Send Orders for Reprints to reprints@benthamscience.ae

The Open Automation and Control Systems Journal, 2014, *6*, 1071-1075 1071

Open Access

Design of A Multi-agent System for Aero Engine Health Management

Ming Zhao[1,2,*] and Jing Wu[1]

[1]*School of Aeronautic Transportation, Shanghai University of Engineering Science, Shanghai, 201620, P.R. China*

[2]*School of Information Science & Technology, Donghua University, Shanghai, 201620, P.R, China*

Abstract: For the four processes of aero engine health management, which are composed of condition monitoring, diagnostics, prognostics and prescriptive action, a four-layer architecture of aero engine health management divided into data layer, transaction layer, decision layer and users layer, is designed based on multi-agent. Corresponding to the function of each layer, six agent roles are defined, such as condition monitoring, information fusion, fault diagnosis, task consultation, rules management and human-computer interaction. FIPA ACL (Agent Communication Language)is the foundation for multi-agent interaction. The various steps involved in developing a multi-agent system using JADE (Java Agent Development Framework) are discussed. This paper provides theoretical conception for intelligent monitoring and intelligent fault diagnosis of aero engine, and for further research of multi-agent collaboration on multi-agent system.

Keywords: Aero engine, health management, multi-agent system (MAS), communication, collaboration.

1. INTRODUCTION

Aero engine health management, is to detect abnormality or anomaly of temperature, vibration, gas-path, lubrication system, fuel system, drive system through condition monitoring, and diagnose the origin of the abnormality, then predict the engine limited life and where degradation is heading, or what fault or failure will occur and when. Ultimately, maintenance decision or some form of action is taken or integrated into the engine control system. However, the status data of engine components gotten from multi-sensors are massive, just-in-time and heterogeneous. Fault diagnosis and performance prediction must be based on more accurate and complete information.

An multi-agent based architecture for supervision of large-scale chemical plant is proposed. The proposed architecture uses an ontology to represent all the process descriptors formally, so that any changes can be captured and their effects propagated seamlessly [1]. Hidden Markov models utilizing a Gaussian probability distribution are proposed as an anomaly detection tool for gas turbine's component. So hidden Markov models is incorporated into a multi-agent system for the condition monitoring of gas turbine(GT) engines. The multi-agent system is shown to be applicable through a case study and comparison to an existing system [2]. The distributed fault detection and isolation problem for a class of second-order discrete-time multi-agent system is studied by using an optimal robust observer approach [3]. The authors bring forward an existence condition for these observers, which is closely related to the topology structure of the MAS. For the detection of the faults injected in position states, they show that only the position information of the MAS is needed, but for the case of detecting faults in velocity states, both position and velocity measurements are needed

In this article, an aero engine health management system based on multi-agent is designed. By intelligent soft agent approach, multiple functions agents cooperate with each other to solve complex problem which single agent can't accomplish. Meanwhile, the communication and collaboration mechanism for solving resources and tasks conflict of multi-agent are discussed.

2. MULTI-AGENT SYSTEMS

A multi-agent system (MAS) is composed of multiple interacting intelligent agents within an environment. MAS are used to solve problems that are difficult or impossible for an individual agent or monolithic system [4].

2.1. Intelligent Agent

Intelligent agent is a software entity that applies artificial intelligence techniques to choose the best set of actions to achieve a goal specified by the user. There are at least four common characteristics of every agent, including:

(1) Autonomy: There is no global control.

(2) Intelligence: Each agent has incomplete information or capabilities for solving a problem.

(3) Mobility: Agents with different abilities can adaptively organize to solve problem that change temporally.

(4) Collaboration: Agents can cooperate with each other by sharing information in order to solve problems that are beyond the scope of any one agent.

1874-4443/14 2014 Bentham Open

Especially in a distributed computing environment, agents can communicate with each other using an expressive language, work together cooperatively to accomplish complex goats, and use local information and knowledge to manage local resource and handle request from peer agents.

2.2. Key Technology of MAS

Multi-agent systems have been applied to solve a variety of problems. But in any application, there are three common problems to be solved.

(1) Communication and Collaboration

MASs are divided into collaborated MAS and self MAS. The collaborated MAS strives for the best performance of entire system, but the self MAS strives for the best performance of individual agent. The communication of multi-agent is the foundation of multi-agent collaboration.

The typical communication mode includes 'blackboard system' and 'message/dialogue system'. The 'blackboard system' is to achieve broadcast communication with the message shared on the 'blackboard'. However, it is not good at real-time performance and cannot meet the requirement of emergency. In the 'message/dialogue system', the agents exchange information directly each other. The requester not only knows the address of replier but also give the address itself in the message, so the message can return correctly.

(2) Communication Language

Two kinds of communication language were applied widely. One is FIPA's ACL, the other is Knowledge Query and Manipulation Language (KQML).

KQML is concerned primarily with pragmatics, and secondarily with semantics. It is a language and a set of protocols that support computer programs in identifying, connecting with and exchanging information with other programs, including communication layer, message layer and content layer. Compared with KQML, FIPA's ACL defines more flexible and simpler semantic.

2.3. Design of MAS

In MAS, the multi-agent collaborates and accomplishes the mission which is unable by single agent.

Firstly, the task is decomposed. According to the general goal of the system, the complex task is break into several sub-tasks. These sub-tasks should be solved by one or more agents in the system.

Secondly, the roles are defined. According to user's requirement, the agent role is regarded as a function entity responsible to complete one or more objectives, such as detection agent, fusion agent, management agent, interaction agent and negotiation agent.

Finally, the task is assigned. After the agent role defined, the logical relationships between roles, roles and tasks are made out. The task is assigned to one or more agents according to collaborated protocols.

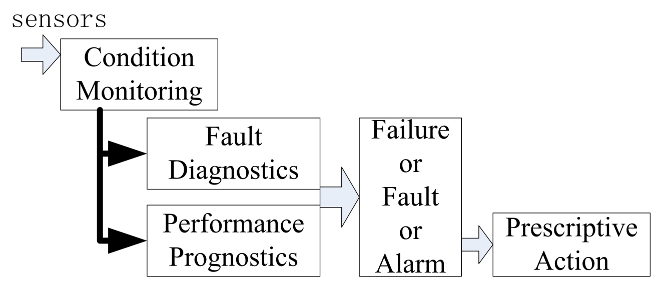

Fig. (1). Function of aero engine health management system.

3. SYSTEM ANALYSIS OF AERO ENGINE HEALTH MANAGEMENT BASED ON MULTI-AGENT

3.1. Management Contents

In 1995, the Department of Defense raised the Engine Monitoring System (EMS) which was designed mainly for fault detection and isolation in the general specification for aircraft engine of turbojet, turbofan and turboprop. In 2004, EMS had developed into the Engine Health Management System (EHMS) which provided the ability of prediction to some extend. In 2007, the system continued developing into the Propulsion and Power Health Management System(PPHMS) which extended the monitor range to other critical flight system, such as the generator and hydraulic pumps. Fig. (**1**) shows the primary functions, including condition monitoring, fault diagnosis, performance prognosis and decision support.

(1) Condition Monitoring

Most of the failures have some signal before they happen. Condition Monitoring is the first step of the EHM process and addresses the awareness of current condition and identification of a symptom or anomaly. This process analyses the data from sensors or periodic inspections, then determines an item's state of degrading or abnormal. Many techniques were applied to determine whether a condition was normal, such as some form of artificial intelligence or a model to analyze data, based on physics, statistic, neural network, Kalman filtering, fuzzy logic, Bayesian belief theory, multidimensional feature analysis.

(2) Fault Diagnostics

Different from Condition Monitoring, this stage must diagnose the cause or origin of the abnormality when an engine is performing abnormally. Health status of the engine can be described as degradation, fault or failure. There is some uncertainty about the current condition. It is likely to involve some fault tracing or fault isolation to determine the underlying cause. Diagnostics requires a knowledge system, much of this knowledge is built into an engine's Failure Modes, Effects and Criticality Analysis (FMECA), which is a powerful diagnostic tool.

(3) Performance Prognostics

Prognostics can be considered as short-term, mid-term (between perhaps two weeks and two months) and

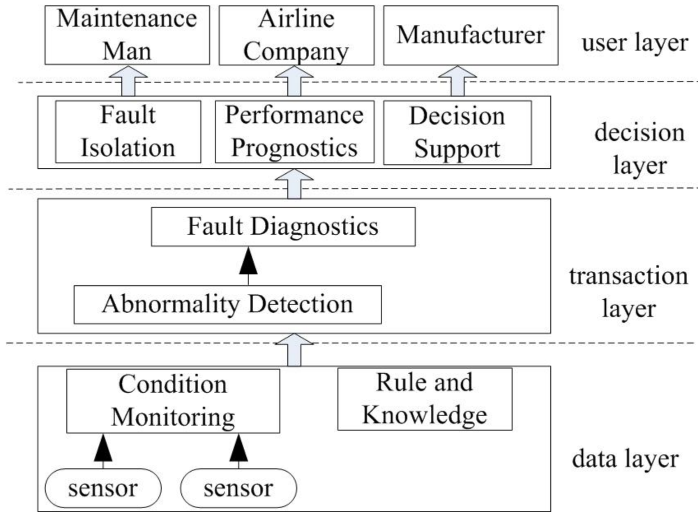

Fig. (2). Architecture of aero engine EHM system.

longer-term. In essence it is about predicting where degradation is heading, or what fault of failure will occur and when, and determining what effects this degradation, fault or failure, or whether it will be critical or impact the aircraft's ability to carry out it's mission.

(4) Prescriptive Action

It's the process to decide what to do about the symptom and then doing it. Similarly to prognostics, prescriptive action can be taken at different levels, the output could be an individual maintenance decision and action to replace a component (or indeed the engine). Ultimately, some form of action could be integrated into the engine control system. Another task is tracking the success of the prescriptive action. It must be determined whether the action(s) taken has eliminated the symptom, and then confirmed the diagnosis or enhanced the prognosis.

3.2. Key Problems

At its core, EHM is essentially an information process, which involves the function of collecting, processing, displaying, evaluating, and disseminating various types of data and information about the past, current, and anticipates health of engines and associated propulsion system elements.

(1) Information Fusion

Due to the status information of engine from multiple sensors, it can improve the accuracy and reliability of fault diagnostics and performance prognostics to extract useful information from various data sources as much as possible. So the rules are important of multi-sources information fusion. Cui Jianguon [5] has used various single classifier and D-S evidence theory to make decision fusion and improve the accuracy of fault diagnosis for key structure members on aircraft.

(2) Prognosis

Compared with Condition Monitoring and Fault Diagnostics, prognosis emphasizes much on quantitative analysis. Its goal is to reduce the dependence on human examination in period. The prognostics methods consist of model-based, knowledge-based and data-based. The typical application is expert system, fuzzy logical and neural network. For the bottleneck of knowledge acquisition and expression in the expert system, Lu Sheng [6] has studied the intelligent fault diagnosis method based knowledge and used semantic knowledge based on ontology to express experience knowledge of expert.

(3) System Integration

EHM depends on massive data from the control system, conversely, the control system adjusts the strategies to eliminate abnormal according to diagnostics report. Therefore, the integration of health management and control systems is inevitable.

In distributed computing environment, intelligent agent technology breaks out the limitations of traditional expert system. Under the supports of communication protocols, the agents cooperate with each other to solve collectively a complex problem that could not be tackles by any agent individually.

4. AERO ENGINE HEALTH MANAGEMENT SYSTEM BASED ON MULTI-AGENT

4.1 Architecture Designing

In general, the architecture of MAS has layered system, implicit calling system and blackboard system. The layered system has good openness and reusability. Aero engine health management includes condition monitoring, fault diagnosis, performance prognostics, and prescriptive action. Thereby, the system architecture was designed as Fig. (**2**).

In the data layer, the data acquisition is accomplished. Most of the data is obtained by distributed sensors and the flight controlling system.

In the transaction layer, analysis of data from sensors, identification abnormality, then the higher reliable fault diagnosis is made, which based on information fusion.

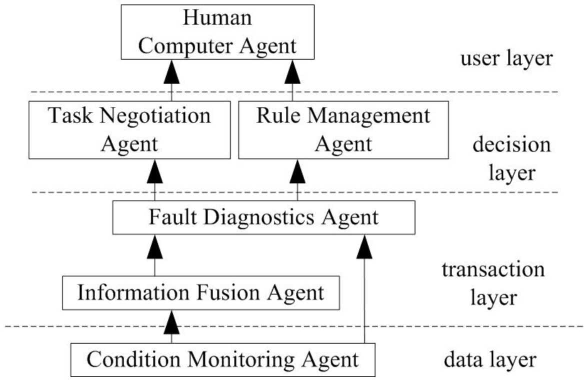

Fig. (3). The function model of six agents.

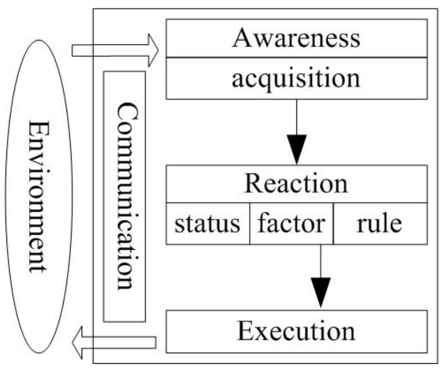

Fig. (4). The logical structure of reactive agent.

In the decision layer, the faults or failures of engine are detected, and the degradation of some component are predicted, then maintenance decision or action to replace a component is made.

In the user layer, it provides the interface for maintenance man, air company management, and manufacturer.

4.2. Agent Roles Definition

According to the function of each layer defined above, there are 6 kinds of agent roles, these are condition monitoring agent, information fusion agent, fault diagnostics agent, task negotiation agent, rule management agent and human-computer interaction agent. The system role function model is as Fig. (**3**).

The condition monitoring agents basically collect and process the datum from the multiple sensors. The information fusion agents extract the effective information of abnormal symptoms according to the intelligent algorithms. The fault diagnostics agents identify the cause or origin of the abnormality. The task negotiation agents coordinate the action of multiple agents according to the task allocation mechanisms. The rule management agent is responsible for the rules of entire system for the intelligent decision. The human-computer interaction agent is responsible for the communication between the system and users.

4.3. Designing of Individual Agent

In this system, as per the event character of engine health management, the agent software structure is reaction type. The reaction soft agent is treated as a reactive system. It is able to sense the various change of environment and itself. And it responds and deals with these changes in time according to the event transaction mechanisms. The logical structure of reactive agent is designed as Fig. (**4**). The major modules are the event acquisition and the response rules.

The awareness module senses the outside environment and the changes inside of agent, and then submits the changes to reaction module as the form of event. The reaction module obtains the rules from the action rule package and submits the execution request to the execution module. Then the execution module implements the action.

5. COMMUNICATION AND COLLABORATION OF MULTI-AGENT

5.1. Based on Event/Message Communication Mechanism

In this system, the abnormality is identified or captured from the condition monitoring agent and information fusion agent. The character of the agent interaction is event -driven and progressive, so the event/message dialog mechanism is adopted, that is to say that individual agent interacts by the receiving and sending messages.

Considering the language structure feature and support environment, FIPA ACL is selected as agent communication language. FIPA ACL language is supported by JADE and fully implemented by Java. The common message structure of FIPA ACL is as below.

```
(inform
:sender //the identity of the sender.
 :receiver // the identity of the intended recipients.
 :reply-to // indicates that subsequent messages to be directed to the agent named in the reply-to parameter.
:content // the object of the action.
 :language //the language in which the content parameter is expressed.
:protocol //the interaction protocol.
 :ontology //the specific encoding of the content language expression.
)
```

5.2. Based on Contract Collaboration Mechanism

The collaboration mechanism based on contract is applied according to the communication model in this system. It depends on the previous agreement of contract whether the agent accepts the messages sent by the other agents during the task is executing.

5.3. Implementation of JADE

JADE provides 3 components: Agent Management System (AMS), Directory Facilitator (DF), Agent Communication Channel(ACC). The major 5 steps are as below for developing multi-agent with JADE.

(1) Creating agents : creating a JADE agent class is the extension of *jade.core.Agent* class, including the methods of *SetUp( )* and *TakeDown( )*. Each agent has identification 'identifier', each 'identifier' is the instance of *jade.core.AID* class.

(2) Defining agent tasks: 'behaviours' defined in JADE represents a task. It is the extension of class *jade.core.behaviours.Behaviour*. Each such behaviour class must implement two abstract methods. The *action( )* define an operation, and the *done( )* return a Boolean value to indicate whether the task has completed or not.

(3) Agent discovery: JADE provides yellow page service. A special agent (Directory Facilitator) becomes dynamic and finds the other agent. The agent can both register services and discover services by DF.

(4) Agent communication: each agent is equipped with an incoming message box and message polling. A message is implemented as an object of the *jade.lang.acl.ACL* Message object, and then calling the *send( )* method of the Agent class.

(5) Defining ontologies: agents must share semantic during communication. Therefore, exchanged message must be expressed by a special language and must share the same ontology. A ontology is the instance of *jade.content.onto. Ontology* class.

CONCLUSIONS AND FUTURE WORK

Previous research has proven that the complicated problem can be solved by intelligent multiple agent methods in highly distributed, heterogeneous, extremely dynamic computational environment. In our case study, we aim to research the solution of the multiple agents in regard of conflict resolution using negotiation. We have designed the multi-agent system of aero engine health management which requires communication and collaboration, and discuss the communication and collaboration mechanism between multiple agents, as well as the implementation methods on JADE platform. Through collaboration and interaction among multiple agents, the multi-sensors information is integrated with flight control system, and intelligent management is implemented, which improves the flexibility and openness of complicated system. But it is not enough to study multi-agent in the face of the multi-issues. The negotiation model, negotiation protocol and negotiation rules of multi-agent need further studies. We will report our progress in our future publication.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work was supported by the Key Subjects Construction Project of Shanghai Education Commission under Grant No.J51403, the Science and Technology Development Foundation of SUES under Grant No.2008xy42, and the Key Subjects Construction Project of Specialty Group of Shanghai under Grant No.B-8932-11-0201.

REFERENCES

[1] S. Natarajan, and R. Srinivasan, "Implementation of multi agents based system for process supervision in large-scale chemical plants", *Computers & Chemical Engineering*, vol. 60, pp.182-196, 2014.

[2] A.D. Kenyon, V.M. Catterson, S.D.J. McArthur and J. Twiddle, "An agent-based implementation of hidden markov models for gas turbine condition monitoring", *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol.44, pp.186-195, 2014.

[3] J. Shi, X. He, Z. Wang and Z. Donghua, "Distributed fault detection for a class of second-order multi-agent systems: an optimal robust observer approach", *IET Control Theory & Applications*, vol.8, pp.1032-1044, 2014.

[4] K. P. Sycara, "Multi-agent systems", *AI Magazine*, vol.19, pp.79-92, 1998.

[5] C. Jianguo, Z. Jie, C. Xicheng, L. Rui and J. Liying, "Application of information fusion in aircraft intelligent health diagnosis", *Journal of Data Acquisition & Processing,* vol. 27, pp. 236-240, 2012.

[6] L. Sheng and L. Tan, "Study on intelligent fault diagnosis method and process based on knowledge," *Machine Tool & Hydraulics*, vol. 37, pp. 167-170, 2009.

Received: November 06, 2014 Revised: December 05, 2014 Accepted: December 13, 2014