PIRAA: DEVELOPMENT OF AN INJECTION MOLDING MACHINE MANUFACTURING MONITORING SYSTEM BASED ON PRODUCTION INFORMATION RETRIEVAL AGENT ARCHITECTURE

Shang-Liang Chen, You-Chen Lin, Yun-Yao Chen and Sin-Ru Wang
Institute of Manufacturing Information and Systems, National Cheng Kung University, Tainan, R.O.C.
E-mail: yewyewchen@gmail.com

ICETI-2014, AA1009_SCI
No. 15-CSME-27, E.I.C. Accession 3802

ABSTRACT

Production information retrieval is an important issue for manufacturing industries. Effective data transmission and retrieval mechanism with data protection is an urgent research topic. Therefore, a socket information retrieval mechanism and agent services for information retrieval on the basis of injection molding machines is proposed in this research and is addressed as Production Information Retrieval Agent Architecture (PIRAA). The proposed agent adapted a customized packet format for client-server connection communication and sent process parameters back to a production process database. Using the experimental results, a cloud-based platform was implemented based on injection molding machines. The results show the PIRAA framework can be used for developing cloud-based platforms in a stabilized and remarkable way.

Keywords: PIRAA; cloud computing; cloud-based injection molding machine manufacturing monitoring system.

PIRAA: DÉVELOPPEMENT D’UN SYSTÈME DE SURVEILLANCE DE PRODUCTION D’UNE MACHINE DE FABRICATION DE MOULAGE PAR INJECTION BASÉ SUR UNE ARCHITECTURE D’EXTRACTION DE L’INFORMATION

RÉSUMÉ

La recherche d’information sur la production est un enjeu important pour les industries manufacturières. La transmission de données efficace et un mécanisme d’extraction et de protection des données est un sujet de recherche pressant. Par conséquent, un mécanisme de recherche d’information et des agents de services pour l’extraction de l’information fondé sur les machines à moulage par injection est proposé *Production Information Retrieval Agent Architecture* (PIRAA). L’agent proposé est adapté d’un format de programme personnalisé via une connexion client-serveur et renvoie les paramètres du processus à une base de données de production. Utilisant les résultats expérimentaux, une plateforme basée sur *cloud* a été mise en service sur des machines de moulage par injection. Les résultats démontrent que le cadre PIRAA peut être utilisé pour développer des plateformes basées sur cloud de manière stable et remarquable.

Mots-clés : PIRAA; système de surveillance d’une machine de fabrication de moulage par injection basé sur *cloud*.

1. INTRODUCTION

In recent years, 3C product casing replacement has become relatively fast, and injection molding machines can be used for rapid mass casing production. Besides 3C technology commodities, most necessities in daily life come from injection molding machines. Such wide application makes the monitoring of production information and communicating and acquiring machine information through the use of PCs a red-hot research topic. Industries often adapt TCP/IP protocol to communicate with machines such as the injection molding machine giants FANUC of Japan and ARBURG of Germany. In this study, a machine information retrieving agent was built, integrated with controllers, and an injection molding machine data center was constructed to provide access to an injection molding machine database for other monitoring systems and for the purpose of checking the machine status in any networking environment. The production process data for the injection molding machine can be accessed by the agent services proposed in any networking environment through agent services in order to construct a variety of systems for machine monitoring and information analysis. If an error or exception occurs, remote machine diagnosis can be executed at once. Chen et al. proposed a novel framework for an agent-based production remote monitoring system design with a case study of injection molding machines. The proposed framework has the following advantages [1–3]. (1) Machine information can be obtained across platform transmissions. (2) It actively pushes information to the manager without a machine operator. (3) The categorization of required information, filtering of extra information, and elicitation of the data can be carried out according to user needs. To further implement the concept of the previous frame-work, in this study, an agent for production information retrieval services was established. In a multi-plant structure, all injection molding machine controller information will be compiled and sent back to the injection molding machine production database through the agent in order to reduce resource waste related to duplicated information. The proposed architecture combined controller and PC-based architecture to achieve non-stop operation with no missing information and high security data center services. The remote linking control mechanism was divided into client (PC-based) and server (controller), and TCP/IP protocol was employed as the data transmission protocol for both the client and server. The following research topics were investigated and proposed:

1. The proposed agent socket packet linking mechanism.
2. A customized transmission packet format and its testing.
3. The design for machine information retrieving agent system architecture.
4. The proposed injection molding machine agent information retrieval services.
5. A plan for an injection molding machine production process database.
6. The provision of an injection molding machine production database agent service.
7. The construction of information security architecture.
8. The development of a cloud-based platform based on injection molding machines along with experimental results.

2. LITERATURE REVIEW

Some studies related to intelligent machine agent architecture designed for the purpose of supporting manufacturing processes have been proposed in recent decades. IA (Intelligent machine agent) architecture with a multi-agent framework consists of a number of autonomous agents (such as a service agent, a fault diagnosis agent, a manufacturing agent and a corporate memory agent) has been discussed. Agent-based approaches for dynamic scheduling in content-based networks for intelligent manufacturing in a virtual work-shop environment have been formulated based on multi-agent architecture. The following studies of agent-based manufacturing systems are listed and discussed:

No.	Authors	Research description
1	Grant H. Kruger et al. [1]	This paper showed that in the case of the IA (Intelligent machine agent) architecture for autonomous process optimization, a great deal of experimental human trial-and-error and uncertainty can be avoided as a result of process optimization, acceleration of the process ramp-up and regulation and reduction of associated production costs.
2	Kumar, V. et al. [2]	This paper proposed a multi-agent framework consisting of a number of autonomous agents, such as a service agent, a fault diagnosis agent, a manufacturing agent and a corporate memory agent. The service agent and fault diagnosis first categorizes the returned warranty failure products and then conducts analyses of the root cause of the failures. This information is then passed on to the manufacturing agent to overcome the problems related to failure.
3	Xiao-yan Zhang et al. [3]	This paper proposed a framework for autonomous collaborative manufacturing development, in which a set of intelligent agents handle major interactive tasks based on ontology. A communication process between agents is provided, which identifies distributed participators and enables the sharing of heterogeneous data via agent messages.
4	Moser, T. et al. [4]	This paper proposed a pattern-based MAST-SAW (Manufacturing Agents Simulation Tool – Simulation of Assembly Workshops) multi-agent production automation simulation tool. Its usefulness was for the simulation of complex and distributed production processes using a layered coordination framework.
5	Rizvan Erol et al. [5]	This paper proposed a multi-agent system for dynamic and simultaneous scheduling of AGVs (Automated guided vehicles) and machines in manufacturing systems. The proposed multi-agent system consists of four Machine Agents (Scheduler and Resource agents) and two AGV Agents (Scheduler and Resource agents).
6	Aburukba, R. et al. [6]	This paper presented an agent-based approach for dynamic scheduling in a content-based network. It considered the dynamic behavior of the system, such as resource additions or breakdowns, and task additions or removal.
7	Qing-lin Guo et al. [7]	This paper proposed an effective method for intelligent manufacturing in a virtual work-shop environment was formulated based on multi-agents. Intelligent manufacturing architecture based on multi-agents was proposed to resolve conflicts and respond to unforeseen events.
8	Qi Hao et al. [8]	This paper proposed a new cooperative framework that covers three levels of manufacturing management, i.e. virtual enterprise (inter-enterprise), enterprise (intra enterprise), and shop floor. Multi-agent and Web services were the main technologies adopted and developed in this framework.
9	Qinglin Guo et al. [9]	This paper proposed architecture consisting of various autonomous agents that are capable of communicating with each other and making decisions based on the methodology of multi-agent systems (MAS) in distributed artificial intelligence (DAI). The Agent-based manufacturing system modeling and processing model have the following features as compared with the general Petri net model and the processing entity agent: autonomy, response capacity and activity.
10	N. He et al. [10]	This paper presented a hierarchical agent bidding mechanism, an agent-based control method that is particularly designed for production planning and scheduling in Make-to-Order manufacturing systems and was an attempt to enhance the operational flexibility of manufacturing systems in order for them to deal with dynamic changes in regard to business environment additions or removals.
11	Wei Jiang et al. [11]	This paper put forward the cloud agent concept and model. Based on a cloud agent, a cloud manufacturing integration service mode is present, which provided effective integration services for manufacturing resources under a network environment.

No.	Authors	Research description
12	Hongbin Zhang et al. [12]	On the Internet, increasing Web information leads users to get lost in the "ocean of information", so this paper proposed a multi-agent information retrieval system based on intelligent evolution (MAIRSIE) by making use of multi-agent technology, combining the history query information of users with current retrieval keywords, and by mining users' potential interests and characteristics.
13	Yazen Alsafi et al. [13]	This paper proposed a novel approach to achieving rapid reconfiguration of modular manufacturing systems based on an ontology-based reconfiguration agent. The agent uses ontological knowledge of the manufacturing environment for the purpose of reconfiguring without human intervention.
14	Yingfeng Zhang et al. [14]	This paper proposed and defined the concept of smart objects and designed an agent model called agent-based smart objects (ASO) in a manufacturing environment. Following service-oriented architecture, a Web services framework for a smart object management system (SOMS) was established.
15	Luis Morales-Velazquez et al. [15]	This paper proposed an open-architecture platform based on multi-agent hardware-software units by developing a novel Multi-Agent Distributed CONtroller (MADCON) system. The MADCON system was developed to simplify the interaction between hardware and software functions in industrial applications, making the system customization simple and spending less time on development and maintenance.
16	Jianxin Jiao et al. [16]	This paper applied the multi-agent system paradigm to collaborative negotiation in a global manufacturing supply chain. Multi-agent computational environments were suitable for studying a broad class of coordination and negotiation issues involving multiple autonomous or semiautonomous problem solving agents. A case study of mobile phone global manufacturing supply chain coordination was also reported.
17	Minarolli, D. et al. [17]	This paper proposed an approach to support the Virtual Machine Monitor in performing resource allocation of VMs running on a physical machine of a cloud provider by expressing the two objectives in a utility function and optimizing this function using fuzzy control. To potentially work for an increased number of virtual machines, a multi-agent fuzzy controller is realized where each agent optimizes its own local utility function.
18	Li-Man Liao et al. [18]	This paper constructed an agent-based system for multi-objective parallel machines scheduling according to machine preferences. The proposed framework consisted of three types of agents, i.e., job agent, machine agent and management agent. Agents could negotiate effectively based on the designed negotiation protocol.
19	Ling, S. H. et al. [19]	This paper presented a Decentralized Multi-Agent System (DMAS) by using the proposed algorithms of Intra- and Inter-Roam-Space movements and a negotiation protocol between agents to solve dynamic resource allocation problems, called Machine Degradation (MD).

3. PRODUCTION INFORMATION RETRIEVAL AGENT ARCHITECTURE

For injection molding machine data retrieval and the reporting of equipment manufacturer information, we integrated real injection molding machine controllers and information retrieving agent architecture to build an injection molding machine database. The production information retrieval agent architecture, shown in Fig. 1, is divided into an injection molding machine, an agent, a service, and a database, which originated from our previous research [20–22]. The proposed architecture is divided into three main modules, including a Production and Process Module (PPM), a Production Information Retrieval Module (PIRM) and an Application Module (AM). The PPM is responsible for production and manufacturing, which means the injection molding machines and production process scenario. The PIRM provides production information retrieval services on the data center and servers. Firewall, encryption mechanism and information retrieval agents (IRA) are ported, which can be represented as a data layer. AM represents the application accessed by end users. In this module, different manufacturing monitoring systems, manufacturing enterprise systems

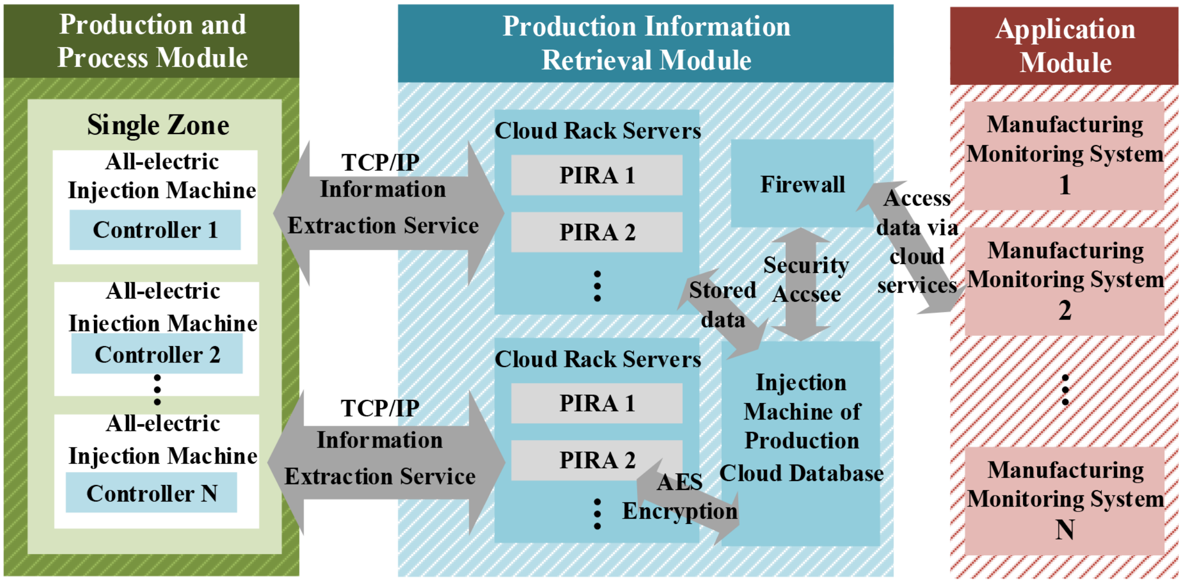

Fig. 1. Production information retrieval agent architecture.

or reports can be developed based on the production data retrieved by the production information retrieval agents. Detailed descriptions of each module are provided as follows, as shown in Fig. 1:

- *All-electric injection molding machine*: An all-electric injection molding machine is used as our case study for developing the manufacturing monitoring system. By receiving the packet transferred by the controller on the injection molding machine as based on the proposed PIRAA framework, the data can be saved in the cloud database safely and in a stable manner.

- *Production Information Retrieval Agent*: The information retrieving agent was deployed and installed on the server. It sent/received packets to/from the controller through sockets. It retrieved production in-formation and saved the obtained data into the database. The socket packet linking mechanism and algorithm are addressed in Sections 4 and 5. The mechanism enabled the agents to receive the packets in a stable manner.

- *Cloud Rack Servers*: The cloud rack server is the data retrieval service, which ensures that the user has permission to access the service through an authentication mechanism. After login, various types of monitoring software can conduct machine condition inspection or data analysis and diagnosis through this service. All the cloud rack servers are clustered as virtualized machines. The resources handshaking mechanism is handled by a virtual machine management platform.

- *Production Database*: The database for the machine, maintenance, user permission, machine parameters, machine temperature data, etc. is only accessed by an authenticated user. Molding parameters, such as molding temperature and molding pressure are saved in our databases. The data can be accessed through cloud services provided in the AM. The data encryption mechanism is used to enhance the data security of IRAs.

- *Manufacturing Monitoring System*: By using the platform, molding machine managers and engineers are able to create or delete IRAs. The resource management and data acquisition functions were implemented in this module. The system interface was built on a web server with firewall protection within gateways.

Table 1. Header data format and length.

Format	Length (Bytes)
msgType	1
msgFlag	1
dataLegth	2
Index	2
BlkName	2
DataType	1
Others	11

Table 2. MsgType packet format.

MsgType	Function
Error	It transmits network error messages.
None	Reserved.
MachineStatus	It monitors the machine status.
ParaRead	It monitors the parameters (RT, RM, RD).
SPC	It monitors SPC.
AlarmRecord	It monitors the alarm records.
HistoryRecord	It monitors the operation records.

Table 3. msgFlag packet menu.

MsgFlag	Number
START	0
CONTINUE	1
END	2
ENABLE	3
DISABLE	4

4. AGENT SOCKET PACKET

A Microsoft.Net Framework socket for the purpose of building a linking mechanism is adapted for implementation in this study. The information retrieving agent and injection molding machine controller can con-duct packet transmission and communication via a TCP/IP interface, and the agent can retrieve the machine status and production information. For faster and more stable transmission, the packet size in this study was specified as 276 bytes. The socket packet data format contained a 20-byte header and 256-byte data. The header and data are explained as follows: The packet format is divided into two parts, including the header format and data format. The function of header is the format type and features of the packet, so the program can make the right response to the packet it receives. Data is the content to be transmitted. The header content includes msgType, msgFlag, dataLegth, index, BlkName, and DataType, and the data format and header data length is described in detail on Table 1 as follows. MsgType represents the packet function, as shown in Table 2, msgFlag indicates msgType status, which can also be described as a detailed setting of MsgType, namely, START, CONTINUE, END, ENABLE, DISABLE, and OTHER. They serve as commands, as shown in Table 3, dataLegth represents data length, index represents the address that stores the parameters, BlkName represents the bank that stores the parameters, DataType represents data types. The types are divided into BYTE, USHORT, INT, STRING, and OTHER for transformation basis of each command, as shown in Table 4.

Table 4. DataType packet menu.

DataType	Number
BYTE	0
USHORT	1
int	2
string	3
other	4

Table 5. Packet header format from PC and controller.

Algorithm: Agent Socket Packet Linking Mechanism

```
//Connect to server which is listening
IPAddress IP = IPAddress.Parse(IpAddress);
IPEndPoint EndPoint = new IPEndPoint(IP, Port);
Socket clientSocket = new Socket(AddressFamily.InterNetwork, //Initialize the client socket
            SocketType.Stream, ProtocolType.Tcp);
clientSocket.BeginConnect(EndPoint); //Begin to connect the target server
//Initialize the packet headerInfo then create a headerMsg
MessageManager.HeaderInfo headerInfo = new MessageManager.HeaderInfo();
headerInfo.setMsgHeader(MsgType, MsgFlag, MsgDataType, blkName,
startIndex, dataLength);
byte[] headerMsg = MessageManager.CreateHeaderMsgBuffer(headerInfo);
//Begin to send the packet to target server
if clientSocket.IsConnected then clientSocket.Send(headerMsg);
//Initialize the dataBuffer dateType as same as MsgData sent from server
byte[] dataBuffer = new byte[20 + 1024];
//Receive the MsgData from server
clientSocket.BeginReceive(dataBuffer, dataBuffer.offset, dataBuffer.Length);
//Close socket connection if received MsgData from server
if clientSocket.IsReceived then clientSocket.CloseConnection();
```

Table 6. Header format of packet from PC and controller.

Packet Category	PC	Controller
msgType	ParaRead	MachineStatus
msgFlag	ENABLE	ENABLE
dataLegth	dataLength	dataLength
Index	startIndex	startIndex
BlkName	blkName	blkName
DataType	INT	INT
Data	None	ParaRead

5. AGENT SOCKET PACKET LINKING ALGORITHM

The agent socket packet linking mechanism algorithm is shown in Table 5. The algorithm describes the server and client connections. The client and server sockets are the linking of the injection molding machine data transfer. The data transactions are divided into packets through either the intranet or the Internet. An agent socket packet consists of a header message and a data buffer. The class socket is responsible for initializing new methods of parsing the message header and buffer. The class MessageManager is responsible for creating new methods to initialize the packet header information to create a header message.

Table 7. Packet category flag PC receives from controller.

Packet Category	Controller flag
msgType	4
msgFlag	3
dataLegth	1
Index	247
BlkName	18

Table 8. Packet category flag PC sends to controller.

Packet Category	PC flag
msgType	4
msgFlag	3
dataLegth	1
Index	247
BlkName	18

Table 9. Test server simulation environment specifications for the information retrieval agent.

Test Items	Specification
OS	Windows Server 2012
Programming language	MS Visual Studio 2012 C#
Database	MS SQL Server 2012
Web server	Internet Information Server(IIS)8.0
RAM	8G
CPU	Intel(R) Xeon(R) Processor E5-2620/2.00GHz (2 Core)

Take the injection molding machine parameters as an instance. The PC sends an inquiring injection molding machine status packet. The controller receives the packet and returns the injection molding machine status packet. The packet header format from the PC and controller is shown in Table 6. The packet category of both the PC and controller are mapped for data transmission. The packet category flag sent from the PC to the controller is shown as Table 7. The packet category flag that the PC receives from the controller is shown in Table 8.

6. EXPERIMENTAL RESULTS

6.1. Experiment Test Environment Settings

When the injection molding machine started to process, large scale data packets poured into the database and was retrieved by our retrieval agents. The information retrieval agent was installed on the data collection server. In order to continuously retrieve the packets without missing data and errors, we further designed an intelligent agent socket packet linking mechanism, as discussed in Section 5.2. The test server simulation environment specifications for the information retrieval agent are listed in Table 9.

The system was implemented with MS visual studio 2012 C# programming language. All the production information was saved into an MS SQL server 2012. An Intel E5-2620 CPU with 8G RAM was adapted to simulate the agent retrieval environment. The injection molding machine manufacturing monitoring system interface is shown in Fig. 2. Each agent has its own agent ID. The injection molding machine manufacturing monitoring system is able to manage multi-agents by creating new agents through a specific generator button called “Create exe”. A delete function is also provided to purge created agents by clicking button “Delete exe”.

Fig. 2. Injection molding machine manufacturing monitoring system interface.

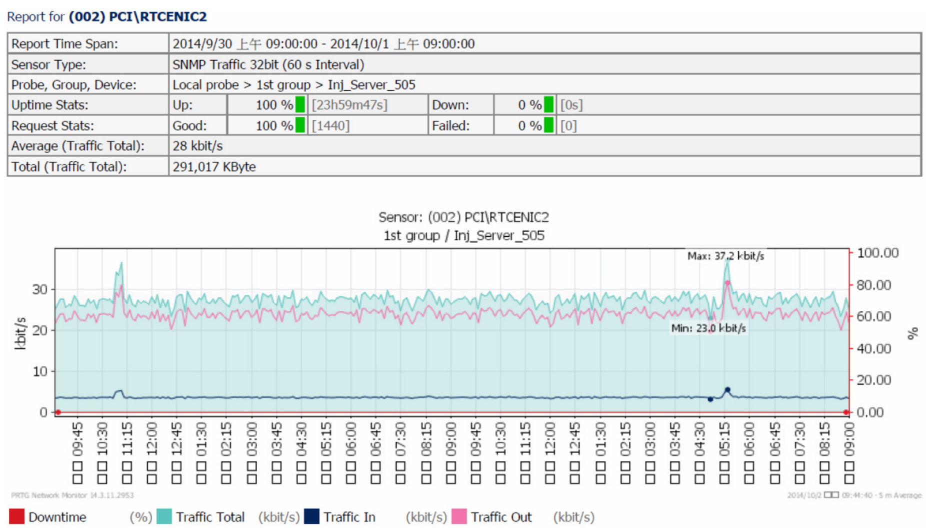

Fig. 3. Test results for the injection molding machine information retrieval agent under continuous execution for 24 hours for a single injection molding machine.

6.2. Experimental Results and Discussion

Figure 3 depicts the test results for the injection molding machine information retrieval agent in the case of continuous execution of a single injection molding machine for 24 hours. The infrastructure of the Ethernet transmission is based on a 10/100M bps transmission bandwidth. As shown in the test results, the data transmission was 100% in the 24 hour interval with no data missing. It can be seen that the average network flow traffic transferred by this single injection molding machine was 28 kbits/s. The total traffic was 291,017 Kbyte in 24 hours, which shows that the implemented production information retrieval agent executed well. As shown in the figure, there was a peak of transmission flow at 05 : 15, which described the data purge and warehousing of the injection molding machine controllers. The results indicate that the maximum transmission flow can be 37.2 Kbits/s at approximately 23.0 Kbits/s. In conclusion, the testing results showed that the proposed injection molding machine information retrieval agent can work well with the designed data encryption mechanism and that the transferred production information is stable.

7. CONCLUSIONS

The experimental results of this study proved that PIRAA is workable for the development of an injection molding machine manufacturing monitoring system through the prescribed socket linking mechanism and that the developed information retrieving agent can successfully communicate with the machine controller. The conclusion of this study is as follows:

1. An agent socket packet linking mechanism and information retrieval service are proposed. The agent and machine controller communicate by sending packets. The agent periodically receives the machine status and production information.

2. Information retrieving agent architecture using agent services are proposed. Client (software) can acquire relevant injection molding machine processing parameters through agent services.

3. TCP/IP communication is used as the client/server transmission data protocol. In addition, a packet resending mechanism is added to ensure safety.

4. An injection molding machine production process database is planned. It contains factory information, machine information, user data, injection molding machine parameters, mode information, and SPC process parameters.

5. Information security architecture is proposed. A high-security asymmetrical random feed hash encryption algorithm is devised to improve information security for both users and factories.

ACKNOWLEDGEMENT

The authors thank the Ministry of Science and Technology for sponsoring this research and providing related technology support (No. MOST-103-2221-E-006-085).

REFERENCES

1. Grant, H.K., Albert, J.S., Danie, G.H. and Theo, I.N., "Intelligent machine agent architecture for adaptive control optimization of manufacturing processes", *Advanced Engineering Informatics*, Vol. 25, No. 4, pp. 783–796, 2011.
2. Kumar, V., Mishra, N., Mishra, N., Chan, F.T.S., Gnanasekar, B., "Warranty failure analysis in service supply chain: A multi-agent framework", in *Proeedings of International Conference on Supply Chain Management and Information Systems*, pp. 1–6, 2010.
3. Zhang, X.Y. and Qu, Z.G., "The study on collaborative manufacturing platform based on agent", *Physics Procedia*, Vol. 25, pp. 1569–1575, 2012.
4. Moser, T., Merdan, M. and Biffl, S., "A pattern-based coordination and test framework for multi-agent simulation of production automation systems", in *Proceedings of International Conference on Complex, Intelligent and Software Intensive Systems*, pp. 526–533, 2010.
5. Rizvan, E., Cenk, S., Adil, B. and Vahit, K., "A multi-agent based approach to dynamic scheduling of machines and automated guided vehicles in manufacturing systems", *Applied Soft Computing*, Vol. 12, No. 6, pp. 1720–1732, 2012.
6. Aburukba, R., Ghenniwa, H. and Weiming, S., "Agent-based approach for dynamic scheduling in content-based networks", in *Proceedings of IEEE International Conference on e-Business Engineering*, pp. 425–432, 2006.
7. Guo, Q.L. and Ming, Z., "An agent-oriented approach to resolve scheduling optimization in intelligent manufacturing", *Robotics and Computer-Integrated Manufacturing*, Vol. 26, No. 1, pp. 39–45, 2010.
8. Qi, H., Weiming, S. and Lihui, W., "Towards a cooperative distributed manufacturing management framework", *Computers in Industry*, Vol. 56, No. 1, pp. 71–84, 2005.
9. Qinglin, G. and Ming, Z., "A novel approach for multi-agent-based intelligent manufacturing system", *Information Sciences*, Vol. 179, No. 18, pp. 3079–3090, 2009.

10. N.H., Zhang, D.Z. and Q.L., "Agent-based hierarchical production planning and scheduling in make-to-order manufacturing system", *International Journal of Production Economics*, Vol. 149, pp. 117–130, 2014.
11. Wei, J., Jun, M., Xiu, Z. and Huan, X., "Research on cloud manufacturing resource integrating service modeling based on cloud-Agent", in *Proceedings of IEEE International Conference on Software Engineering and Service Science*, pp. 395–398, 2012.
12. Hongbin, Z., Guixia, Y., Gang, L., Baorui, C. and Xin, X., "An intelligent inter database retrieval system based on multi-agent", in *Proceedings of World Automation Congress*, pp. 101–105, 2010.
13. Yazen, A. and Valeriy, V., "Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing", *Robotics and Computer-Integrated Manufacturing*, Vol. 26, No. 4, pp. 381–391, 2010.
14. Yingfeng, Z., George, Q.H., T.Q., Oscar, H. and Shudong, S., "Agent-based smart objects management system for real-time ubiquitous manufacturing", *Robotics and Computer-Integrated Manufacturing*, Vol. 27, No. 3, pp. 538–549, 2011.
15. Luis, M.V., Rene, J.R.T., Roque, A.O.R., Gilberto, H.R. and Eduardo, C.Y., "Open-architecture system based on a reconfigurable hardware-software multi-agent platform for CNC machines", *Journal of Systems Architecture*, Vol. 56, No. 9, pp. 407–418, 2010.
16. Jianxin, R.J, Xiao, Y. and Arun, K., "An agent-based framework for collaborative negotiation in the global manufacturing supply chain network", *Robotics and Computer-Integrated Manufacturing*, Vol. 22, No. 3, pp. 239–255, 2006.
17. Minarolli, D. and Freisleben, B., "Virtual machine resource allocation in cloud computing via multi-agent fuzzy control", *International Conference on Cloud and Green Computing*, pp. 188–194, 2013.
18. Liao, L.M. and Huang, C.J., "Applying agent-based negotiation approach for parallel machine scheduling with preference of machines", in *Proceedings of International Conference on Computers and Industrial Engineering*, pp. 1–6, 2010.
19. Ling, S. H., Lau, S. C., Luo, M. and Ge, S.S., "Dynamic resource allocation with machine degradation using decentralized multi-agent approach", in *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 1176–1181, 2009.
20. Chen, Y.Y., Chen, S.L., Hsiao, Y.H. and Wang, S.R., "A novel framework for agent-based production remote monitoring system design: A case study of injection machines", *Advances in Mechanical Engineering*, Vol. 2013, pp. 1–8, 2013.
21. Chen, S.L., Lin, Y.C., Chen, Y.Y. and Wang, S.R., "Research of agent-based production information retrieving system for injection molding machines: A case study", in *Proceedings of Conference on Precision Machinery and Manufacturing Technology*, Kenting, Taiwan, 2014 [in Chinese].
22. Chen, S.L., Lin, Y.C., Chen, Y.Y. and Wang, S.R., "Design of agent-based production information retrieval system architecture for injection molding machines", *Applied Mechanics and Materials*, in press.