MONITORING AND CONTROLLING OF INDUSTRIAL AUTOMATION SYSTEM CONSIDERING VAGUE INPUTS

NASSER JAZDI-MOTLAGH

Institute of Industrial Automation and Software Engineering – University of Stuttgart
Pfaffenwaldring 47, 70550 – Stuttgart – Germany
nasser.jazdi@ias.uni-stuttgart.de

VICENTE F. DE LUCENA JR.

CETELI / PPGEE / PPGI – Universidade Federal do Amazonas - UFAM
Av. General Rodrigo Octávio Jordão Ramos, 6200, Setor Norte, Coroado I -- Manaus, AM, Brasil
vicente@ufam.edu.br

Abstract— This paper presents a new model for controlling and monitoring Industrial Automation Systems. The architecture of this model does not limit it to be connected to only one specific system but makes it able to be adapted to different automation systems as well as reused for new ones, packing any industrial communication protocol. The basic components have their hardware and software described, as well as the interfaces with the users, with the automation system, and between components. Thus, the proposed structure is presented, and later it is shown how it is possible to reuse this system for other applications. Another aspect of the approach is to deal with uncertain statements of the user. It allows the application to interpret these inputs. Furthermore, the application learns, over time, how to better interpret user preferences and behavior patterns.

Keywords— Neuronal Networks, Fuzzy, Uncertainty, Industrial Automation, Machine Learning, Industrial Automation HMI

Resumo— Este artigo apresenta um novo modelo para controle e monitoramento de sistemas de automação industrial. A arquitetura deste modelo não a limita a estar conectada a apenas um Sistema de Automação Industrial específico, mas a torna capaz de ser adaptada a diferentes sistemas, bem como reutilizada para novos sistemas, englobando qualquer protocolo de comunicação industrial. Os componentes básicos da proposta têm seu hardware e software descritos, assim como as interfaces com o usuário, com o sistema de automação e entre os componentes. Assim, a estrutura proposta é apresentada e posteriormente é mostrado como é possível reutilizar este sistema para outras aplicações. Outro aspecto da abordagem é lidar com declarações incertas do usuário. O sistema aqui apresentado permite que o aplicativo interprete essas entradas. Além disso, o aplicativo aprende com o decorrer do tempo e de novos eventos a interpretar melhor as preferências do usuário e os padrões de comportamento.

Palavras-chave— Neuronal Networks, Fuzzy, Uncertainty, Industrial Automation, Machine Learning, Industrial Automation HMI

1 Introduction

The economic benefits brought using Industrial Automation Systems (IAS) in the industry made these solutions vital to modern industry (Maga et al., 2011). Therefore, controlling and monitoring them has become a critical task for these industries (Sauter et al., 2011). These systems are composed solely of electronic and mechanical devices, which, combined with software embedded in microcontrollers, allow industrial activities to be performed more accurately and quickly than a human could do. However, these systems mostly operate in specific applications and use proprietary solutions regarding interfaces for controlling and monitoring technical processes, which leads to higher construction and maintenance costs. Other aspects are their low portability, scalability, modularity, and compatibility, as well as partial standardization (Wang, 2002). Those aspects altogether increase the costs of construction and maintenance of industrial automation systems.

To develop automation systems that are more suitable to the modern industries reducing their maintenance costs and making them easier to use, many research proposals try to increase the intelligence of the systems themselves. For instance, Machine Learning methods try to replicate the biological ability to learn in machines and thus provide methods for those machines to modify their structure, their program or their data, in order to achieve performance improvement next time, the system runs. That can be achieved by using various approaches, such as Knowledge-based approaches, Statistical Models, Evolutionary Models, etc. These approaches are used primarily to create software agents that can process data inputs, e.g., sensor readings or speech processing, and decide what kind of output they are supposed to deliver at that moment.

Therefore, mechanisms of making IAS modular and adaptable are essential goals of the research field of Control and Automation. Devices capable of operating with vague and uncertain input information from an operator reduce the need for deep knowledge of how the machine's interface works. Likewise, devices able to modify their structure by themselves to obtain better results in future executions also reduce the need for parameter calibration.

Based on the concepts mentioned above, this paper proposes a modular system that reduces the developing effort to connect to Industrial Automation Systems with the intention of monitoring and controlling them. A Neuro-Fuzzy system extends the proposal, used to interpret user's vague and uncertain intentions given as inputs and ultimately to improve system's performance for future executions. It uses a remote user interface device with computing capability, such as Smartphones, Tablets or PCs, integrated to an Embedded Platform (EP) that then connects to the industrial automation system. That allows controlling and monitoring the automation system remotely through available networks, such as LAN or WLAN.

In the interface, which the user can access the industrial automation system, Neuro-Fuzzy algorithms are used as Cloud Computing services, given computational limitations of Smartphones and Tablets, where this interface runs on. The Cloud Computing services would be responsible for processing user requests, as vague and uncertain inputs, and then modify its algorithms for improving results acquired from the industrial automation system.

Another proposal for this system is a standardized and platform independent interface to access industrial communication protocols, which are present in the Embedded Platform. This goal is reached using Web technologies, such as REST-based Web Services over TCP/IP protocols.

This paper is organized as follows: section 2 presents the structure of the model. In section 3 the related work is presented. In section 4 it is described, how this concept is being idealized, and finally in section 5, this work is summarized, and the proposed future work will be presented.

2 Related Work

The Institute of Applied Computer Science (Fraunhofer, 2017) and the Institute of Intelligent Analysis and Information Systems (Fraunhofer, 2018) research the topic Machine Learning, which is the topic of computer science field that deals with algorithms to analyze patterns and data dependencies (Kemmerich et al., 2011). They aim to allow predictions about data and to enable faster and better decisions. That is a very useful approach as more data is collected daily in business, science, and also in our society. Practically, they work on topics such as Mobility Mining, Text Mining, and Information Retrieval. For the realization of the applications, they used Fuzzy Logic, Artificial Neuronal Networks and Evolutionary Algorithms (Liebig et al., 2011).

The Institute of Computer Science (Uni-Paderborn, 2018) also works on the topic applied Machine Learning. Their research activities focus on the merging of two research topics, namely, agent-based solutions and machine learning algorithms. They incur that the knowledge and capabilities of agents are restricted due to resource limitations. They do research on partitioning the problems and on the hierarchical organization of multiple agents as well as acting in highly dynamic environments. Techniques taken from evolutionary computation and swarm intelligence contributed to powerful learning approaches in this research context, i.e., the evolution of sets of rules for reactive agent control.

Nilsson presents an approach for machine learning (Nilsson, 2010) where he describes the theory involved thoroughly and brings many practical examples. His notes present an overview about different methods and concepts of machine learning. Also, different learning methods such as Computational Learning Theory, Inductive Logic Programming, Decision Trees and Boolean Functions are described and discussed in detail.

Other authors focused on artificial intelligence applications for image processing on industrial environments (Wang et al., 2015) and diverse maintenance approaches of industrial plants (Kroll et al., 2014, and Matthew et al., 2017). Moreover, techniques for monitoring and controlling the industrial process itself still being a research object of the automation community (Ferreira et al., 2017, and Budiman et al., 2014).

3 Overview of the Proposed System

The system consists of three major components, as seen in Fig. 1. The first, called Human-Machine Interface (HMI), is a device that provides a graphical user interface to monitor and to control the IAS, as well as to access configuration screens for setting up communication with other components (Pfeiffer et al., 2016), (Wang and Canedo, 2014). The second is the Embedded Platform, responsible for connecting the IAS' communication bus to the LAN and WLAN networks through Web technologies. Finally, the third component is the Cloud Computing service that interprets vague and uncertain inputs from the user, as well as modifies its structure to improve outcomes in new executions.

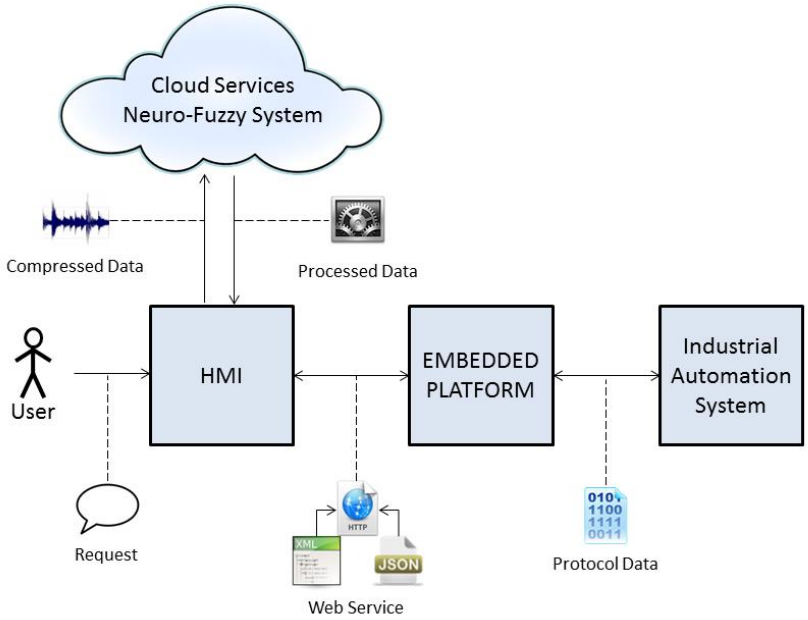

Figure 1. System's Block Diagram

The IAS is not considered a major component of the system because it is the target of monitoring and controlling. However, it is necessary to know how it operates and, consequently, which data needs to be sent by the Embedded Platform to retrieve data and not to damage it correctly.

From the user's perspective, the operation of the proposed system consists of three interactions, shown in Fig. 2. It begins with a request that serves as input to the Neuro-Fuzzy application running through Cloud Computing. This application generates the parameters for the user's request, and the system sends them to the IAS. Furthermore, the system responds to the user with the delivery of his request. Finally, the user provides the system with a quality evaluation of the response given. If the user reports that the response does not comply with what was requested, this information is fed back to the Neuro-Fuzzy application, to modify the interpretation structure (Fullér, 1995). This procedure aims to improve the system for the next time an order is placed. Thus, the proposed concept provides a user interface that requires minimal knowledge on the system, and that calibrates itself to improve the results of future requests.

The use of Cloud Computing eliminates the need for complex hardware and software structures in the HMI, as these types of services provide simplified interfaces based on Web technology and with support for multiple programming languages. The only requirement is a connection to the server where the application will be running. Examples of other approaches applied to industrial automation system were presented by Hegazy (Hegazy et al., 2015) and by Dai (Dai et al., 2018).

The integration of the HMI and the Embedded Platform is specified to keep them independent of any connected device. That means that all information regarding the operation of the IAS is stored in the HMI and that the Embedded Platform has only rules necessary to generate the messages used by the industrial communications protocol, i.e., no data processing is performed at the platform.

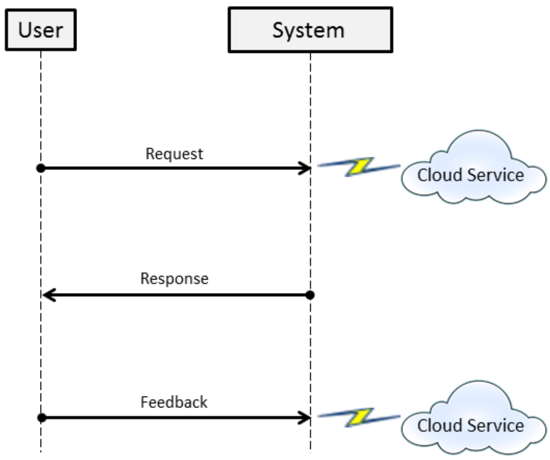

Figure 2. Operation Mode

With this architecture, it is possible to reuse the system's structure to communicate with different Industrial Automation Systems, independent of the used communication protocol. The Embedded Platform, once connected to an IAS, provides an access interface through LAN or WLAN, based on the HTTP protocol, since it is only necessary for an HMI to have knowledge about the IAS, which uses this interface to exchange monitoring or control data. In order to add new software components, such as industrial communications protocols, IAS' knowledge bases or interpretation algorithms for user's requests, it does not require much development effort since flexible interfaces and standardized technologies are being proposed as internal components.

4 Proposed Approach Details

Playing the role of the HMI, an Android OS device is recommended, being it a smartphone or tablet, but the concept can also be applied to other operating systems for mobile devices, such as iOS or Windows Phone, as well as to PC Operating Systems, like Windows, Mac OS or Linux. However, not all programming languages provide libraries that can process vague data inputs. Therefore, the availability of such libraries must be checked before choosing the OS and the runtime platform for the HMI. For example, if interaction with the HMI is being done via speech, libraries as Dragon NaturallySpeaking for Windows or as the Dragon Mobile for Android, iOS and Windows Phone can be used. Another example would be the use of images or videos to interact with the HMI, then libraries such as OpenCV could be used.

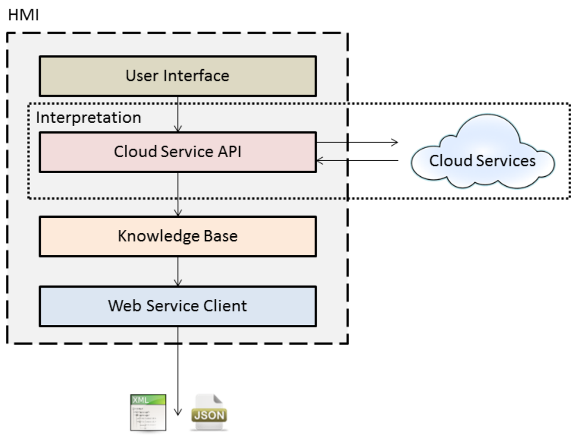

Figure 3. HMI's Software Structure

The HMI software structure is shown in Figure 3. There is a layer dedicated to user interface, through which the request is made. The interpretation layer is where data regarding the user's request is processed by Cloud Computing services and where the key aspects of each request is identified (Hazarika et al., 2015). These data are related to the functions offered by the IAS and finally processed by Fuzzy operators that generate the parameters that are going to be forwarded. After interpretation, data goes to the knowledge base layer where a set of messages is built and passed as arguments to the client Web Service based on REST architecture (Representational State Transfer). This client formats the data in JSON or XML standards and sends it via HTTP protocol over LAN or WLAN connections.

The Embedded Platform, shown in Figure 4, consists of a processing unit based on the ARMv5 architecture, used by the NXP LPC3250 ARM926EJ-S processor, accompanied by 64MB of RAM, 128MByte NAND FLASH, NOR FLASH 4MByte and WLAN, available via USB adapter, and LAN interfaces for connection to the HMI.

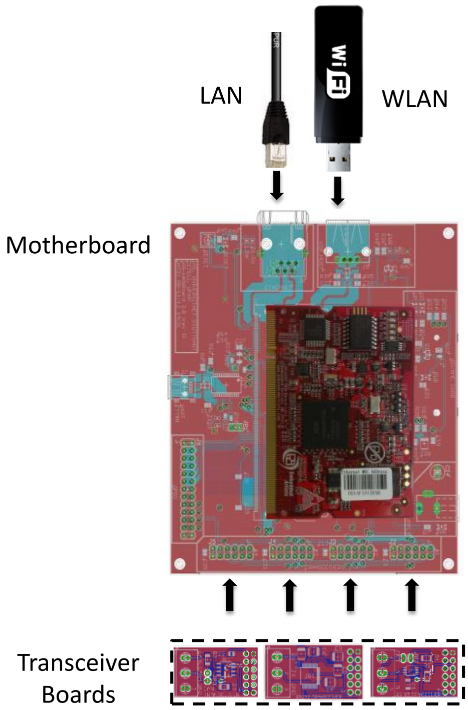

Figure 4. Hardware Components for the Embedded System

Auxiliary boards are used to electrically connect the Embedded Platform with industrial communication buses and communicate via UART, I²C, and SPI to the processor. This concept assumes that several integrated circuits can receive data through UART, SPI or I²C buses and convert it to electrical levels of industrial communication buses, such as CAN, RS-485, LIN, etc. Consequently, when a connection to an IAS is necessary, an auxiliary board containing a transceiver along with its conditioning circuit is needed and it must also provide one of the interfaces mentioned above to allow the Embedded Platform to exchange data with the IAS.

The Embedded Platform's software runs on an Embedded Linux operating system, which is a customized version of Linux OS, and its structure is shown in Figure 5. This operating system was chosen because of the advantages offered to manage protocol stacks such as TCP/IP, HTTP, and the USB ports.

As mentioned previously, the interface provided by the embedded platform consists of a Web Service server based on REST architecture. The set of messages is received in XML or JSON format and converted to the specified communication protocol. The conversion is as follows. A binding is done between the received data and the parameters for the communication protocol (Data Binding Layer). In other words, the received data is associated to the components of a protocol message, such as device addresses, memory addresses, values and other parameters. Afterward, when the data correlation is done the vectors that represent the messages are created (Message Generation Layer).

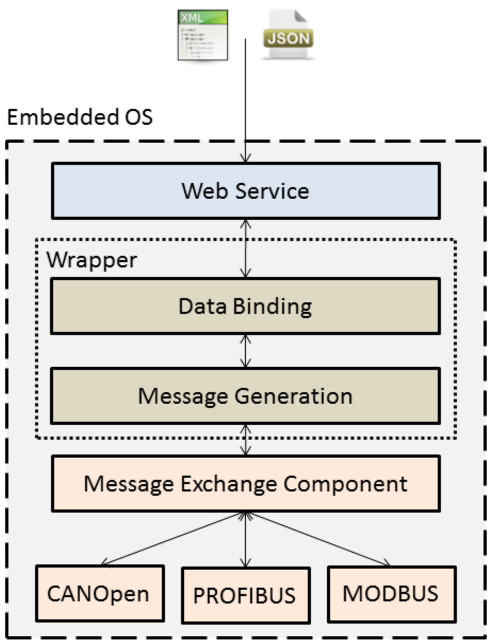

Figure 5. Embedded System's Software Structure

After the creation of vectors, these are referred to the software component responsible for exchanging messages with the IAS, present in the Messaging Exchange Layer. In this layer, a state machine performs the procedures of communication, i.e., reading and writing on the bus. It is also responsible for storing the data read from the IAS so that they can be accessed and processed by the HMI.

Finally, the model for the interpretation and learning based on Neuro-Fuzzy systems is shown in Figure 6. Unlike most models of this type, in which the results are fed into a learning algorithm which, in turn, generates the data required for the Neural Network to modify the Fuzzy system, in this proposal the user will feed the trained neural network to generate the IF-THEN rules of the Fuzzy system. Therefore, based on this evaluation, the Fuzzy system will be able to improve its structure to deliver better future results.

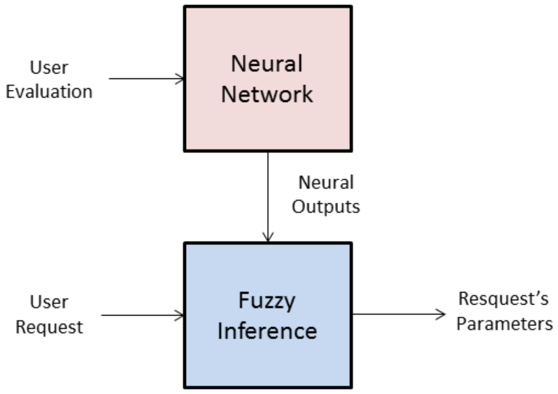

Figure 6. Neuro-Fuzzy Model

4.1 Application Example

As an application example, one automatic coffee machine, the CombiNation S from WMF, was used in conjunction with the proposed system. This machine provides a CANOpen protocol interface through which it is possible to control and diagnose the complete device.

In Fig. 7 it is shown how the application was built. The user speaks to an Android Smartphone using natural sentences to order a coffee, e.g., "I would like a big cup of coffee, but not too hot" or "Please make me a warm and strong espresso." Then, all keywords are identified and forwarded to the Fuzzy system, which generates the parameters for making the desired cup of coffee (water temperature, water amount and coffee amount). When these parameters are found, the Application-specific Knowledge Base builds the message set containing Indexes, Subindexes, and Values that need to be modified in the coffee machine's Object Dictionary. These parameters are sent in XML format to the Embedded Platform via REST-based Web Service.

After the information is extracted from the XML file, the Embedded Platform builds all CANOpen message frames with designated COB-ID and data based on the message type information that is also included in the XML file. It specifies whether the messages should be built as Process Data Object (PDO), Service Data Object (SDO), Emergency Object, or Special Function Objects. With the assembled messages ready, a new thread is created to send them and make the communication with the IAS independent from the Web Service server. If any data is received from the automation system during the communication, it is saved for later evaluation.

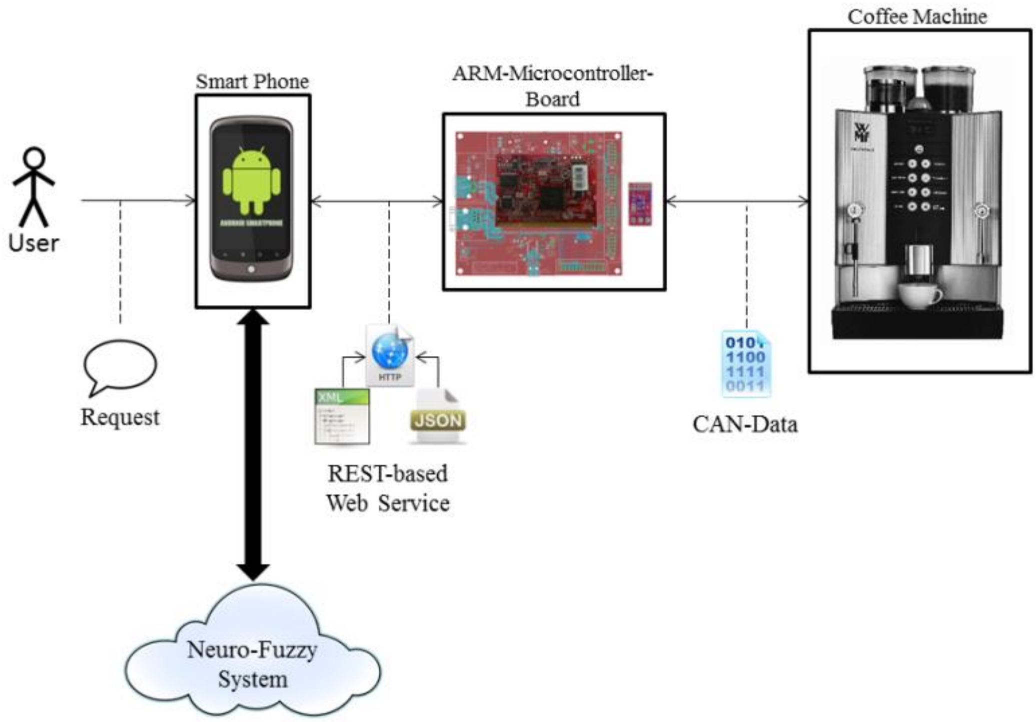

Figure 7. System's Application Example

When the order process is finished, the user makes his evaluation of the coffee in the Smartphone also via speech commands. However, different from the first verbal interaction, the user now says sentences such as “The coffee was a little bit too hot,” “My order is exactly as I wanted” or “It could have been stronger and bigger.” Then, based on the evaluation, the Neural Network processes these data and changes the ranges of each parameter for the next execution.

In fact, we take the wishes from each user and convert it to real values. Firstly, we fuzzify this value (e.g., a little warm means 30% lukewarm and 70% warm for a specific user). Then we interpret these values and generate a concrete value for the coffee machine. For example, the temperature for one espresso for user XY should be 52 degrees. We give this parameter to the system, and the espresso is produced by the XY user preference. Details on the Fuzzy rules will be presented on the conference.

The application was implemented and evaluated for over six months. About 15 different users (some employee and some students of our institute) has worked with the application. The behavior of the device was adapted to different users. They got with the time better products made by the industrial coffee machine. The coffee machine produced different type of the same product (e. g. espresso) for different users.

The initial calibration of the system was done as usual. After starting the system, a routine, previously specified, started running to calibrate the system. The application interpreted the specific and vague wishes of a user by using standard values initially. Afterward, these values were automatically adjusted to each know user. The application generated the necessary parameters for the system.

5 Conclusion

In this paper, a model system that allows access to IAS through LAN or WLAN networks was proposed. The strength of this model is the modularity of hardware and software. One can replace software components, such as protocol stack, knowledge bases, and systems for interpretation and learning, without changing the interfaces between communication devices. From the hardware perspective, the only device that has no flexibility is the Embedded Platform, as it must meet the interface specifications, but the industrial communication buses, the HMI device, and the execution environment for the Neuro-fuzzy system are flexible. The only requirement for these hardware components is to keep the same communication interface between the devices.

Another contribution of this model is to propose a system to control one IAS using uncertain data entries, such as speech. That requires less knowledge about the user who is interacting with the system and keeps the focus on the desired result.

One disadvantage of this proposal is that, on its actual development stage, it is not possible to make actual real-time controlling of the IAS. Indeed, the addition of real-time features for the Embedded Operating System is planned to be a next step for this model.

Acknowledgments

We would like to thank you very much FAPEAM, DAAD, and CAPES that supported the cooperation between UFAM and IAS-University of Stuttgart.

References

Budiman, A.; Fanany, M.I.; Basaruddin, C. (2014). Constructive, robust and adaptive OS-ELM in human action recognition. International Conference on Industrial Automation, Information, and Communications Technology 2014, pp. 39 – 45.

Dai, W.; Riliskis, L.; Wang, P.; Vyatkin, V.; Guan, X. (2018). A Cloud-Based Decision Support System for Self-Healing in Distributed Automation Systems Using Fault Tree Analysis. IEEE Transactions on Industrial Informatics. Volume: 14, Issue: 3, 2018, pp. 989 – 1000.

Ferreira, B.; Silva, R.G.; Pereira, V. (2017). Feature selection using non-binary decision trees applied to condition monitoring. 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2017), pp. 1–7.

Fraunhofer (2017). Fraunhofer-Institut für Angewandte Informatik, http://www.iai.fzk.de/www-extern

Fraunhfer (2018). Fraunhofer-Institut für Intelligente Analyse- und Informationssysteme IAIS - http://www.iais.fraunhofer.de/4859.html

Fullér, R. (1995). Neuro-Fuzzy System. Abo Akademi University - Finland,1995.

Hazarika, P.; Shenoy, S.; Tolety, S.B.; Kalekar, N. (2015). Mobile cloud integration for industrial data interchange. Int. Conf. on Advances in Computing, Communications, and Informatics (ICACCI), 2015, pp. 1118 – 1122.

Hegazy, T. and Hefeeda, M. (20150. Industrial Automation as a Cloud Service. IEEE Transactions on Parallel and Distributed Systems. Volume: 26, Issue: 10, 2015, pp. 2750 – 2763.

Kemmerich, T.; Leine Büning, H. (2011). A Convergent Multiagent Reinforcement Learning Approach for a Subclass of Cooperative Stochastic Games. In Proceedings of the Adaptive and Learning Agents Workshop 2011 (ALA@AAMAS 2011). , pp. 75-82, 2011.

Kroll, B.; Schaffranek, D.; Schriegel, S.; Niggemann, O. (2014). System modeling based on machine learning for anomaly detection and predictive maintenance in industrial plants. Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), pp. 1 – 7.

Liebig, Th.; Xu, Z.; May, M.; Wrobel, S. (2012). Pedestrian Quantity Estimation with Trajectory Patterns. In: Flach, Peter and De Bie, Tijl and Cristianini, Nello u.a. (Hrsg.): Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD) 2012, SEP 24-28, Bristol, Großbritannien. Springer Berlin / Heidelberg (LNCS, Bd. 7524, Nr. 2), S. 629-643

Maga, C.; Jazdi, N., Goehner, P. (2011). Reusable Models in Industrial Automation: Experiences in Defining Appropriate Levels of Granularity. 18th World Congress of the International Federation of Automatic Control (IFAC 2011).

Mathew, J.; Luo, M.; Pang, C.K. (2017). Regression kernel for prognostics with support vector machines. 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2017), pp. 1 – 5.

Nilsson, Nils J. (2010). Introduction to Machine Learning, Description. Artificial Intelligence Laboratory, Department of Computer Science, Stanford University, Stanford, CA 94305- June 19, 2010.

Pfeiffer, T.; Hellmers, J.; Schön, E.M.; Thomaschewski, J. (2016). Empowering User Interfaces for Industrie 4.0. Proceedings of the IEEE, Volume: 104, Issue: 5, 2016, pp. 986 – 996.

Sauter, T.; Soucek, S.; Kastner, W.; Dietrich, D. (2011). The Evolution of Factory and Building Automation. Industrial Electronics Magazine, IEEE, September 2011, 5(3), pp. 35-48.

Uni-Paderborn (2017). University of Paderborn, Knowledge-Based Systems -http://www.cs.uni-paderborn.de/fachgebiete/fg-kleine-buening/startseite.html

Wang, L., (2002). Factory Automation Systems: Evolution and Trends. AUTOTESTCON Proceedings, 2002. IEEE, pp. 880-886.

Wang, L. and Canedo, A. (2014). Offloading industrial human-machine interaction tasks to mobile devices and the cloud. Proc. of the IEEE Emerging Technology and Factory Automation (ETFA), 2014, pp. 1 – 4.

Wang, Y. and Weyrich, M. (2015). An assistant for an incremental learning based image processing system. IEEE International Conference on Industrial Technology (ICIT 2015), pp. 1624 – 1629.