ISSN 2229-5518

Towards the Design of Cyber-Physical System via Multi-Agent System Technology

Ali Aliyuda

Abstract— Advances in digital electronics have led to a significant increase in the number of systems that couple the digital (cyber) systems with the physical world, namely what have become known as the Cyber-Physical System (CPS). The design of CPS requires a significant amount of reasoning with respect to unique challenges and complex functional, reliability, and performance requirements. This paper presents the suitability of Multi-Agent System technology towards resolving some challenges in the design of CPS. In the proposed method, challenges that are common to most CPS design and the attributes and behaviour characteristics of agents in Multi-Agent System to tackle such challenges are outlined. The correctness of this approach is shown from the perspective of Agent-Oriented Programming (AOP) with Java Agent Development (JADE) Platform where agents exhibit autonomous and interactive behaviour. The capability of agents in MAS to interact with one another and with their environments; their flexibility and freedom to monitor, control and change their behaviours are considered appropriate solutions to the prominent CPS design challenges.

Index Terms— Agents, behaviours, Agents' Communication, Cyber Physical System, Environment, JADE, Multi-Agent System.

1 Introduction

Cyber-physical systems are systems where real-time computing and physical systems interact tightly [27]. The author in [27] added that the concept of cyber-physical systems can be taken as meaning large complex physical systems that are interacting with a considerable number of distributed computing elements for monitoring, control and management which can exchange information between the system and its environment and with human users.

Although Cyber-Physical System (CPS) is a relatively new concept, the systems' components are well-known. A CPS is composed of the physical world, interfaces, and cyber systems. The physical world refers to the physical phenomena wanted to be monitored or controlled. The cyber systems refer to the next generation embedded devices, which process information and communicate with their distributed environment. The interfaces refer to the communication network and other intermediate components responsible for bridging the cyber systems with the physical world [19].

Various research areas and terminologies are relatively similar to CPS [30]. A range of concepts similar to CPS is illustrated in Figure 2 and brief explanation of each is given as follows:

The term Big Data refers to the datasets that are too large and complex to capture, store, manage, and analyse with standard methods or database tools [11]. A large scale CPS can be envisioned as millions of networked smart devices, sensors, and actuators being embedded in the physical world, which can sense, process, and communicate the data all over the network.

Aliyuda Ali obtained his MSc. In Computer Science from Usmanu Danfodiyo University, Sokoto, Nigeria in 2016 and B. Sc. in Computer Science from Gombe State University Gombe, Nigeria in 2009. He is currently working as a Lecturer in the Department of Mathematics/Computer Science of Gombe State University, Nigeria. His areas of research include Multi-Agent Systems and Empirical Software Engineering.
Email: aliyuda.ali6@gmail.com

Inspired by the idea of interconnecting smart devices, the term the Internet of Things (IoT) was coined in 1999. IoT was envisioned as future Radio Frequency Identification (RFID) technology that enables the automatic identification of the physical objects via a small electronic chip called "RFID tag". IoT provides an opportunity to observe, identify, and understand the real world by capturing data about the things (i.e. RFID tagged objects) and helping businesses achieve greater efficiency and accountability [14], [24]. IoT greatly overlaps with CPS, because IoT addresses observing the things in the physical world, exploiting communication capabilities, and capturing data needed to manage the things that are not efficiently managed today [14], [15].

Cloud is a paradigm shift in the Information and Communications Technology (ICT), through which businesses and users can have an on-demand network access to a shared pool of configurable computing resources (e.g. hardware, applications, services, etc.) [21], [25]. Cloud computing provides new opportunities for CPS in management and processing of aggregated sensor data and decision-making methods based on a cloud model which allows CPS to enhance the system capability.

The term Cybernetics refers to an approach describing the study of communication and control characteristics in both machines and in living beings [18]. The theory of Cybernetics and the practice of mechatronic system design lay the foundations for the design of CPS [26].

Systems of Systems (SoS) refers to large-scale, heterogeneous systems networked together for a common goal and composed of inherently autonomous components that can be operated and managed independently [17].

Machine-to-Machine (M2M) communication is another concept related to CPS. M2M refers to smart devices, such as computers, embedded processors, smart sensors, actuators, and mobile devices, talking to each other via a communication network [2], [28]. M2M is a communication standard that is a subset of both IoT and CPS.

ISSN 2229-5518

The term Mechatronics is the combination of "mecha", referring to mechanical systems, and "tronics", referring to electronic systems. The term was coined in the late 1960s. However, it has evolved over the decades comprising software and information technologies. Therefore, it can be considered as a systematic approach to design, develop, and implement complex engineering systems which incorporate information technologies into the physical domain [31].

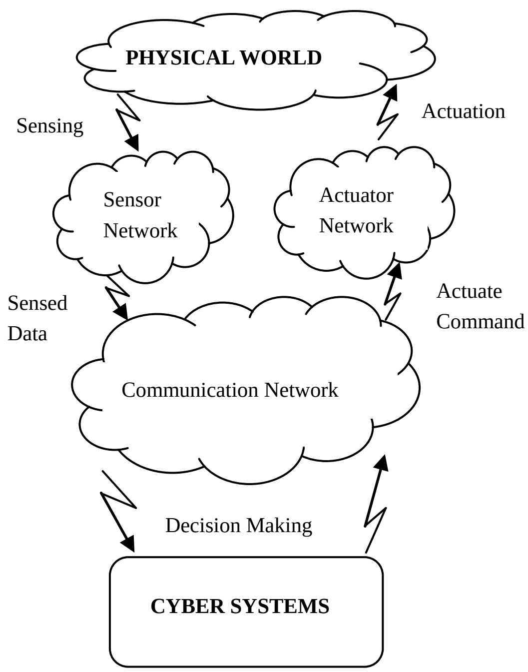

Fig. 1. CPS Holistic View [30]

A wireless sensor network is a collection of nodes organised into a cooperative network [13]. Each node consists of processing capability (one or more microcontrollers, CPUs or chips), may contain multiple types of memory (program, data and flash memories), has a transceiver (usually with a single Omni-directional antenna), has a power source (e.g., batteries and solar cells), and accommodate various sensors and actuators.

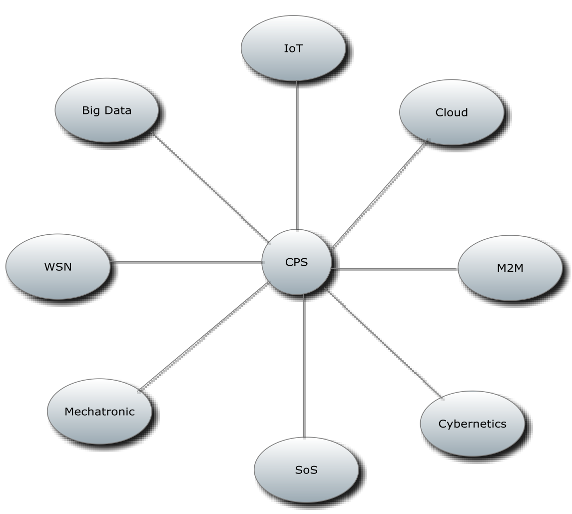

Fig. 2: CPS Similar Concepts [30]

Agents and Multi-Agent Systems have recently emerged as a powerful technology to face the complexity of a variety of modern Information Systems [8]. It is clear that an agent system cannot be simply reduced to a group of interacting agents. Instead, the complete modelling of the system requires explicitly focusing also on the environment in which the MAS and its constituent agents are situated and on the society that a group of interacting agents give rise to. Modelling the environment implies identifying its basic features, the resources that can be found in the environment, and the way via which agents can interact with it. Modelling agent societies implies identifying the overall rules that should drive the expected evolution of the MAS and the various roles that agents can play in such a society [6].

Looking at the structure of CPS holistically (Figure 1), the author is motivated that the concept of (MAS) Technology can play vital roles in the design of reliable CPS for the following reasons:

i. Agent-based computing promotes designing and developing applications in terms of autonomous software entities (agents), situated in an environment, and that can flexibly achieve their goals by interacting with one another and with the environment in terms of high-level protocols and languages [6].

ii. The flexible way in which agents operate and interact (both with each other and with the environment) is suited to the dynamic and unpredictable scenarios where software is expected to operate [6];

iii. The concept of agency provides a unified view of artificial intelligence (AI) results and achievements, by making agents and MAS act as a sound and manageable repositories of intelligent behaviours [31].

2 CPS DESIGN CHALLENGES

A CPS is a promising technology for the design of current and future engineered systems and is expected to make an important impact on our interactions with the real world. Cyber-Physical Systems revolutionise our interaction with the physical world. Of course, this revolution does not come free. Since even legacy embedded systems require higher standards than general-purpose computing, we need to pay special attention to this next generation physically-aware engineered system requirements if we really want to put our full trust in them. As such, the authors in [30] give consideration to some system-level requirements/challenges that need to be addressed in CPS design. In their survey, the authors listed among others the following as challenges associated with the design of CPS:

Dependability refers to the property of a system to perform required functionalities during its operation without significant degradation in its performance and outcome. Dependability reflects the degree of trust put in the whole system. Cyber and physical components of the system are inherently interdependent and those underlying components might be dynamically interconnected during the system operation, which, in return, render dependability analysis very difficult. The authors suggested that a common language to express dependability related information across constituent systems/underlying components should be introduced in the design stage.

Availability refers to the property of a system to be ready for access even when faults occur. A highly available system should isolate malfunctioning portion from itself and continue to operate without it. Malicious cyber-attacks (e.g. denial of service attacks) hinder the availability of the system services significantly.

Compositionality refers to the property of how well a system can be understood entirely by examining every part of it. A highly compositional system should provide great insight about the whole from derived behaviours of its constituent parts/components. The authors noted that achieving high compositionality in the CPS design is very challenging especially due to the chaotic behaviour of constituent physical subsystems.

Adaptability refers to the capability of a system to change its state to survive by adjusting its own configuration in response to different circumstances in the environment. A highly adaptable system should be quickly adaptable to evolving needs/circumstances.

Re-configurability refers to the property of a system to change its configurations in case of failure or upon inner or outer requests. A highly reconfigurable system should be self-configurable, meaning able to fine-tune itself dynamically and coordinate the operation of its components at finer granularities. CPS can be regarded as autonomously reconfigurable engineered system. Remote monitoring and control mechanisms might be a necessity in some CPS application scenarios [16].

Heterogeneity refers to the property of a system to incorporate a set of different types of interacting and interconnected components forming a complex whole. CPS is inherently heterogeneous due to constituent physical dynamics, computational elements, control logic, and deployment of diverse communication technologies.

Interoperability refers to the ability of the systems/components to work together, exchange information and use this information to provide specified services. A highly interoperable system should provide or accept services conducive to effective communication and interoperation among system components.

3 RELATED WORKS

CPS has been defined by the scientific community from different perspectives. In [20] CPS is described as "physical and engineered systems, whose operations are monitored, coordinated, controlled, and integrated by a computing and communicating core". In [12] CPS is described as "integrations of computation with physical processes". In [3] CPS is described as "embedded systems together with their physical environment". The authors in [10] described CPS as "physical, biological, and engineered system whose operations are integrated, monitored, and/or controlled by a computational core. Components are networked at every scale. Computing is deeply embedded into every physical component, possibly even into materials. The computational core is an embedded system, usually, demands real-time response, and is most often distributed". A number of articles have addressed necessary problem formulations, system-level requirements, and arising challenges in CPS design. In [22] CPS technology was introduced and research directions for CPS design were suggested. Lee [4] specifically points out the failure of standard abstraction layers, the need for reliable timing behaviour, and lack of temporal semantics of existing programming language models for CPS design. Rajkumar [23] touches on system level aspects of CPS from the scientific and social impact standpoints. Lee [5] suggests two approaches, namely cyberizing the physical and physicalizing the cyber, for integrating the cyber systems with the physical systems.

4 MAS CAPABILITY TO TACKLE CPS DESIGN CHALLENGES

All of CPS applications need to be designed considering the cutting-edge technologies, necessary system-level requirements, and overall impact on the real world. Looking at the challenges CPS design faces and some suggestions given in the related literature, the author of this paper outlined some capabilities of MAS towards resolving the challenges in CPS design as follows:

4.1 MAS versus CPS

Figure 3 gives an abstract view of an agent. In this diagram, the agent takes sensory input from the environment and produces as output actions that affect it. The interaction is usually an ongoing, non-terminating one. We can see the action output generated by an agent in order to affect its environment.

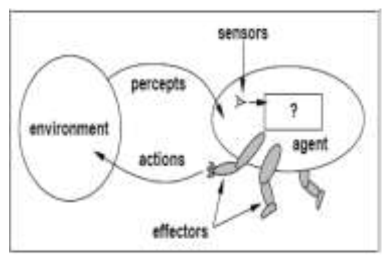

Fig. 3. Agents interact with environments through sensors and effectors [29].

If we compare the behaviour of an agent in its environment (Figure 3) and the arrangement of CPS in Figure 1, we shall see a very close working principle among MAS and CPS. This implies that incorporating an agent in the design of cyber part of a CPS will enhance the manner in which the cyber part will exchange information with the physical part and use this information to provide specified services. According to [9], three main attributes of an agent are defined:

a. *Autonomy*, which refers to the fact that an agent should run independently, with little or no human intervention,
b. *Temporal continuity*, which signifies that an agent should run continuously rather than simply perform a task and finish, and
c. *Social skills*, which signifies that an agent should possess some form of social skills since the agent's advantages lie in its interactive communication with other agents.

In addition to these core attributes, the authors in [6] also added that an agent can also be classified according to the following social behaviou*r* characteristics:

i. *Pro-activeness*: this refers to how the agent reacts to - and reasons about its environment, and how it pursues its goals. The agent can directly react to stimuli in its environment by mapping an input from its sensors directly to an action, or it can take a purely planning, or goal-oriented, approach to achieving its goals. This last approach relies upon utilising planning techniques.

ii. *Adaptability*: this describes an agent's ability to modify its behaviour over time. In fact, the term "agent" is often taken to implicitly mean "intelligent agents", which combine traditional artificial intelligence techniques to assist in the process of autonomously performing tasks. This feature includes other sub-features such as learning and sub-submission.

iii. *Mobility*: this refers to the agents' capability of transporting their execution between machines on a network. This form of moving can be physical, where the agent travels between machines on a network, or logical, where an agent which is running on a single machine is remotely accessed from other locations on the Internet.

iv. *Collaboration*: collaboration among agents underpins the success of an operation or action in a timely manner. This can be achieved by being able to coordinate with other agents by sending and receiving messages using some form of agent communication language, and permits a high degree of collaboration, thus making social activities such as distributed problem solving and negotiation possible.

v. Reactivity: the agents perceive their environments (which may be the physical world, a user via a graphical user interface, other agents, etc) and respond to changes that occur in the environment.

A software or hardware that possesses such attributes or characteristics can be considered an agent.

According to [30], the main aim of CPS is to fulfill the vision of the world in which all systems, machines, and objects become smart and physically-aware, have a presence in the cyber-physical space, exploit the digital information and services around them, and communicate with each other as well as with the environment. Looking at the aforementioned attributes and behaviour characteristics of agents, we can clearly see the possibility of achieving the said aim of CPS when agents-based computing techniques are incorporated in the design of CPS.

Furthermore, some of the aforementioned CPS design challenges can be tackled considerating the attributes and behaviour characteristics of agents as follows:

- Dependability challenge can be tackled by an Agent since Agent is autonomous i.e they can deliver the requested services as specified without the need for intrusion;
- Adaptability challenge can also be tackled by agents since agents are reactive i.e they perceive their environments (which may be the physical world, a user via a graphical user interface, other agents, etc) and respond to changes that occur in them;
- Predictability challenge can also be tackled since agents are not just Reactive but also Proactive.
- Interoperability challenge can also be tackled since agents are interoperable as agents communicate using some kind of agent-communication language with other agents (humans or computational) in order to solve a problem.

4.2 Agent-Oriented Programming (AOP): JADE platform

Based on the vision of interoperability between systems with different manufacturers and operators, FIPA released as reference the FIPA standard pattern. This pattern has a primary focus on the external behaviour of system components, leaving open the implementation details and the architectures of the agents [6]. In addition to this feature, it sets the reference model for an agent's platform, as well as the set of services to be offered by the platform. Among these services, there are Directory Facilitator (DF), Agent Management System (AMS), the Message Transport Service (MTS), and the Agent Communication Language (FIPA-ACL). FIPA does not formally implement any agent architecture because its open standards allow various ways to implement it, simply by following the recommendations and abstract mechanisms defined within. Among the generic frameworks that use the FIPA pattern it is possible to cite: JADE [1]. This platform is described as follows:

The Java Agent Development (JADE) Framework is an environment for developing applications according to the FIPA patterns. It is implemented in Java and was developed at the University of Parma, Italy [1]. Some characteristics of this platform are listed below:

- Distributed platform of agents - JADE can be divided into multiple hosts or machines. The agents are implemented in Java threads and inserted into repositories of agents called containers, which provide all the support for the implementation;
- Graphical user interface (GUI) - JADE has a GUI interface that assists in managing agents and agent containers;
- Running multiple, parallel and concurrent activities of agents - JADE provides these features through their pre-defined models of agent's behaviour. The structure of the agent's behaviours using the JADE platform takes place via a scheduler that automatically manages the scheduling of these behaviours.

The JADE architecture is based on the coexistence of several Java Virtual Machines (JVMs) that can be distributed over multiple computers, independently of the operating system. Figure 4 shows the distribution vision of the JADE platform in many hosts. Each host runs the JADE agents that form a container. These containers are registered in the main container of the platform. Each host has a JVM, indicating platform independence, and each JVM has a container of agents that provides a complete running environment for these agents, in addition to allowing multiple agents to run concurrently on the same processor/host. The execution of the JADE platform occurs at the main container of a platform. The other hosts who own the remaining containers should only have the files needed to run the platform.

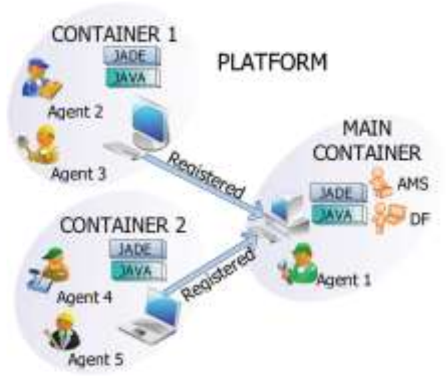

Fig. 4. Functional Architecture of the Jade Platform [6].

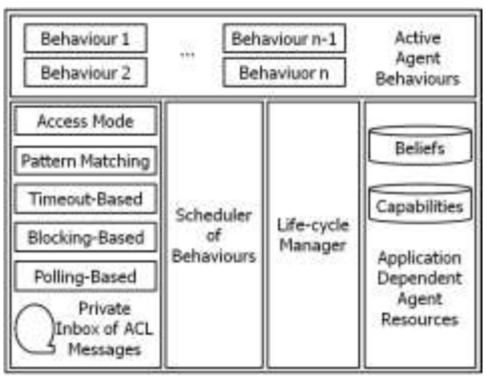

Fig. 5. Internal Architecture of an Agent in the Jade Platform [1].

Figure 4 shows the main container where the AMS, the DF and the Remote Method Invocation (RMI) registry are located. The RMI is a name server used by Java to record and retrieve references to objects by name. With the RMI registry, the JADE platform keeps references of other containers that connect to it.

In the JADE environment, an agent is a process that is autonomous with an identity and requires communication with other agents to execute their functions. The JADE platform is neutral as regards the internal architecture of its agent. A JADE agent runs as a thread that employs multiple tasks or behaviours and simultaneous conversations. Figure 5 shows the internal architecture of the JADE agent. At the top, there are the active behaviours of the agent that represent actions that the agent can perform. The computational model of an agent in JADE is multitasking, where tasks (behaviours) are performed concurrently. Each function provided by an agent must be implemented as one or more behaviours. At the bottom of Figure 5 is possible to see a private messaging box of ACL. Every agent in JADE has this box, and can decide when to read incoming messages and which messages to read. In the centre, there are the behaviour scheduler and life cycle manager. The scheduler is responsible for scheduling the execution order of the behaviours. The life cycle manager is the controller of the current state of the agent. The agent is autonomous; it uses the life cycle manager to determine their current status (active, suspended, etc.). On the right side of Figure 5 there are the application-dependent capabilities of agents, where the beliefs and capacities that the agent acquires during the execution of the application will be stored.

Looking at how agent is being programmed in the JADE platform, the author briefly summarises as follows:

- Parameters that define the state of an agent include but not limited to beliefs, commitments, capabilities, choices and the types of message among agents include but limited to inform, request, offer, promise, decline.
- Autonomy and interaction are the key features that make agents different when they are programmed.
- Agents have freedom to control and change their behaviours; these make them active and passive elements;
- The agent has the ability to communicate with the environment and other entities. Agents may engage in multiple transactions concurrently, through the use of multiple threads or similar mechanisms.
- Agents act with intelligence in their environment and achieve their desires when they are provided with the required knowledge and beliefs.
- Agents provide a more flexible approach. That is, once created; they can change their class or become instances of multiple classes not by inheritance.

The features of Agent-Oriented Programming (AOP) using the appropriate platform such as JADE makes it possible to develop a system that can be dependable, available, adaptable, compositional, reconfigurable, heterogeneous, interoperable, maintainable, safe, reliable, predictable, sustainable and thereby overcome most of the challenges in CPS design.

4.3 Suitability of Multi-Agent Systems in CPS Environment

A multi-agent system (MAS) is a system that contains a set of agents that interact via communications protocols and are able to act on their environment. Different agents have different spheres of influence, mainly because of their control (or at least an influence) on different parts of the environment. In some cases, these spheres of influence may overlap which causes dependency of reports between the agents [6].

From the above definition, we see both MAS and CPS operate in a distributed heterogeneous environment. The major component integrated into the CPS includes Observation, Communication and Controls aspect [30]. This integration can easily be modelled in a Multi-agent based system where Agents can have the ability to observe any changes in their environment, be able to act (sends control) to another agent via Communication.

Various research areas and terminologies are relatively similar to CPS and a range of concepts similar to CPS is illustrated in Figure 2. Based on the working principle of agents in MAS, this paper proposes that a strong relationship between the two disciplines can be established where the concepts can be shared as described in the following figure:

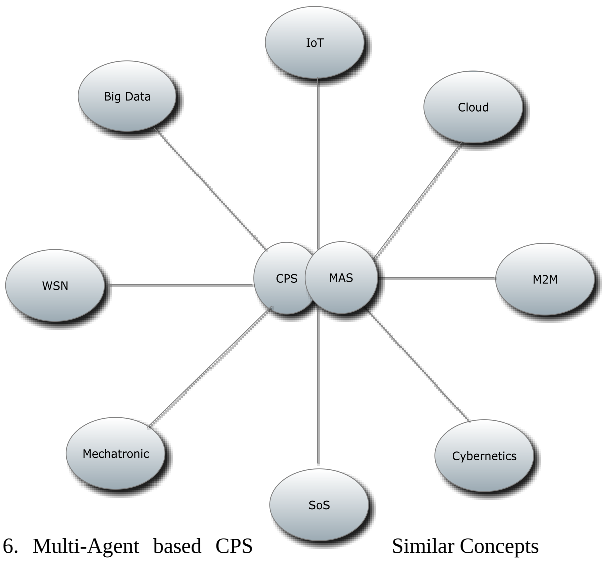

Fig. 6. Multi-Agent based CPS Similar Concepts

5 CONCLUSION

This paper presents the suitability of MAS towards resolving some noticeable challenges in CPS design. Some of the concepts and various research areas and terminologies relatively similar to CPS are discovered closer and similar to MAS as well. Considering the attributes and working principles of agents in MAS, the author proposes the possibility of modelling CPS in a MAS environment.

In future works, MAS approach will be used to model Machine-to-Machine (M2M) communication especially mobile devices.

ACKNOWLEDGMENT

The author wishes to express his gratitude to Prof. H. S. Tantawy, a visiting Professor at Department of Computer Science, Usmanu Danfodiyo University, Sokoto, Nigeria for his interest to supervise this work and his provision of excellent computation facilities to carry out this work. Appreciation also goes to Dr Aminu Mohammed Head of Computer Science Unit, Usmanu Danfodiyo University, Sokoto, Nigeria for his advice, moral support and help to carry out this research work.

REFERENCES

[1] Bellifemine, F. L., Caire, G. and Greenwood, D. (2007). Developing multi-agent systems with JADE, (Wiley Series in Agent Technology), *John Wiley and Sons*, pp. 37 – 40.

[2] David S. Watson (2004). “Machine to Machine (M2M) Technology In Demand Responsive Commercial Buildings,” in *Proc. of the ACEEE 2004 Summer Study on Energy Efficiency in Buildings: Breaking out of the Box*, pp. 22–27.

[3] Edward A. Lee and Sanjit A. Seshia, “Introduction to Embedded Systems: A Cyber-Physical Systems Approach,” 1st Edition, 2011.

[4] Edward A. Lee (2010). “CPS Foundations,” in *Proc. of 47th IEEE/ACM Design Automation Conf*erence.

[5] Edward A. Lee (2008). “Cyber Physical Systems: Design Challenges,” in Proc. of 11th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, pp. 363–369.

[6] Faisal Alkhateeb, Eslam Al Maghayreh and Iyad Abu Doush (2011). “Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications”. Janeza Trdine 9, 51000 Rijeka, Croatia. ISBN 978-953-307-174-9, pp. 7, 327 - 328

[7] Grit Denker, Nikil Dutt, Sharad Mehrotra, Mark-Oliver Stehr, Carolyn Talcott, and Nalini Venkatasubramanian (2012). “Resilient Dependable Cyber-Physical Systems: A Middleware Perspective,” *Journal of Internet Services and Applications*, vol. 3, no. 1, pp. 41-49,

[8] G. Weiss (2000). “Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence,” The MIT Press.

[9] Hector, A. and Lakshmi Narasimhan, V. (2005). “A New Classification Scheme for Software Agents.” *Proceedings of the Third International Conference on Information Technology and Applications (ICITA’05)*, pp. 191 – 196, ISBN: 0-7695-2316-1, Sydney, Australia, July 2005, IEEE Computer Society, Washington, DC

[10] Imre Horvath and Bart H. M. Gerritsen (2012). “Cyber-Physical Systems: Concepts, Technologies and Implementation Principles,” in *Proc. of the Tools and Methods of Competitive Eng. (TMCE) Symposium*, pp. 19-36.

[11] James Manyika, et al., “Big data: The next frontier for innovation, competition, and productivity,” *McKinsey Global Institute (MGI)*, 2011. Retrieved on October, 31, 2014 from http://www.mckinsey.com/~/media/McKinsey/dotcom/Insights and pubs/MGI/Research/Technology and Innovation/Big Data/MGI_big_data_full_report.ashx

[12] Jianhua Shi, Jiafu Wan, Hehua Yan and Hui Suo (2011). “A Survey of Cyber-Physical Systems,” in *Proc. of the IEEE Int. Conf. on Wireless Communications and Signal Processing*, pp. 57 -59.

[13] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, “System Architecture Directions for Networked Sensors,” ASPLOS, November 2000.

[14] Kevin Ashton (2009). “That ’Internet of Things’ thing”, *RFID Journal*. Vol. 4, Issue 5, pp. 23 -24.

[15] Mark Roberti (2010). “The Internet of Things Revisited,” *RFID Journal*, Vol. 5, Issue 2, pp. 74 -75.

[16] Misra, Sanjay, and Emmanuel Eronu (2012) “Implementing Reconfigurable Wireless Sensor Networks: The Embedded Operating System Approach,” *Embedded Systems - High-Performance Systems, Applications and Projects*, *Intechopen*, Chapter 11, pp. 221-232.

[17] Mohammad Jamshidi (2011). “System of Systems Engineering: Innovations for the 21st Century,” *John Wiley &Sons*, vol. 58, pp. 101 -102.

[18] Norbert Wiener, “Cybernetics,” *Bulletin of the American Academy of Arts and Sciences*, vol. 3, no. 7, pp. 2-4, 1950.

[19] Peter Marwedel (2010). Embedded System Design, *Springer*, 2nd Edition, pp. 254 – 256.

[20] President's Council of Advisors on Science and Technology (PCAST), “Leadership under Change: Information Technology R&D in a Competitive World,” 2007. Retrieved on June 25, 2014, from http://www.nitrd.gov/Pcast/reports/PCAST-NIT-FINAL.pdf

[21] P. Mell and T. Grance (2009). “The NIST Definition of Cloud Computing,” *Recommendations of the National Institute of Standards and Technology*, pp. 28.

[22] Radhakisan Baheti and Helen Gill (2011). “Cyber-physical systems,” *The Impact of Control Technology, IEEE*, pp. 161-166.

[23] Ragunathan (RAJ) Rajkumar, Insup Lee, Lui Sha, and John Stankovic (2010). “Cyber-physical systems: The next computing revolution,” in *Proc. of 47th IEEE/ACM Design Automation Conf.*, pp. 731–736.

[24] Ramon Caceres and Adrian Friday (2012). “Ubicomp Systems at 20: Progress, Opportunities, and Challenges,” *Pervasive Computing, IEEE*, vol.11, no.1, pp.14-21.

[25] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic (2009). “Cloud computing and emerging IT platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility,” *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–611.

ISSN 2229-5518

[26] Sang C. Suh, U. John Tanik, John N. Carbone and Abdullah Eroglu (Editors), "Applied Cyber-Physical Systems," *Springer*, 2013.

[27] Sebastian Engell (2014). "Cyber-physical Systems of Systems – Definition and Core Research and Innovation Areas". Towards a Sousa-Poza, A., Kovacic, S. and Keating, C. "System of Systems Engineering: An Emerging Multidiscipline", Int. J. System of Systems Engineering, Vol. 1, Nos. 1/2, 2008, pp.1–4.

[28] Syed Gilani (2009). "The Promise of M2M: How Pervasive Connected Machines are Fueling the Next Wireless Revolution," White Paper, *Mentor Graphics*. Pp. 15 -16.

[29] S. Russel and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice-Hall, 1995. Pp 45-46.

[30] Volkan Gunes, Steffen Peter, Tony Givargis, and Frank Vahid (2014). " A Survey on Concepts, Applications, and Challenges in Cyber-Physical Systems." *KSII Transactions on Internet and Information Systems*. Vol. 8, no. 12, pp. 4243 – 4248, 4254 - 4255.

[31] William Bolton (2013). "Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering," *Pearson Education*, 5th Edition, pp. 172 – 173.

[32] Yen-Tsung Pan and Men-Shen Tsai (2009). "Development of a BDI-based Intelligent Agent Architecture for Distribution System Restoration Planning," Proc. o*f the 15-th International Conference on Intelligent Systems Applications to Power Systems*, Curitiba, Brazil, pp. 63 – 65.