International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:9, No:9, 2015

A Middleware Management System with Supporting Holonic Modules for Reconfigurable Management System

Roscoe McLean, Jared Padayachee, Glen Bright

Abstract**—**There is currently a gap in the technology covering the rapid establishment of control after a reconfiguration in a Reconfigurable Manufacturing System. This gap involves the detection of the factory floor state and the communication link between the factory floor and the high-level software. In this paper, a thin, hardware-supported Middleware Management System (MMS) is proposed and its design and implementation are discussed. The research found that a cost-effective localization technique can be combined with intelligent software to speed up the ramp-up of a reconfigured system. The MMS makes the process more intelligent, more efficient and less time-consuming, thus supporting the industrial implementation of the RMS paradigm.

Keywords**—**Intelligent systems, middleware, reconfigurable manufacturing.

I. Introduction

THE frequency and unpredictability of changes in modern industry requires a manufacturing system capable of a rapid response to change [1]-[3]. This market-driven necessity led to a shift in the focus of manufacturing research towards the responsiveness of systems. Research into Reconfigurable Manufacturing Systems has been underway since it was introduced in 1999 in a paper by Koren et al [1] at the University of Michigan, driven by the need for production systems that are able to economically evolve according to changes in markets and products [1]-[5]. The paper by Koren et al has been the basis for the RMS paradigm and its 5 essential characteristics and others have elaborated on them [2], [3]. To add to *Modularity, Integrability, Diagnosability, Convertibility,* and *Customization*, a 6th characteristic has since been defined – *Scalability* [4]-[6]. RMS aims for a system which can respond to new market needs rapidly, economically, efficiently, and effectively. This design for reconfigurability is the central concept of RMS and makes RMS an economically attractive system for high variety and Mass Customization Manufacturing(MCM) [4]. The cited papers contain in-depth explanations of the six characteristics and are thus not elaborated upon in this paper.

Reconfigurable Machine Tools (RMTs) are an essential components of RMSs; they aim to combine the best attributes of the Dedicated Machine Tool (DMT), used in most high-volume manufacturing and the multi-axis Computer Numerical Control (CNC) machine, used in lower-volume industries [5], [6]. The RMT aims to have the ability to produce some finite subset of parts or operations within a family [6], while maintaining the ability to produce high volumes. RMTs provide this functionality by allowing software and hardware modules to be interchanged or shifted to a different position which provides broader functionality. For this paper, the combination machine configurations and their positions on the level of the factory floor is referred to as the *factory state* or *RMS state*.

Roscoe McLean is with the University of KwaZulu-Natal (UKZN) in Durban, South Africa (Phone: +27 31 260 1227; e-mail: 214584328@stu.ukzn.ac.za).
Jared Padayachee and Glen Bright is with the UKZN (e-mail: padayacheej@ukzn.ac.za, brightg@ukzn.ac.za).

II. Reconfiguration in the RMS Paradigm

There are many reasons for a RMS to evolve its configuration; changes are mostly market-driven with new competition and new products forcing production changes at a high frequency. A RMS should handle market changes with a rapid and cost-effective reconfiguration [1], [7]. The illustration below [8] (Fig. 1) shows an example of physical reconfiguration in a cellular system.

A RMS can evolve in a combination of the following ways in order to adapt to new requirements: machines can change position within the system or be removed from/added to the system, the physical configuration of the machines may be altered and/or the operational program can be changed.

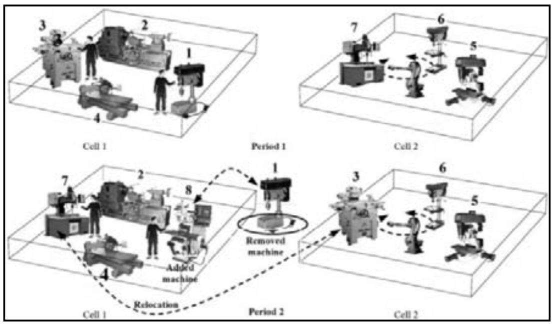

Fig. 1 A Simple Cellular Reconfiguration [8]

III. Enumeration of RMS States

The combination of possible changes in an RMS leads to a huge number of configurations; literature by Koren et al [2] gives a formula for the number of possible factory layout configurations, K (where N is the number of machines and m is the number of possible stages in which they can be arranged). Equation (1) gives the number if *m* is defined as *exact* and (2) if it is defined as *up to*:

$$K=\left(\frac{(N-1)!}{(N-m)!(m-1)!}\right) \tag{1}$$

$$K=\sum_{m-1}^{N}\binom{N-1}{m-1}=2^{N-1} \tag{2}$$

However, what is not documented is that each RMT also has its own finite number of configurations which changes the effective configuration of the RMS. The formula for the total enumeration of possible factory floor states must include the number of possible machine layouts on the factory floor *and* of the number of configurations of those machines themselves.

For the physical configuration of a modular RMT there are two parameters to consider, the number of reconfigurable aspects, j, and the number of possible configurations of each of these aspects, p_j. The number of configurations which are redundant or impossible, q, should also be taken into account and subtracted – this should be determined empirically. The software changes which are possible with no change in hardware, s, have to be found empirically and added, giving the formula for the number of configurations of a machine:

$$K_m=\prod_{i=1}^{j}(p_j+s_j)-q \tag{3}$$

Equation (3) above can be merged with the research by Koren and Shpitalni (1) and (2) to give the total amount of factory floor states, S_{ff} :

$$S_{ff}=K\prod_{i=1}^{N}K_{m_i} \tag{4}$$

It is clear from the above equations that the number of possible factory floor layouts is huge, which illustrates the power of the RMS paradigm in dealing with uncertainty and customization through a physical and control reconfiguration.

IV. Gaps in the Current Literature

RMSs require physical and software reconfiguration in order to handle the required operational changes. Industrial control and planning uses levels of software, Enterprise Resource Planning (ERP) and Manufacturing Resource Planning (MES) software, supported by engineers using knowledge and simulations, make reconfiguration decisions and produce production plans [9]. These need to be passed down to the rest of the system [8]. Once the machine moving and tool/module changes have taken place the decision makers need real information on the new factory floor state in order to implement the new operations plan to be passed to the controllers. This process, when done manually is time consuming and may be prone to errors. Speeding this up can drastically reduce ramp-up time, a core goal of the RMS paradigm [1]. It is the goal of this supporting technology to speed this up. The developed technology aimed to handle three particular problem areas: Communication, State Detection and the Re-establishment of Control of Heterogeneous Entities.

Communications had to handle the heterogeneous nature of a RMS, while providing fast reconnection after a reconfiguration. During a reconfiguration the machines are moved and the wireless nature of the MMS presented in this paper allows this movement to take place without the disruption of communications. Also, no communications rewiring is needed after the reconfiguration has taken place – shortening the ramp-up.

Currently in literature there is research on the intelligent reconfiguration of RMS [10], [11] and on middleware capable of handling the heterogeneous nature of the RMS controllers using software-based Holonic methods [14]-[16]. This literature does not, however, address factory floor state detection. Traditional SCADA, too, has no support for reconfiguration – it cannot detect the state of the factory floor and does not allow for reprogramming after a reconfiguration. Supporting technology is needed to create a link between the controllers and the decision makers.

Re-establishing control of heterogeneous entities is needed to complete the reconfiguration process. The current software-intensive method [12] routes data along the existing communication infrastructure of the RMS and is a traditional middleware system - the heterogeneity is handled using software which is configured to communicate with the heterogeneous entities. The need to gather factory floor state for the control elements slows the ramp up time of the RMS when done manually, inhibiting this core goal. The automation of this process would greatly increase the industrial practicality of the RMS paradigm.

V. The Thin, Hardware-Supported MMS

The new method for handling heterogeneity, the thin Middleware Management System (MMS) also acts as a state detection system which can communicate the factory floor state to the control elements. Using a hardware supported middleware system negates the need for a thick software-intensive middleware layer in the control system. A hardware module, the State Communication Module (SCM) is proposed as a solution to the shortcomings outlined in the previous section. This hardware-supported MMS differs from traditional middleware solutions, such as IceHMS [12], by shifting much of the burden to a hardware module; it also adds further functionality in the form of a Real-Time Location System (RTLS) and a method for machine configuration detection. The module system was developed to replace the complex middleware layer in the network with a Mechatronic hardware solution. A major motivator for this system is the simplicity achieved by using this hardware – the hardware supported system requires little debugging and is universal for machine controllers. The MMS presented here is simpler to implement than software-intensive methods and can be used as a replacement (including SCADA functionality) or as an

ISSN: 2517-9950
Vol:9, No:9, 2015

addition to a RMS containing standard SCADA because of the universality and modularity of the system. It stands alone as a system and thus is independent of the current installation, decreasing overall system cost.

The concept used a microcontroller module with wireless communication, a bit-wise communication link with the machine and position detection capability. The module attached to the machine and provides three main functionalities:

1) A method of gathering information on the factory floor after a reconfiguration operation, providing the decision making software with this information.
2) Passing software reconfiguration information from the decision makers to the machines.
3) Feeding supervisory data to the central controller via the wireless network during operation, to be displayed as a SCADA interface.

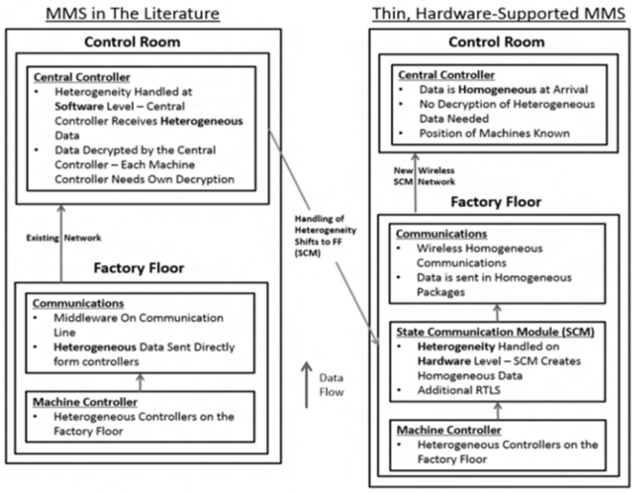

Fig. 2 Comparison of Traditional and Developed Middleware

Fig. 3 shows how this thin MMS differs from the traditional middleware in the literature.

The thin, hardware-supported MMS differs from the current proposed system by using hardware placed on the factory floor to handle heterogeneity and to provide wireless communication. This eradicates the need for complex software which must decrypt the heterogeneous data by using the modules to process the information and send the data in a homogeneous manner understood by the central controller. This allows the system to have no communication wiring, aiding mobility and ramp-up speed. In addition to replacing the need for the complex software, the thin, hardware-supported MMS has configuration detection and real-time location capability to allow for state detection. The functionality of this design matches the needs of the supporting technology discussed in Section III.

A. Communication between the SCM and the Central Controller

The literature contains a large amount of research into multiple different communication techniques ranging from simple wired protocols to Internet Protocol (IP) based wireless methods [13]. Initially a comparison was made between wireless and wired communication, based on the data speed, reliability, reconnection time, reliance on current architecture and cost.

Wireless technology was chosen for the further development, primarily for the universality and non-reliance on current installations and, in addition, it is a *future technology* –envisioned to grow. It is beneficial to develop a new framework based on emerging technology. For the purpose of RMS specifically, wireless technology has the additional benefit of being portable – the modules can be connected when other wired communications have not been plugged in yet. Because the proposed MMS is envisioned to be implemented internationally and to be used by many engineers who are not licensed to use restricted wireless bands, it must be designed in an unlicensed band. The Zigbee® protocol was chosen for its ease of use in the development of prototypes.

B. Communication between the SCM and Heterogeneous Controllers

Communication between the machine controller and the state communication module was chosen to be done in a bit-wise fashion or over a simple communications protocol. Many industrial controllers are Programmable Logic Controllers (PLCs) and are capable of reading High/Low signals on input pins. However, not all are compatible with one other standard protocol without the addition of an add-on module. Other machine and cell controllers are also generally capable of binary reading, making the option of bit-wise communication practically universal and cost effective. In addition, although many commonly available microcontrollers are capable of the simple communication protocols, every MCU is capable of digital bit-wise binary signaling. Using this bit-wise communication may seem simple, but it is important to note that a 7-bit message can provide 128 different messages.

C. State Detection

The MMS is able detect the state of the factory floor using the SCM and is able to communicate it to the higher level entities in the system.

Detection of the Machine's Physical Configuration: The module gathers the physical state of the machine through communications with the machine controller. Research has been conducted [14] into modular control architecture which knows the state of the machine. In simpler cases this can be hard-coded into the controller. The SCM requests the configuration from the machine controller using a 7-bit signal and the machine controller can be made to respond with a 7-bit code giving its configuration. The SCM then sends this information to the MMS controller.

Detection of the Machine's Position on the Factory Floor: There had been much research into various indoor positioning or localization techniques, these pieces of modern research summarized in a thesis by Mautz in 2012 [15], a version this work was also published as a journal article [16]. Two broad options described in these works are realistic in the RTLS technique based on the requirement constraints for the RMS paradigm application of a RTLS. Either the detection technology must be accurate (±1cm) or a less accurate (±1m) technique could be used along with some Artificial Intelligence (AI) to give accurate data. The AI should be able to produce a realistic and accurate model based on less accurate data. It is clear that the RTLS technique needs to have a range capable of covering a factory floor (>50m).

Primarily for the reason of cost reduction, a radiofrequency signal strength method was used as a RTLS. The development was based on previous work on RTLSs based on the Zigbee® protocol, which was already chosen to be included as the wireless protocol.

RTLS Implementation: There are many methods of RTLS, most of which make use of some form of triangulation method, in the same way as the familiar GPS. The methods investigated all required some form of distance estimation between the modules and some fixed beacons in order to operate. Received Signal Strength Indication (RSSI) was chosen in the research. Wireless signal strength decreases exponentially with increased distance in a predictable fashion, so this was used as an indication of distance. RSSI was used, with the Zigbee® protocol in the literature [17], [18] and was implemented in the MMS.

Multilateration is a localization technique which can handle inaccurate distance estimations and produce accurate (<1m in the implemented system) positional estimations. The multilateration pseudo-code algorithm, implemented in two dimensions in this research, is shown in Fig. 2.

This method proved accurate enough for the purposes of the research, producing position estimations within a 350mm radius of the true position over 80% of the time and within 400mm 95% of the time.

```
for (each beacon) do
        request and receive beacon position as: (x_b, y_b)
        for (number for average) do
                request and receive RSSI ping
        end for
        find average RSSI
        find estimated distance according to RSSI prediction
        end for
draw Circles from to: (x − a)^2 + (y − b)^2 = p_j^2
find the intersection points of circles which are not diagonally opposite
remove points outside of the factory floor region
find average of valid intersections and store the estimated position
```

Fig. 2 Pseudo-code for the RTLS

D. The SCM-Based MMS and the Key Characteristics

The SCM was designed specifically as a mechatronic solution to fit into the RMS paradigm, thus it should apply all the key characteristics needed for good reconfigurability. The SCM provides *Integrability* by having the ability to be integrated with any existing or future machines, thus being compatible with new technologies, but allowing RMSs with older machine hardware to be catered for. The SCM can also be fitted to a broad variety of machine and mobile robot platforms and could even be carried by people, helping a large variety of units to be integrated into the RMS. The SCM aids *Customization* because it is designed to be universal and does not carry excess functionality. The wireless nature of the communication and the ability to operate on battery power affords the SCM the ability to communicate errors before the system is up and running, helping the *Diagnosability* of the RMS. The SCM is fully *Scalable*, with the ability to have as many units as is needed added to the factory floor – with any machine addition a SCM can be added, allowing the RMS to be scaled up or down. The ability to fit the SCM to any machine gives it *Modularity* in the RMS. The SCM is designed to speed up the ramp-up time of the RMS thus allowing the RMS to change at a higher frequency, thus the SCM directly aids the *Convertibility* of the RMS.

E. The Make-Up of the SCM

Fig. 3 shows the functions, in block form, which are needed to create a successful SCM. Each block represents a process which the SCM must handle to provide the functionality described in the preceding sections.

VI. Knowledge-Based System and Clustering

Knowledge-Based Systems (KBSs) are a simple but tested branch of basic artificial intelligence which use a symbolic and numeric *knowledge base* which contains data historic data and an *inference engine* to interpret that data [19]. In the case of the MMS this knowledge base was a set of known factory floor states, including the programming data, stored as objects in the object-orientated C# Central Controller.

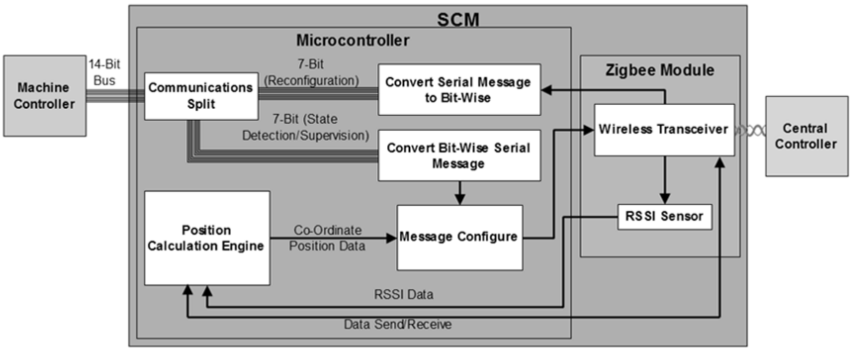

Fig. 3 Function Blocks of the SCM

Before the KBS receives positional data, a clustering algorithm converts it into a cellular model. The KBS for this thin MMS uses known machine states and cell configurations in its KB. These cell configurations contain the known combinations of machines in the cells and what machine configuration combinations are expected to be grouped together. The positioning intelligence works by first running a Density-based spatial clustering of applications with noise (DBSCAN) clustering algorithm; this can be simple because of the two-dimensional nature of the positional data provided by the SCMs. Once the clustering has been done the program assumes each cluster of machines to be a cell. These cells are then analyzed based on the machines present and their states. Each cell is then compared to the KB of cells – the cluster should match some known combination of machines. The matched cluster is then read by the system to be a cell formation. If the cell did not match any of the known cell layouts then the user is prompted to either run the positioning again or to add a new cell.

The KBS can use basic factory floor design data, rules and principles as discussed in previous research papers. Much value is added to the research through the development of the suitable AI (the KBS and clustering together) to make the more affordable RTLS more plausible in the RMS paradigm. Through this intelligence, raw co-ordinate data fed by the SCMs is converted into a factory floor model by the central controller. This data can also be used by the MES and ERP software. New model states can also be found this way (with some guidance) and added to the Knowledge Base (KB).

VII. MMS Design Summary and Description of Operation

The MMS is made up of SCMs, positional beacons and a central MMS controller (CMC). The CMC runs the middleware layer and communicates with the ERP/MES and SCMs. The SCMs gather their positional data by using the beacons and send this data to the CMC along with the configuration of their machine. The CMC converts this data into a factory floor model which is used by the MES and ERP in the preceding sections.

Fig. 4 shows the data flow and operation order of the MMS after a reconfiguration has taken place. The operation runs from after the physical reconfiguration has been done until the RMS is back online. A step-by-step description of operation after a physical reconfiguration is included after.

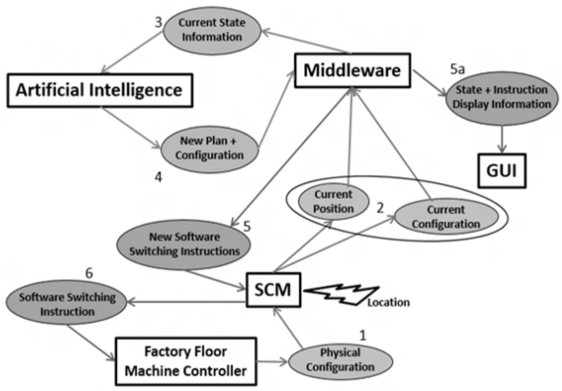

Fig. 4 Data Flow and Operation for Reconfiguration

1. Configuration from machine controller is read by the SCM and converted into a configuration message
2. The SCM determines its location and sends it, along with the machine configuration information in a message, to the CMC
3. The Middleware on the CMC sends the information to the artificial intelligence engine
4. The artificial intelligence creates a model including the production plan
5. The middleware sends the new software switching instructions to the SCM
5. (**a**) The CMC GUI displays the configuration, with machine positions
6. Once the physical reconstructions have taken place (according to GUI instructions) the software switching instructions are sent to the SCM
7. The bit-wise software instructions are sent to the machine controller.

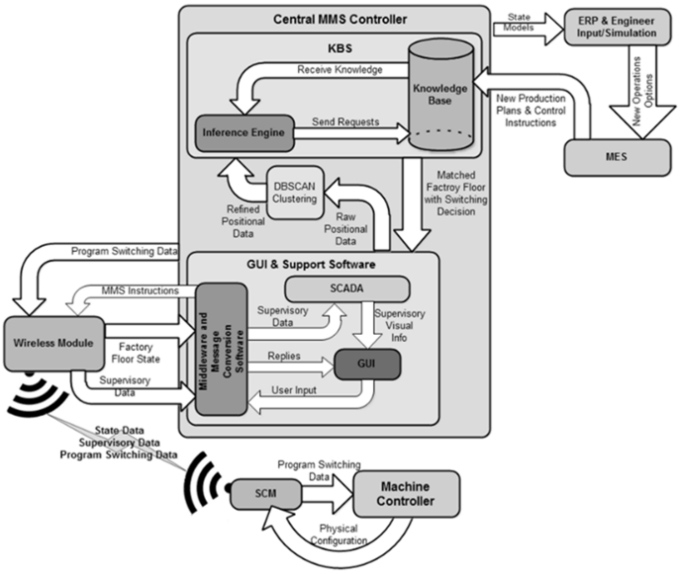

Fig. 5 Infrastructure of the MMS

VIII. Conclusion

Although not documented in this paper, the MMS's individual sub-systems were tested alone before they were combined into the complete technology. The combination of RTLS and clustering produced factory floor layouts which were accurate over 90% of the time. Given the new nature of the technology, this was a promising result. The KBS was able to assign programs in every case presented. The cost of the system is a fraction of other positioning techniques and it added the middleware ability.

The system itself was tested by implementing physical factory floors and running the MMS so that it switched between hypothetical programs as dictated by the layout. The testing was successful and well within the desired parameters of the research.

Although the system operated within the goals set, there is currently a limitation on the technology with the accuracy of the RTLS. RSSI data can be noisy and unreliable and some additional work on another form of distance measurement, such as Time Distance of Arrival (TDOA) which may provide a more robust and accurate RTLS.

The MMS fills a gap that existed in the research regarding the re-establishment of control of heterogeneous entities after a reconfiguration. This technique made use of the factory floor state to assign programs to machine controllers. This method looks specifically into the future where the system may include program assignment for materials handling systems and will need to use the true positions of the machines. It is hoped that this system will feed further research into the rapid ramp-up of a RMS after a reconfiguration.

Acknowledgment

Authors would like to thank Technological Innovation Agency (TIA) in South Africa who provided the funding for this research.

References

[1] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, et al., "Reconfigurable Manufacturing Systems," Annals of the CIRP, vol. 48, pp. 527-540, 1999.

[2] Y. Koren and M. Shpitalni, "Design of Reconfigurable Manufacturing Systems," Journal of Manufacturing Systems, vol. 29, pp. 130-141, 2010.

[3] M. G. Mehrabi, A. G. Ulsoy, and Y. Koren, "Reconfigurable Manufacturing Systems: Key to Future Manufacturing," Journal of Intelligent Manufacturing, vol. 11, pp. 403 - 419, 2000.

[4] B. Xing, G. Bright, N. S. Tlale, and J. Potgieter, "Reconfigurable Manufacturing System for Agile Mass Customization Manufacturing," in International Conference on CAD/CAM, Robotics and Factories of the Future, India, 2006, pp. 473-482.

[5] B. Xing, G. Bright, N. S. Tlale, and J. Potgieter, "Reconfigurable Modular Machine Design for Reconfigurable Manufacturing System," 2005.

[6] R. Landers, B. Min, and Y. Koren, "Reconfigurable Machine Tools," CIRP Annals - Manufacturing Technology, vol. 50, pp. 269-274, 2001.

[7] Z. M. Bi, S. Y. T. Lang, W. Shen, and L. Wang, "Reconfigurable manufacturing systems: the state of the art," International Journal of Production Research, vol. 46, pp. 967-992, 2008.

[8] N. Safaei and R. Tavakkoli-Moghaddam, "Integrated multi-period cell formation and subcontracting production planning in dynamic cellular manufacturing systems," International Journal of Production Economincs, vol. 120, pp. 301-314, 2008.

[9] A. Bajahzar, A. Alqahtani, and A. Baslem, "A survey study of the Enterprise Resource Planning System," in International Conference on Advanced Computer Science Applications and Technologies, Kuala Lumpur, Malaysia, 2012, pp. 246-252.

[10] B. Xing, F. V. Nelwamondo, K. Battle, W. Gao, and T. Marwala, "Application of Artificial Intelligence (AI) Methods for Designing and Analysis of Reconfigurable Cellular Manufacturing System (RCMS)," presented at the 2nd International Conference on Adaptive Science & Technology, Accra, Ghana, West AFRICA, 2009.

[11] L. Tang, Y. Koren, D. M. Yip-Hoi, and W. Wang, "Computer-Aided Reconfiguration Planning: An Artificial Intelligence-Based Approach,"

Journal of Computing and Information Science in Engineering, vol. 6, pp. 230-240, 2006.
[12] O. Roulet-Dubonnet, M. Lund, and A. Skavhaug, "IceHMS, a Middleware for Distrebuted Control of Manufacturing Systems," in Industrial Applications of Holonic and Multi-Agent Systems, 1 ed Berlin, Germany: Springer Berlin Heidelberg, 2013, pp. 95-105.
[13] D. Dickinson, "Industrial Wireless Technology," Phoenix Contact 2006.
[14] B. Xing, G. Bright, and N. S. Tlale, "Modular Mechatronic Control of Reconfigurable Manufacturing System for Mass Customisation Manufacturing," in International Conference on Competitive Manufactruing, Stellenbosch, South Africa, 2007, pp. 223-228.
[15] R. Mautz, "Indoor Positioning Technologies," Habilitation PhD, Civil, Environmental and Geomatic Engineering, ETH Zurich, 2012.
[16] R. Mautz, "Indoor Positioning Technologies," Sechsundachtzigster Band, vol. 86, 2012.
[17] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, "Weighted Centroid Localization in Zigbee-based Sensor Networks," in IEEE International Symposium on Intelligent Signal Processing, Alcala de Henares, Spain, 2007.
[18] S. Schwarzer, V. Martin, M. Pichler, and A. Stelzer, "Precise Distance Measurement with IEEE 802.15.4 (ZigBee) Devices," in IEEE Radio and Wireless Symposium, Orlando, USA, 2008, pp. 779-782
[19] R. G. Smith, "Knowledge-Based Systems: Concepts, Techniques, Examples," in Canadian High Technology Show, Ottawa, Canada, 1985.