

An OSA-CBM Multi-Agent Vehicle Health Management Architecture for Self-Health Awareness

Cecilio ANGULO, Gustavo GONZÁLEZ, *Member, IEEE*, Cristóbal RAYA and Andreu CATALÀ

Abstract- Integrated Vehicle Health Management (IVHM) systems on modern aircraft or autonomous unmanned vehicles should provide diagnostic and prognostic capabilities with lower support costs and amount of data traffic. When mission objectives cannot be reached for the control system since unanticipated operating conditions exist, namely a failure, the mission plan must be revised or altered according to the health monitoring system assessment. Representation of the system health knowledge must facilitate interaction with the control system to compensate for subsystem degradation. Several generic architectures have been described for the implementation of health monitoring systems and their integration with the control system. In particular, the Open System Architecture - Condition-Based Maintenance (OSA-CBM) approach is considered in this work as initial point, and it is evolved in the sense of self-health awareness, by defining an appropriated multi-agent smart health management architecture based on smart device models, communication agents and a distributed control system. A case study about its application on fuel-cells as auxiliary power generator will demonstrate the integration.

Index Terms— Multi-agent systems, OSA-CBM, Self-health awareness, Smart adaptive systems.

1. INTRODUCTION

Vehicle health management [22], machinery monitoring and diagnostic systems are tasks related with maintenance. Main objectives in maintenance of systems and equipment are ensuring safety, equipment reliability, and reduction of costs. Traditionally, there have been two common maintenance philosophies employed: preventive maintenance (sometimes called schedule-based or event based) that uses statistics to estimate the machine behavior, leading to very conservative estimates of the probability of failure; and corrective (sometimes called restorative) maintenance philosophy, with the machine running until it fails and being then restored to good health; hence maintenance costs are reduced, but unexpected failures may result in longer than expected system down times.

Current state-of-the-art in aircraft preventive maintenance includes, for instance, the Joint Strike Fighter (JSF) program which has incorporated prognostics health management (PHM) into its design, using sensors, advanced processing and reasoning, and a fully integrated system of information and supplies management [2]. The on-board JSF PHM system is hierarchical, dividing the

aircraft into areas. Area data is generated by a mixture of dedicated, purpose-built sensors and analysis on existing control sensors to identify degradation and failures, compiled and correlated by area reasoners, and then correlated by system-level model-based reasoners.

For an in-deep review of the state-of-the art in vehicle health management, especially in aerospace, you can visit the webpage of the ISHEM First International Forum on Integrated System Health Engineering and Management in Aerospace [15].

The objective in Condition-Based Maintenance (CBM) is to accurately detect the current state of engineered systems and their operating environments and use that information for maintenance and prognosis activities [16]. A CBM system is intended to monitor the operation of a complex system and provide the operator or the autonomous control system with an accurate assessment of the system's current health. In the traditional CBM analysis flow path information is directed from a layer to the nearest one. A conceptual view of a general condition-based monitoring system is shown in Fig. 1.

There are numerous differences between collaboration in a computer-supported environment versus a face-to-face environment. Even though scientists are working hard to simplify collaboration by producing systems that provide virtual environments, and collaborators work hard to be cooperative, there are still unknown problems for collaborators to find and overcome when using a collaborative system. In building a collaborative system, conflict resolution is one of the most important problems to overcome.

Implied software in a traditional CBM system is application or hardware specific, no standardized, so it is difficult to upgrade it. Moreover, control and monitoring machinery is hard coded and wired, no distributed software is possible.

Standardization of an interface specification within the CBM community is the first step driving CBM suppliers to produce interchangeable hardware and software components, reducing cost and development time. In this sense, an Open System Architecture (OSA) is defined as a systems engineering approach that facilitates the integration and interchangeability of components from a variety of sources. An open system standard should consist of publicly available descriptions of component interfaces, functions, and behavior. A brief introduction about OSA-CBM systems [3] is introduced in Section 2.

Openness is a general interesting concept improving the CBM system design for health monitoring. However an OSA-CBM monitoring system is not enough when self-

Manuscript received October 15, 2006. This research was supported by the Spanish Science and Education Ministry project DPI2006-15630-C02-01 "Adaptive Systems for Human Mobility Aid"

C. Angulo, C. Raya and A. Català are with the Knowledge Engineering Research Group (GREC), Department of Automatic Control, Technical University of Catalonia (UPC), 08800 Vilanova i la Geltrú, Spain. (e-mail: {cecilio.angulo; cristobal.raya; andreu.catala@upc.edu})

G. González is with the Agents Research Lab (ARLab), Institute of Informatics and Applications (IiA), University of Girona (UdG) and the EASY Innova Centre. Campus Montilivi, Building P4. E-17071, Girona, Spain. (e-mail: gustavog@eia.udg.es).

architecture, in front to the standard one, is that, in general, a component can access data directly from any layer. This fact favors distributed computing and, in particular, the use of multi-agent system paradigm.

The OSA-CBM development has defined three different module interfaces that may be used to query, configure and communicate data:

- Data interface. All the data relating to a particular event is defined by the data interface. This interface has a different structure for each layer (Dynamic information)
- Configuration interface. It contains the configuration information about the certain module. Semi-dynamic interface, it changes only as often as the configuration of the device changes.
- Explanation interface. Information about the device and its setup (Static information).

Configuration and Explanation interfaces, containing information about the module, its configuration and setup, will be used for the communication agents in the proposed architecture to define the Smart Device Model (SDM). This concept, adopted from the user-modeling [17] community, attempts to capture the behavior of every device monitored by the IVHM system. So, the focus on the health management system is moving from the diagnosis techniques to the devices behaviors and their interrelation.

3. TOWARDS SELF-HEALTH AWARENESS

Usually, research in health management has been focused on diagnosis techniques and tools; however it remains unsolved the problem about how to determine the loads and demands on critical components and subsystems, in order to prevent failures or adapt to device degradation. This determination for each planned mission is on the base of the definition of a ‘failure’, whether a failure is occurring in then system, and what it signifies for the mission. This new perspective allows to evolve health management from fault-tolerant or diagnostics systems towards self-health awareness. Hence, in [23] a generic architecture for the implementation of health monitoring within a complex system, the representation of system health information, and an approach for integrating health information with autonomous control is described. The core of the proposed architecture consists of a perception module and a response module. The perception module processes sensor information, commands, and messages to provide situational awareness for the controller. Depending on the source of the inputs to the perception module, the “situational awareness” may be internal, self-situational awareness, external situational awareness, or a combination of both. The perception module determines which behaviors should be activated to respond to the current situation. The response module accepts the behavior recommendations of the perception module and considers the necessary actions within the context of the current mission plan. Monitoring systems in this case must

employ smart sensors dealing with high level information as opposed to typical sensor data.

3.1 Developing the Architecture on Distributed Units

An extreme centralization on a ‘master’ element of health management routines does not allow that functionality and capabilities of specific devices can be completely exploited. Many hardware devices or subsystems are not sufficiently smart as they could be. The main causes are the information and communication overload of the central processing unit. Architecture introduced in [23] is very useful to obtain self-health awareness by reducing centralization, however ‘subsystem monitors’, a hierarchically superior controller managing several smart devices, completely determines the actuation on the ‘intelligent sensor nodes’; no interaction exists peer-to-peer between devices in the whole complex system.

Knowledge of interactions and dependencies between subsystems are key points in the architecture proposed in [20] jointly with a typical functional decomposition of the system. Here, the effort of the designer is directly proportional to the hierarchical dependencies between subsystems in the designed architecture. This work proposes a mechanism to allow interaction between intelligent sensor nodes and to reduce the charge in the subsystem monitor, even suppress it. In this way, knowledge of the subsystems’ dependencies will be captured by the nodes [7] and it will not be mandatory to design the subsystem monitors integrating interaction knowledge. In addition, their tasks will not be so critical [14].

Information processing in intelligent sensor nodes integrates sensor data fusion (perception) within nodes by using standard procedures, expert systems or some other technologies. Information is transmitted through messages without restriction to send only collected data, but also augmented information. (Fig. 2).

3.2 Smart Device Modeling

Smart User Models [11] have demonstrated their efficacy improving the quality of services personalization reducing the overload of processed information and capturing the behavior of the user in the next generation of open, distributed and networked environments. The aim of the proposed architecture is to move from the user-centered to a device-centered objective using the flexibility of intelligent agents in order to capture the most relevant functionalities and behaviors of a device through an adaptive process. Mainly, a multi-agent architecture must be developed in order to manage services and device preferences in several complex systems (i.e. domains).

Context is a multi-dimensional parameter that includes time, place, states among other variables. Context-aware health management (i.e. self-health awareness) can be achieved through internal representations of the devices in the domain, i.e. device models. Such representations can be

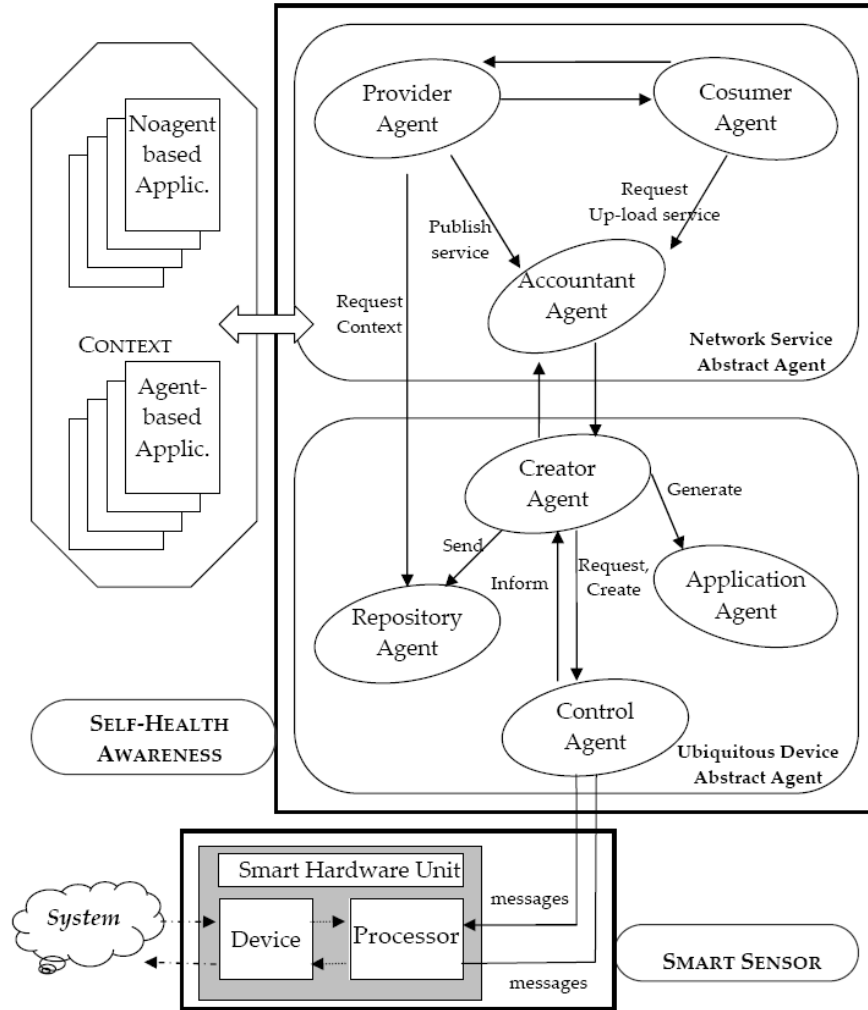


Fig. 3. Multi-agent System Architecture for the Smart Device Model. Two abstract agents exist: the *Network Service Abstract Agent (NSAA)* and the *Ubiquitous Device Abstract Agent (UDAA)*. The *NSAA* provides capabilities of autonomy. The *UDAA* gives initialization, identification, interoperability, control, coordination and management of the specific device preferences.

information derived from the unfeasible analysis that arises when are evaluated all the possible relevant alternatives to improve the *SDM*. We have choose this method to simplify our approach and by its practical implementation.

Therefore, $DM_{(j,k)}$ for each application are defined by shifting information from and to $DM_{(j,k)}$ of different existing domains according to the weighted graphs $G(SDM_{(j)}, DM_{(j,k)})$ defined by each application where device runs.

Finally, we want to stress that there are several methods to deal with multi-attribute decision analysis, so, we encourage the reader to read [5] to obtain a more extended revision of these techniques.

4. MULTI-AGENT SMART MANAGEMENT ARCHITECTURE

To support our Self-Health Awareness theoretical approach on a networked (even a web-based) application, it is proposed a multi-agent architecture defined at two main levels of abstraction, following the design defined in

[10]. At the highest level, two abstract agents exist: the *Network Service Abstract Agent (NSAA)* and the *Ubiquitous Device Abstract Agent (UDAA)*. The *NSAA* provides capabilities of autonomy regarding the automatic discovering of services in the network for the device [6]. It communicates with the applications in a specific domain. When applications are not agent-based, a *wrapper agent* operates like a middleware between the *NSAA* and the application. The *UDAA* gives initialization, identification, interoperability, control, coordination and management of the device preferences allowing a flexible and autonomous device-agent interaction.

Coordination between *NSAA* and *UDAA* is established mainly by two mechanisms:

- (i) *NSAA* requests to *UDAA* personalized information to deal with the applications in the environment (health management)
- (ii) *UDAA* receives information from *NSAA* regarding the success of the interaction.

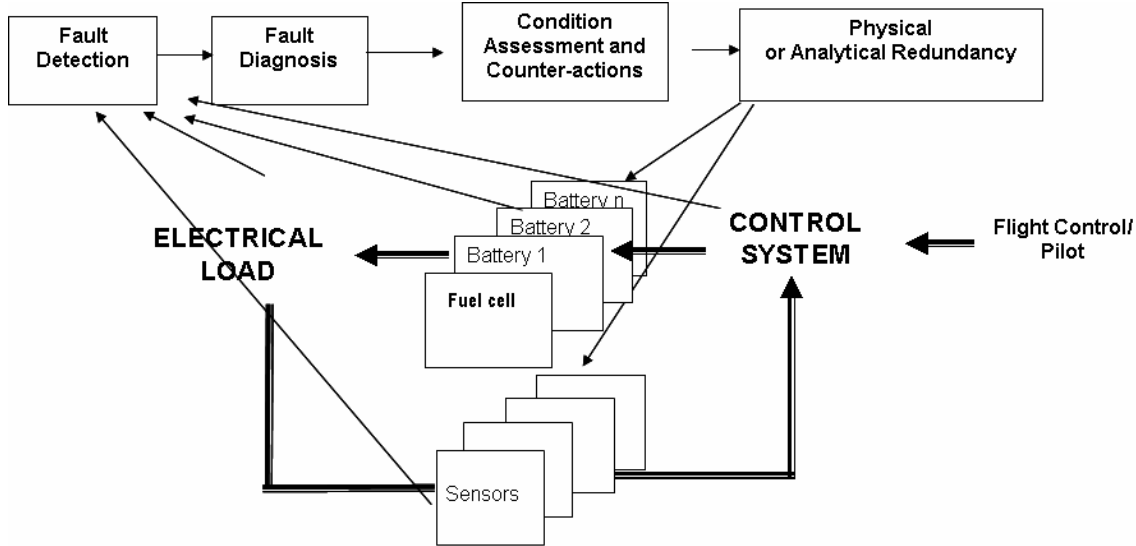


Fig. 4. An illustrative scheme about the fuel cell use in the case of study.

Such latter agent gathers the current information about the fuel cell FC_1 in the 'Airbeing 747 domain' (the domain for a specific type of aircraft), when the device was in a different airplane of the same series, and sends it to the *Repository Agent* of the IHVM of this airplane, in order to increase knowledge. Then, the *Control Agent* creates an *Application Agent* for the aircraft Airbeing 747 health management system, and registers the aircraft Airbeing 747 application to the *Control Agent*.

In fact, it is not necessary to physically move the fuel cell. It can be used as a training device for new airplanes flying on certain routes (i.e. context-awareness).

As a second example, let's suppose that the *NSAA Provider Agent* gathers information about new IVHM systems in the 'Airbeing 747 domain'. That is the case, the FC_1 fuel cell has used until now a health management system of a certain airplane, and the *Provider Agent* has discovered an IVHM about an airplane managing well when a fault in the hydrogen supply occurs. Since the fuel cell is observing that some degradation exists in the hydrogen supply system (contextual information), the *Provider Agent* believes that such information can be interesting for the device. Hence, the *NSAA* requests to the *UDAA* about the possibility of generating a new application on this new health management system.

Finally, after a while the fuel cell FC_1 is moved to an Airbeing A-380 aircraft, a different Device Model DM . Newly, the *Control Agent* of the IHVM system log in the device. In this case, at the *UDAA*, the *Control Agent* prompts the fuel cell with the information regarding Airbeing A-380 aircrafts and the fuel cell selects one application. Then, the *Control Agent* creates an *Application Agent* to deal with the new application. The *Application Agent* looks in the network for a fuel cell that has deal with the new application. It is the case that the fuel cell FC_2 has already interacted with the new application (for the 'Airbeing A-380 domain'). So, the corresponding

Application Agent of FC_2 and FC_1 dialog. The *Application Agent* of FC_1 acquires the Device Model 'Airbeing A-380 domain' from the graph $G(SDM, DM_i)$ corresponding to the relationship of the SDM of FC_2 in the specified domain. Weighted graph G is adapted to the Smart Device Modeling of FC_1 and its *Application Agent* is ready to deal with the health management process.

In this latter example, communication is directly established between two similar devices. However this idea can be extended to deliberation between different devices in a same airplane executing the health management of a certain mission. For instance, dialogue can be established between fuel cell node, undercarriage node and passengers' comfort node when aircraft is landing.

6. CONCLUSIONS

Openness is a general interesting concept improving the CBM system design for integrated vehicle health monitoring. However an OSA-CBM monitoring system is not enough to obtain self-health awareness. Moving the focus of the research from the classical diagnosis techniques to devices, components can be considered in a distributed form, so multi-agent systems (MAS) paradigm fits the novel approach.

Defining smart devices models, a concept imported from the user-modeling literature, it has been possible to design an appropriated OSA-MAS CBM architecture for IVHM systems through the definition of the different communication agents and a distributed control system. This system avoids to exactly defining the functional characteristics of the smart devices and their integration within the whole system by allowing direct communication between smart devices.

A case study with three possible situations about its application on fuel-cells as auxiliary power generator demonstrating the integration between the main control module and the IVHM system has been presented.

