An Agent-Based Concept for Problem Management Systems to Enhance Reliability

H. Wang, N. Jazdi, P. Goehner

A defective component in an industrial automation system affects only a limited number of sub functions. However, the affected sub functions often lead to break down of the whole system. The goal of this paper is to propose an intelligent agents-based concept for a Problem Management System (IAPMS) to enhance the reliability of industrial automation systems by resolving not only the known problems but also the unknown problems. In this paper, six types of component agents represent all the necessary components of an industrial automation system and a coordination agent type orchestrates the coordination between the component agents. The component agents detect the defective component by using the existing diagnosis system. On one hand, the component agents and the coordination agent could solve the known problems by using the pre-set measures. On the other hand, since no measures for an unknown problem exist, the corresponding component agents negotiate with the coordination agent to keep the unaffected sub functions alive; hence the system can be further used.

Keywords: reliability, intelligent component agent, Problem Management System, unknown problems

ACM 2012: Multi-agent systems, Cooperation and coordination, Intelligent agents

UDC: 004.02, 004.8, 681.5

0 Introduction

Nowadays, usability, reliability and stability of the industrial automation systems are getting more important. To prevent and to deal with the possible problems, the developer will perform many measures in the industrial automation systems. However, some unknown problems may occur during the operation phase as reported in [1], [2], [3], [4] and [5]. The reasons are concluded as followed:

- Short development time and constraint of budget;
- No systematic reuse of the software;
- Resource constraints in test;
- Defective experience of developer;
- Not completely known operational profile in the development phase;

Meanwhile, there are no existing solutions for the emerging problems, this type of the problems are referred to as **Unknown Problems**. In this case, a component of an industrial automation system can either not properly be performed or cannot be executed at all, as well as its consequent sub functions. This can lead to malfunctions of the whole system. In most cases, a sub function of the system can also meet the users' requirements. This is to say, because of the unknown problems, the reliability of the unaffected sub functions is also declined.

In this paper, we propose an agents-based concept for Problem Management System that attempts to deal with the unknown problems and to perform the unaffected specific sub functions, in order to enhance the reliability of the specific functions. All of the components in an industrial automation system will be represented by intelligent component agents, i.e. user agent, computational unit agent, sensor agent, bus agent and actuator agent. Bus agent type consists of field bus agent and communication agent. It is necessary to have a coordination agent for the Problem Management System, in order to find proper solutions for the unknown problems. Further more, a component agent realizes the problem in its component and attempts to solve it by itself. Afterwards, the component agent consults with the coordination agent on the unknown problems. In the end, they succeed in finding solutions to solve the problems, as well as preforming the unaffected sub functions, thereby improving the reliability of the sub functions.

This paper is organized as following: section 2 introduces the theoretical aspects of an industrial automation system and reliability, and state of the art refers to the approaches of problem management in an industrial automation system. A comparison among the approaches will be performed. In section 3 we will show the architecture of agents-based Problem Management System. In section 4, we present the workflow of problem management in the concept. In section 5 we give some possible applicable scenarios of IAPMS. Finally, conclusions are drawn and prvided (Section 6).

1 Theoretical Aspects and State of the Art

1.1 Reliability in an industrial automation system

An industrial automation system consists of user, computational unit (CU), actuators, sensors [6]. The following part discribes the different components of an industrial automation system:

- *User*: inputs value to the computational unit by empolying the user interface;
- *computational Unit*: achieves the tasks that are assigned by the user. The input value will be processed and set to the acturators;
- *actuator*: receives comands from the computational unit and influences the technical plant;
- *sensor*: measures the physical values of technical plant and converts it into electrical values for the computational unit;
- *bus*: Depending on the size of the system might consist of multi levels. Typically, in a plant automation we have field bus and plant bus. Field bus transports the signals between the computational unit and the actuator, as well as the sensor; Plant bus transports the signals between the user and the

computational unit.

Reliability acoording to IEEE 610.12-1990 [7] is defined as the ability of a system or a component to perform its required functions under stated conditions for a specific period of time. This definition consists of four parts: ability, required functions, conditions, time. The required functions are such functions that are specified by the customer or the developer.

In an industrial automation system, the different components impact the different sub functions and the overall function consists of a number of sub functions. When a component in an industrial automation system is defective and there are no solutions for the occurred problem, the corresponding sub functions will be break down, leading ro disfunctin of the whole system. However, the unaffected sub functions, which can fulfill the requirement of the user, could not be performed because of the loss of the overall function. So we propose a definition of **Reliability of the sub functions**: the ability of an industrial automation system to perform its required sub functions under stated conditions for a specific period of time.

In order to perform the required functions, some measures must be preset to figure out the possible problems in the operation phase. In our paper, the **Unknown Problem**, which can be solved by the agents-based concept of Problem Management System, is defined as followed:

- A defective component could lead to the stop of a whole industrial automation system;
- the cause of the defective component is unknown;
- there are no solutions to solve the occured problem;
- the defective component can be detected by an existing industrial automation system or an existing fault management system [8].

1.2 Problem Management in an industrial automation system

To manage the occurred problems or faults in an industrial automation system, four types of conventional approaches are regrouped in the following four techniques [9]:

Fault prevention: this approach attempts to prevent the occurrence or introduction of faults. Bordasch [8] proposes to prevent a fault using a functional model and a hybrid abnormity identification concept. In this approach, all abnormities are identified and diagnosed. Then the system generates the removal actions and assists the technical staff to solve the problem. This approach aims to sustain the system in a fault-free state. However, in case of an unknown problem, i.e. a problem that has never had occured, no solution can be provided.

Fault removal: this approach tries to reduce the number or severity of faults. The work by [10] proposes a stochastic model that relates the software failure intensity function to development and debugging error occurrence throughout all software life-cycle phases, in order to remove the failures in the development phase. This approach aims to develop a system without faults. Yet since no measures for detekting unknown problems exist, this approach cannot solve the unknown problems in an industrial automation system.

Fault tolerance: this approach attempts to deliver correct service in the presence of faults. Generally, it composes of error detection and subsequent system recovery. For the problem, the error handling is performed. It can take three forms: rollback, roll forward, redundancy. Redundancy is usually used for fault tolerance in an industrial automation system [11]. Fault tolerance aims at the occurred errors or problems, but in the practical field, it's almost impossible to fulfill the redundancies of all the components in an industrial automation system.

Fault forecasting: this approache attempts to estimate the present number, the future incidence and the likely consequences of faults. This approach can forecast the possible faults. But when there are no corresponding measures, the problem will not be solved.

As it can be concluded, it is impossible to solve the unknown problems by the conventional approaches in an industrial automation system. Modern research activities in the field of intelligent agent systems has opened new horizon for managing the faults in an industrial automation system. In [12], Merdan presents an automation agent approach with agents comprising a software component with an integrated world model repository besides the related hardware. World model is used for the representation of the external surroundings and internals of the agent. In [13], Cerrada proposes a reference model for fault management in industrial processes. A set of models describe the general characteristics of the agents, specific tasks, communications and coordination. In the fault management system, the actions are related to the decision-making in the scheduling of the preventive maintenance task and the running of preventive and corrective specific maintenance tasks. But there are no approaches to perform the unaffected funcitons. Consequently, a new concept for the Problem Management System is proposed in this paper. With the help of the intelligent agents, the Problem Management System provides a possibility to analyze the occurred problems, to search the unaffected sub functions, and to make a new decision by itself. Furthermore the Problem Management System can help an industrial automation system to perform the unaffected sub functions as well as to maintain the reliability of the sub functions.

2 Architecture of Agents-based Problem Management System

In the last section, we have reviewed some conventional approaches without agents and some approaches with agents to solve the problems in an industrial automation system. In this section, we propose a new architecture of agents-based Problem Management System to enhance the reliability of the sub functions (IAPMS). Hence, we use different component agents to represent different types of the system components by using the abstraction of knowledge, information and data of the components in an industrial automation system.

2.1 component agents

In IAPMS, the component agents represent all components in an industrial automation system. They consist of 7 types of agents (see figure 1):

- Sensor agent: It represents all the sensors in an industrial automation system. It knows the defective sensors and the affected sub functions. Furthermore, the sensor agent can perform the pre-set solutions or negotiate with the coordination agent.
- Actuator agent: It represents all the actuators in an industrial automation system. It knows all the relationships among the actuators and the affected sub functions. With a defective actuator, the actuator agent can perform the pre-set solutions or negotiate with the coordination agent;
- Computational unit agent (CU agent): It represents all the computational units in an industrial automation system. It knows the processes or domains of the computational unit and the corresponding sub functions as well as the sub-systems;
- User agent: It represents the user of an industrial automation system. It knows the task and the required parameters of the user and it is in the charge of communication between the user and the coordination agent. Moreover, the user agent could provide an interface for the user [14];
- Field bus agent: It represents the field bus between the computational unit and actuators as well as sensors;
- Communication agent: It represents the communication between the user and the computational unit;
- Coordination agent: a coordination agent is needed to find a solution for the problems by coordinating the component agents. Coordination agent coordinates the agents community [13] and deals with the unknown problems in an industrial automation system. It negotiates with the component agents to make a new decision to deal with the defective component and to perform the unaffected sub functions;

2.2 The Architecture of IAPMS

IAPMS includes six types of component agents and a coordination agent (see figure 2). The intelligent component agents represent different types of system components in an industrial automation system. The occurred problem of a component will be firstly handled by the corresponding component agent. The coordination agent can communicate with the corresponding component agent and make a decision to handle the unknown problems. The user agent is a special component agent. It represents the interface between a user and an industrial automation system and deals with the possible problems in the interface. Moreover, the user agent provides

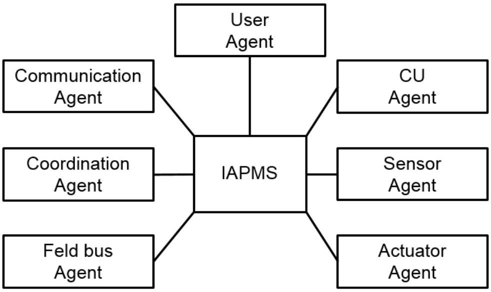

Figure 1. Six types of component agents and a coordination agent in IAPMS

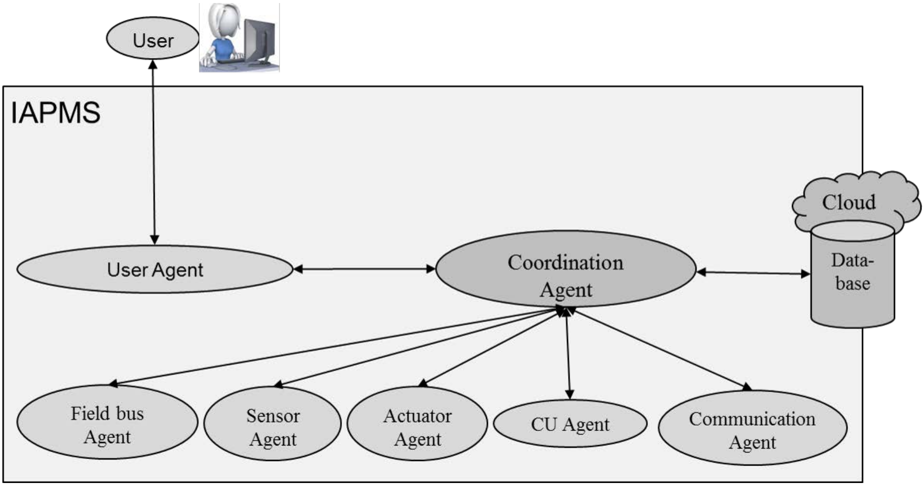

Figure 2. Architecture of IAPMS

an interface for the user in the IAPMS. The user can decide whether to perform the decision by the coordination agent or not. The database stores a lot of decision for the different problems. With the help of the cloud, the coordination agent can remote access through the internet the resources in the server. It is necessary to have the required information on the server. With this the IAPMS can get further information, e.g. similar problems within other industrial automation systems, and employ them for better decisions.

3 Workflow of IAPMS

In this section, the workflow of IAPMS will be illustrated in figure 3. The workflow of the problem management consists of four parts: realizing the problem, dealing with the local problems, dealing with global problems, and communicating with database.

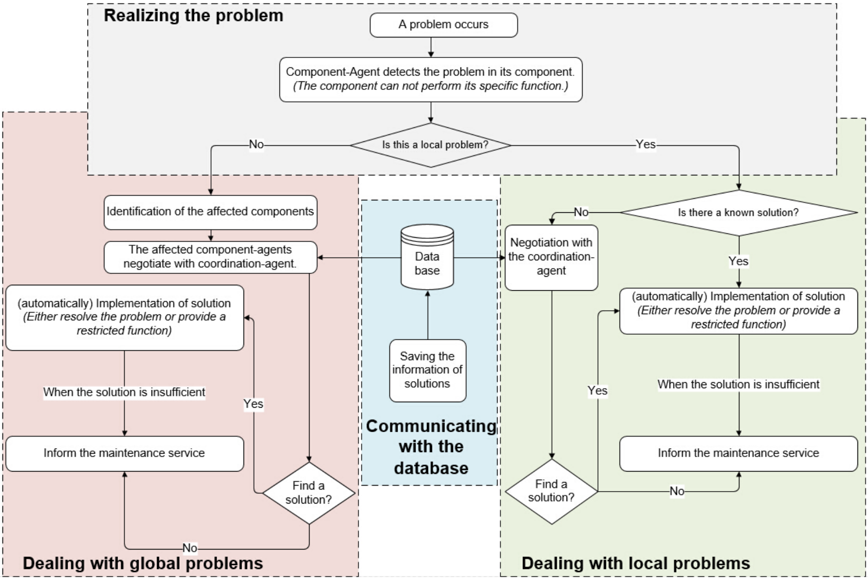

Figure 3. Workflow of IAPMS

Realizing the problem:

Firstly, when a problem occurs, the component agent can realize via the existing diagnosis system that there is a problem in its represented component, i.e. the corresponding sub function cannot be fully performed. The component agent identifies if the problem has occured only in its component, i.e. whether the problem is local or global. A local problem means that the problem affects only one component; a glocal problem means the problem affects more than one component.

Dealing with local problems:

If the occurred problem is local, the component agent searches its knowledge whether there is a known solution to the problem. If there is a solution, the IAPMS implements the solution (either solves the problem or provides a restricted functionality). If the solution is insufficient, IAPMS informs the maintenance service; If there is no known solution, the component agent negotiates with the coordination agent to find a solution to handle the problem (e.g. perform the not affected sub functions, which can fulfill the requirement of the user). If after the negotiation, the coordination agent cannot find a solution, then the IAPMS will again inform the maintenance service.

Dealing with global problems:

If the occurred problem is global. The component agent should identify all the affected components. All the affected component agents negotiate with the coordination agent to find a solution. In addition, the coordination agent can remote access the resource on the server via internet. If a solution is found, then the IAPMS should implement the solution (e.g. perform the unaffected sub functions, which can fulfill the requirements of the user or the customer). If after the negotiation, the coordination agent cannot find a solution or the solution is insufficient, then the IAPMS will inform the maintenance service.

Communicating with the database:

When the coordination agent is needed to solve the problem, the coordination agent inquiries the database, which stores the solutions for the known problems and all the sub functions of an industrial automation system. After the negotiation among the coordination agent and the component agents, the solution should be stored in the datebase by the coordination agent. Hence, the solution can be used for the next time.

4 Possible Applicable Scenarios of IAPMS

For demonstrating the possible applicable scenarios, the applications of IAPMS will be introduced in this section.

4.1 Scenario 1: Industrial Coffee Machine

Problem: In an industrial coffee machine, after several dispensing of coffee, the drip tray of the coffee machine is full (see figure 4). Then the coffee machine will stop working unless someone empties the drip tray. Because of the full drip tray, the sub function "hot water" is also out of order. However, in the real operation phase, someone may just need the sub function "hot water". While this problem is not considered in the development phase, the coffee machine has no solution to solve this unknown problem in the operation phase and to fulfill the requirement of the users.

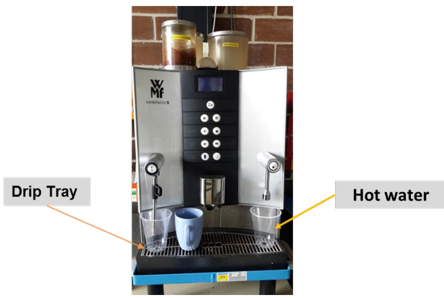

Figure 4. IAS Industrial Coffee Machine

Solution: On this occasion, all the components in the industrial coffee machine are represented by component agents, and the sensor agent can detect and analyze the problem. By the inquiry of the database and the reasoning of all the corresponding component agents, a new solution can be reasoned by the coordination agent to perform the sub function hot water.

4.2 Scenario 2: High-bay Warehouse

Problem: Here we consider a model of a high-bay warehouse at the Institute of Industrial Automation and Software Engineering. It consists of three sub functions, input the work pieces from the transporter, store the work pieces in a slot, output the work pieces to the users. Figure 5 shows a picture of the high-bay warehouse at IAS. Because of the requirement of the users, typically, the direction from input to output is fixed. However, when the motor of input is broken, the high-bay warehouse must stop to wait for the maintenance service, until the motor has been repaired. In this case, the sub function of storing and output is broken down.

Figure 5. IAS High-bay Warehouse

Solution: On this occasion, all the components in the high-bay warehouse are represented by the component agents. The broken motor can be tested by the existing industriail automation system. Then the actuator agent realizes the problem. Because there are no existing solutions in the actuator agent and its knowledge, the actuator agent must communicate and negotiate with the coordination agent to find a solution as well as to perform the sub function of storing the work pieces and outputting the work pieces.

5 Conclusion and Future Work

In this work, we reported an intelligent agents-based approach for Problem Management System (IAPMS) to enhance the reliability of specific functions . This approach improves not only the reliability of the whole system but also improves the reliability of the sub functions of an industrial automation system. Meanwhile, a nevol way is proposed to deal with the occurred unknown problems by performing the unaffected sub functions that might still fulfill the requirement of the user.

For the relization of the concept, all the components in an industrial automation system will be represented by the component agents. In the operation phase, a defective component affects only a limited number of sub functions, the affected sub functions result in the stop of the whole industrail automation system. The component agent realizes the defective component through the existing diagnosis system. The component agents and the coordination agent solve the local problems by pre-set measures. Afterwards, since no solutions for a problem, the correspondring component agents negotiate with the coordination agent to keep the unaffected sub functions alive, which can still fulfill the requirement of the user. In addition, the coordination agent can access the resources on the server via Internet to find a possible solution, assuming that the nessary information are available.

As a future work, we are implementing the concept of IAPMS in a practical project, on an industrial coffee machine and a high-bay warehouse at IAS. Evalutaion of the results based on the prototpy will follow.

Acknowledgement

We thank Chinese CSC (China Scholarship Council fellowship Grant) for financial support.

References

[1] R. Baskerville and J. Pries-Heje, "Short cycle time systems development," *Information Systems Journal*, vol. 14, no. 3, pp. 237–264, 2004.

[2] M. Luguna and et al., "Introducing systematic reuse in mainstream software process," in *Proceedings of the "Euromicro Conference"*, pp. 351–358, IEEE, 2003.

[3] J. Kim and A. Porter, "A history-based test prioritization technique for regression testing in resource constrained environments," *Software Engineering, 2002. ICSE 2002. Proceedings of the 24rd International Conference on*, pp. 119–129, 2002.

[4] A. Mockus and D. Weiss, "Predicting risk of software changes," *Bell Labs Technical Journal*, vol. 5, no. 2, pp. 169–180, 2000.

[5] J. Musa, *Operational profiles in software-reliability engineering*, vol. 10. IEEE, 1993.

[6] K. L. S. Sharma, *Overview of industrial process automation.* Elsevier, 2011.

[7] Institute of Electrical and Electronics Engineers, *IEEE Std 610.12-1990, IEEE Standard Glossary of Software engineering Terminology*, 1999.

[8] M. Bordasch and P. Goehner, "Fault prevention in industrial automation systems by approachs of a functional model and a hybrid abnormity identification concept," in *Proceedings of the "IECON 2013-39th Annual Conference of the IEEE"*, pp. 2845–2840, IEEE, 2013.

[9] B. Lussier and et al., "On fault tolerance and robustness in autonomous systems," in *Proceedings of the "3rd IARP-IEEE/RAS-EURON Joint Workshop on*

Technical Challenges for Dependable Robots in Human Environments", pp. 351–358, laas.fr, 2004.

[10] M. Stutzke and C. Smidts, "A stochastic model of fault introduction and removal during software development," *Reliability, IEEE Transactions on*, vol. 50, no. 2, pp. 184–193, 2001.

[11] M. Kauer and et al., "Fault-tolerant control synthesis and verification of distributed embedded systems," in *Proceedings of the "Conference on Design, Automation and Test in Europe "*, pp. 1–6, European Design and Automation Association, 2014.

[12] M. Merdan and et al., "Monitoring and diagnostics of industrial systems using automation agents," *International Journal of Production Research*, vol. 49, no. 5, pp. 1497–1509, 2011.

[13] M. Cerrada and et al., "Agents-based design for fault management systems in industrial processes," *Computers in Industry*, vol. 58, no. 4, pp. 313–328, 2007.

[14] H. Mubarak and P. Goehner, "An agent-oriented approach for self-management of industrial automation systems," in *Proceedings of the "Industrial Informatics (INDIN), 2010 8th IEEE International Conference on"*, pp. 721 – 726, IEEE, 2010.

Authors

Huiqiang Wang — The 2nd year PhD student, Institute of Industrial Automation and Software Engineering, University of Stuttgart, Stuttgart, Germany; E-mail: *huiqiang.wang@ias.uni-stuttgart.de*

Nasser Jazdi — Scientific staff member, Institute of Industrial Automation and Software Engineering, University of Stuttgart, Stuttgart, Germany; E-mail: *nasser.jazdi@ias.uni-stuttgart.de*

Peter Goehner — Director of the Institute of Industrial Automation and Software Engineering, University of Stuttgart, Stuttgart, Germany; E-mail: *peter.goehner@ias.uni-stuttgart.de*