continuous dynamic system. Transition between these modes (i.e. mode change) is considered as reconfiguration of the complex system.

The reconfiguration can be either structural or parametric, but unfortunately both can cause transients as undesired side effects. Recently transient management techniques have been proposed to suppress or decrease the reconfiguration transients in open loop [4] and closed loop [5], [6] scenarios. These methods can either be passive methods, where the proper choice of the structure ensures the low level of the transients; or active methods, where run-time interactions (e.g. state variable initialization or anti-transient signal injection) help to suppress the transient effects.

In this paper an implementation of a transient management system is proposed in a complex fault-adaptive control application. A short overview is given on transient

management in fault adaptive systems in Section II. In Section III the proposed control architecture is described, with the details of the transient management subsystem in Section IV. In Section V a simulation example is presented.

II. TRANSIENT MANAGEMENT IN FAULT ADAPTIVE CONTROL SYSTEMS

The problem of transient management is recognized in the control literature, see, e.g. [10], and multiple-step reconfiguration is proposed as a possible solution. In some other works, see e.g. [11], transients are identified as convergence trajectories of adaptive schemes. In the signal processing literature some attempts were made to reduce the transients of variable filters [12].

Different solutions have been proposed lately to decrease the transient effects in reconfigured dynamic systems. The importance of proper structures is addressed in [4]. Active transient management algorithms were also proposed [5], [6].

Fault adaptive control systems are being developed, using reconfigurable monitoring and control systems to maintain the operation after faults [1]. The transient management in such systems has not been addressed nor solved yet, although in certain applications the performance of the system may degrade severely because of the side effects of the reconfiguration. The structure proposed in this paper undertakes the handling of this problem.

III. CONTROLLER ARCHITECTURE

The controller architecture used in our fault-adaptive control system is shown in Fig. 1. Similarly to the concept described in [7], a multi-layer approach is used to separate low-level and high-level control objectives. The physical plant is controlled by the regulators (low-level controllers), while the overall control objectives are managed by the supervisory (high-level) controller. The separation of control objectives provides a clean hierarchical controller structure and enables the design of controllers in different levels. The control close

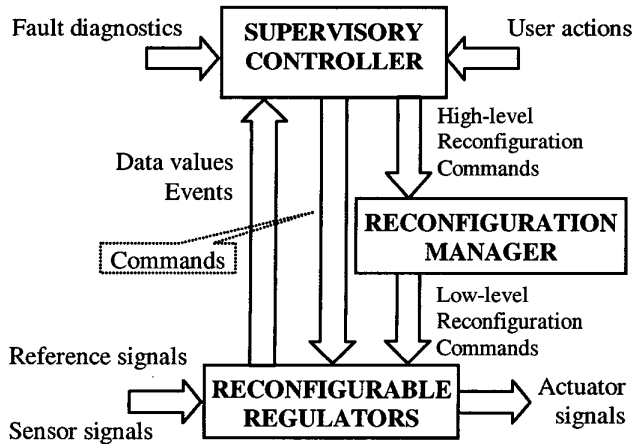


Fig. 1. The control architecture and data flow graph of the fault adaptive control system

to the physical system can be distributed and simple since each regulator takes care of only a particular part of the whole control objective (e.g. control of one aileron actuator of an airplane). The supervisory controller deals only with high-level control objectives by adequately configuring the system components.

The supervisory controller receives information from other high-level system components (e.g. from the fault diagnosis subsystem or from the system monitor with human interactions) and it monitors the regulators and the plant through sampled data values and discrete events, as shown in Fig. 1. Based upon the observed information the supervisory controller can initiate the reconfiguration of the regulators (and possibly the plant as well). The reconfiguration is managed through an intermediate layer, the Reconfiguration Manager. It receives reconfiguration commands from the supervisory controller, and translates them into low-level reconfiguration command sequences, which are executed by the regulators. The reconfiguration manager also takes care of the transient management of the reconfiguration process. Note that the reason of a reconfiguration can be either a regular change of the operational mode, or a result of a detected fault.

A convenient description of systems with different modes of operation is the hybrid finite state machine (HFSM) [8], [9], as shown in Fig. 2. The states of the HFSM correspond to the modes of operation, and the transitions between states correspond to mode changes. The HFSM is hierarchically structured, and thus supports a compact, comprehensible and logical system description.

The HFSM-based system representation gives the possibility to represent the supervisory controller itself as a finite state machine. In its states the appropriate regulator descriptions can be found, while the state transitions – in addition to the mode change – describe the transient management techniques.

A. Supervisory layer

The supervisory controller is represented by a hierarchical finite state machine, using the state chart formalism of Harel [13]. Each state on the highest hierarchy level represents an operating mode (i.e. a system configuration). Sub-states represent different phases of an operating mode.

Based on the information from the environment, the fault diagnostics unit, or the user, the supervisory controller evaluates state transitions and thus initiates system reconfiguration. The state incorporates the description of the current reconfiguration, i.e. the controller (and occasionally the plant) configuration, and the connection layout. The state transition describes the reconfiguration mechanism to be used by the Reconfiguration Manager (see example in Section V).

Controller Description. The regulator components can be described by one of the three possible description modes: Set, Design, and Construct. From the controller description the Reconfiguration Manager is able to synthesize the reconfigured regulator, using different amounts of design-time and run-time information.

- **Set:** The controller is described by explicitly defining the structure and its parameters.
- **Design:** The controller's structure is given, and the parameters are calculated when the system is reconfigured to provide optimal performance in some sense. The type and measure of performance are parameters, when appropriate.
- **Construct:** Only the quality criteria (i.e. the measure of performance) are given by the designer, the optimal structure and the corresponding parameters are determined on-line, when the reconfiguration happens.

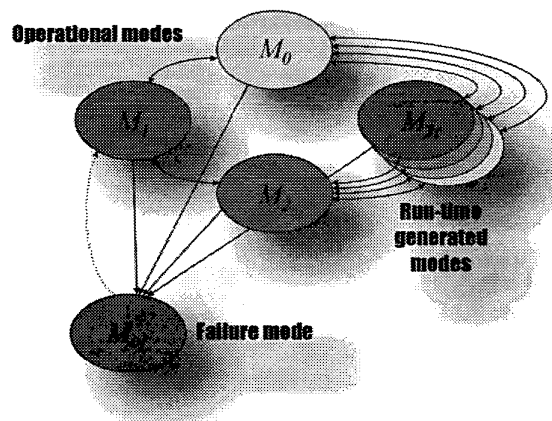


Fig. 2. Example of a hybrid system with modes of operation

B. Regulatory layer

The primary task of the regulatory layer is to perform the lower level control of the physical system. Thus the regulators receive sampled plant data from the sensors and generate control signals to the actuators, the control law depending on their current configuration. In addition to the basic control activity, the regulator components support the upper layers to perform the reconfiguration.

The regulator components contain

- low-level controllers (either software components, or different hardware representations) performing the associated control task, with basic ability to support the reconfiguration (start, stop, parameter/state update, etc.), and
- methods to support advanced reconfiguration and transient management techniques (e.g. regulator components can provide information on their capabilities, calculate their control and reconfiguration parameters, and also provide different performance measures). Note that these methods do not run on the low-level controllers.

IV. THE RECONFIGURATION MANAGER

The purpose of the reconfiguration manager is to evaluate high-level reconfiguration commands initiated by the supervisory controller. The activity of the Reconfiguration Manager has three different aspects: topological, structural and transient management. The topological management covers the creation of the system topology (removal of old components and connections, creation of new components and connections in an adequate order). Loosely speaking, the topological management creates boxes with connections between them. The structural management fills the boxes with the actual system components; it determines and sets the structure and parameters for each system components. The transient management ensures the good transient behavior of the system during the reconfiguration. The three management activities are strongly coupled.

In the reconfiguration manager each reconfiguration mechanism is described by a state machine. When the supervisory controller initiates a reconfiguration, the appropriate state-machine in the Reconfiguration Manager is evaluated. Each state represents a simple activity:

Topological activities: remove component, remove connection, add component, add connection, configure component.

Structural activities: create controller based on the controller description received from the Supervisory Controller. The controller creation uses information and design methods provided by the regulator components.

Transient management activities: The undesired side effects of the reconfiguration can be decreased by various transient management techniques, e.g. blending, state variable initialization, anti-transient signal injection [5], [6]. The transient management technique for each component (or component group) is specified in the reconfiguration method. The reconfiguration manager uses the methods supplied by the regulator components.

Reconfiguration Description. The transient management method is described by either the *Define* or *Select* keywords.

- **Define [method]:** The transient management method is explicitly defined (e.g. bumpless state initialization). Only its parameters are calculated online.
- **Select [optimality criterion]:** The optimal method is selected, based upon the previous and the next system structure, the parameters, and the supported transient management method set. An optimality criterion is an input parameter.

The transient management methods use the information and design methods provided by the old and new regulator components.

V. EXAMPLE

As a test scenario the roll control of a simulated airplane is used. The airplane can drop loads (from the wings and from the body), thus the dynamics of the plane are 'reconfigurable'. The model also enables injection of different aileron faults to test fault-detection abilities. Based on the dynamics and the faults different control strategies can be used to provide optimal performance, so the roll controller should also be reconfigured.

The control loop contains the aircraft dynamics, a gyro position sensor, the controller and an aileron actuator (see Fig. 3). The implemented reconfiguration manager is a simplified version in the sense that no topological reconfiguration is required (there is only one controller, only its inner structure and parameters can be changed).

The test system is built in MATLAB using Stateflow graph [14] and Simulink models [15]. For each controller type the designer have to supply a handler function, in which the controller is described in terms of input-output behavior, parameter calculation methods, and the available transient management techniques. This handler file contains also

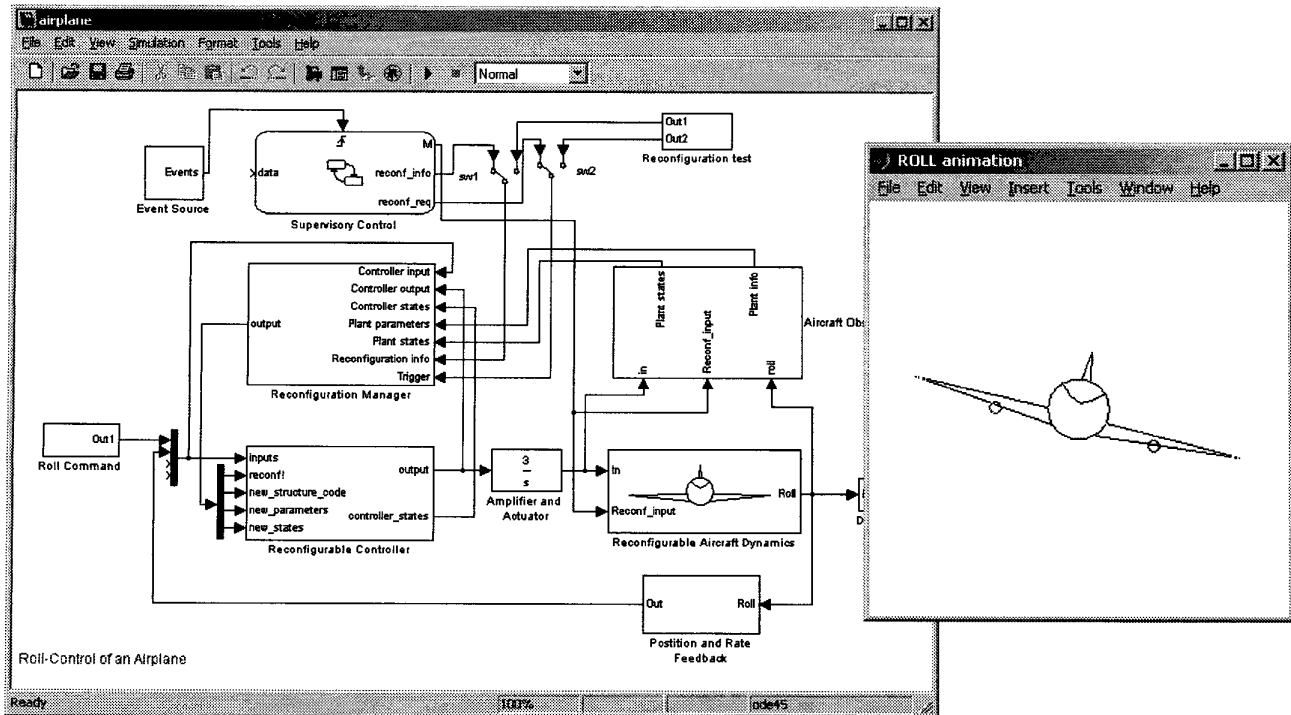


Fig. 3. The simulated aircraft roll control system with reconfigurable controller.

information on the supplied methods for the Reconfiguration Manager.

In the simulation example one load was dropped from each wing, so four operational modes were used, each of them corresponding to one mass distribution, as shown in Fig. 4. The controller descriptions (Set and Design methods of different lead-lag controllers) and reconfiguration descriptions (Define of state zero, state preserve, and bumpless state initialization) can be seen in the figure. The evaluation of the reconfiguration procedure is illustrated in Fig. 5.

VI. SUMMARY

A multi-layer transient management system was proposed, which can be used in reconfigurable fault adaptive control systems. The lower level regulator blocks are extended to provide additional services to support parameter calculation methods, reconfiguration methods, and controller design. The upper level supervisory layer represents the different operating modes of the system. The reconfiguration manager performs the reconfiguration of the regulatory layer, based on the commands received from the supervisory controller, and the available low-level reconfiguration methods, provided by the regulator components.

The reconfiguration and transient management procedures are described in the transient manager. Different description

levels facilitate the reconfiguration and transient management based on different amount of design-time information.

The concept was tested and illustrated in a simulated reconfigurable airplane roll controller.

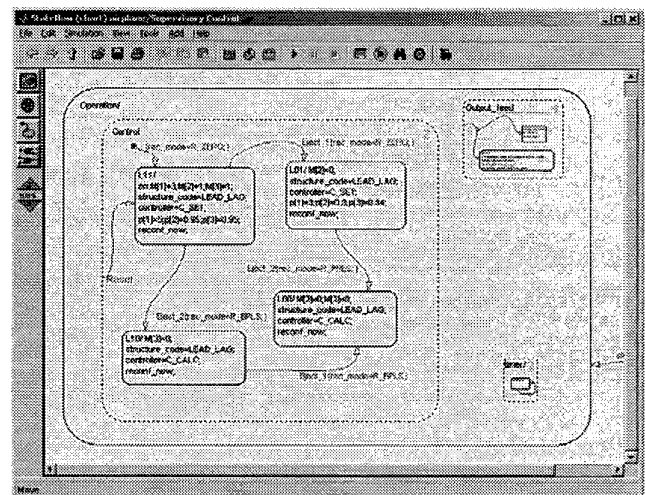


Fig. 4. The Supervisory Controller implemented as a MATLAB StateFlow graph. Each operating mode represents a mass distribution (i.e. a dynamic behavior) of the airplane.

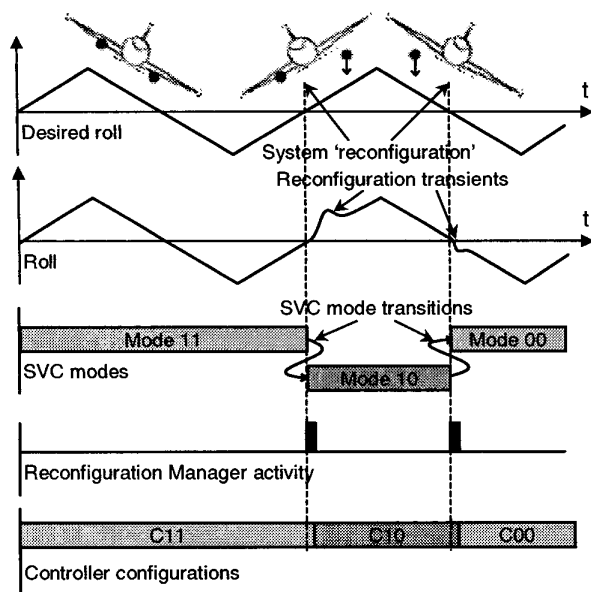


Fig. 5. Airplane roll example with three active operational modes

ACKNOWLEDGEMENT

This research was sponsored by DARPA (US) (F33615-99-C-3611) and by the Hungarian Ministry of Education (OM-FKFP 0654/2000).

REFERENCES

[1] Karsai G., G. Biswas, T. Pasternak, S. Narasimhan, G. Peceli, G. Simon, T. Kovacsazzy, "Fault-Adaptive Control: A CBS Application," Proceedings of the Eighth Annual IEEE International Conference on Engineering of Computer Based Systems (ECBS '01), Washington D.C., 18-19 April 2001, pp. 205-211

[2] Grossman, R.L., A. Nerode, A.P. Ravn, and H. Rischel, Eds., Hybrid Systems, Lecture Notes in Computer Science, no. 736, New York: Springer Verlag, 1993.

[3] Branicky, M.S., V. Borkar, S. Mitter, "A Unified Framework for Hybrid Control: Background, Model and Theory," Proc. 33rd IEEE Conference on Decision and Control, Lake Buena Vista, FL, USA, paper no.: LIDS-P-2239.

[4] Kovácszázy, T., G. Péceli, Gy. Simon, "Transients in Reconfigurable Signal Processing Channels," IEEE Trans. on Instrumentation and Measurement, Vol. 50, No. 4, pp. 936-940, Aug. 2001.

[5] Simon, Gy., T. Kovácszázy, G. Péceli, "Transients in Reconfigurable Control Loops," IEEE Instrumentation and Measurement Technology Conference, IMTC/2000, Baltimore, Maryland, USA, May 1-4, 2000, Vol. 3, pp. 1333-1337.

[6] Simon, Gy., G. Péceli, T. Kovácszázy, "Transient Reduction in Control Loops in Case of Joint Plant-Controller Reconfiguration," IEEE Instrumentation and Measurement Technology Conference IMTC/2001, May 21-23, Budapest, Hungary, 2001, Vol. 2, pp. 1172-1176.

[7] Parisini, T., S. Saccone, "Fault Diagnosis and Controller reconfiguration: an Hybrid Approach," Proceedings of the , IEEE ISIC/CIRA/ISAS Joint Conference, Gaithersburgh, Maryland, USA, Sep 14-17, 1998, pp. 163-168.

[8] Szatpanovits, J., D.M. Wilkes, G. Karsai, Cs. Biegl, L.E. Lynd, "The Multigraph and structural adaptivity," IEEE Transactions on Signal Processing, Vol.41, No.8, pp. 2695-2716, Aug. 1993.

[9] Narendra, K. S., J. Balakrishman, "Adaptive Control Using Multiple Models," IEEE Trans. on Automatic Control, Vol. 42, No.2, pp.171-187, Feb. 1997.

[10] Zhang, Y., J. Juang, "Design of Integrated Fault Detection, Diagnosis and Reconfigurable Control Systems," Proc. of the 38th Conference on Decision & Control, Phoenix, Arizona, USA, Dec. 1999, pp. 3587-3592.

[11] Sun, J., "A Modified Model Reference Adaptive Control Scheme for Improved Transient Performance," IEEE Trans. on Automatic Control, Vol. 38, No. 8, pp. 1255-1259, Aug. 1993.

[12] Valimaki, V., T.I. Laakso, "Suppression of transients in variable recursive filters with a novel and efficient cancellation method," IEEE Trans. on Signal Processing, Vol. 46, (1998), pp. 3408-3414.

[13] Harel, D., "Statecharts: A visual formalism for complex system," Science of Computer Programming, Vol.8, pp.231-274, 1987.

[14] Stateflow User's Guide. The MathWorks, Inc., Natick, MA, USA. 1999.

[15] Using Simulink. The MathWorks, Inc., Natick, MA, USA. 2000.