Middleware Communication System for Fixture based Flexible Manufacturing Systems

N. Chiraga, A. Walker and G. Bright
Discipline of Mechanical Engineering, University of KwaZulu-Natal, Durban, South Africa

Keywords: Distributed Flow Control, Wireless Communication in Manufacturing, Middleware, Mass Customization.

Abstract: Today's factory floors are a distributed system of heterogeneous components. They exist a high degree of production customization to cater for this most factory floor systems are being adapted for customization. Most current manufacturing communication systems do not have a flexible architecture to serve the changeable and dynamic market needs thus presenting a research gap in flexible communication. This paper discusses a middleware communication system that allows for flexible optimal control of information flow in a fixture based production system. The architecture of the system aims to integrate seamlessly a heterogeneous factory environment, for autonomous monitoring and control of information on the factory floor.

1 INTRODUCTION

The manufacturing industry is currently experiencing a paradigm shift into the Fourth Industrial Revolution in which customers are increasingly at the epicentre of production (EFFRA, 2013). This revolution has prompted major research into factories of the future; forcing industries to re-examine their systems (Bloem et al., 2014). Schwab et al stated in the recent world economic forum stated that the Fourth Revolution is evolving at an exponential rather than linear pace (Schwab, 2016).The high degree of production customization and personalization requires a flexible manufacturing system that will rapidly respond to the dynamic and volatile changes driven by the market (Qiao et al., 2000). Mass customization has seen limited adoption, thus giving raise to research into the implementation of manufacturing systems that are highly responsive to these rapid changes in market demands. They is a gap in technology that allows for optimal flow of information and optimal manufacturing operations on the shop floor regardless of the rapid changes in fixture and part demands. The mass customisation manufacturing (MCM) paradigm has created a problem in manufacturing control implementations, as each customer has the potential to disrupt production operations resulting in downtime (Walker and Bright, 2013). Factory communication systems now also have to cater for the high degree of production customization and personalization.

It is essential to attain at least 99.99% uptime or better on the shop floor during these rapid changes (Qiao et al., 2000). Depending on the size and nature of business, indirect costs due to downtime can range from tens of thousands to hundreds of thousands of dollars (Qiao et al., 2000). A reliable advanced factory communication system architecture is critical for optimal network performance. A deep and efficient integration of information flow, information analysis, and customer input in the production network is necessary. There is a need to provide the right information at the right time and show the manufacturing decision maker how current conditions on the plant floor can be optimised to improve customized production output. Salvador et al. identified three fundamental capabilities determining the ability of a factory to mass-customize its offerings, i.e. solution space development, robust process design, and choice navigation (Salvador et al., 2009).

This paper aims to develop the robust process design capability in Factory Communication Systems; the capability to create, reuse, or recombine existing factory shop fixture resources to fulfil a stream of differentiated customer needs (Salvador et al., 2009). Flexible automation, process modularity, and adaptation are approaches that can be taken to develop robust process design. Flexible automation can be described as automation that is not fixed or rigid and can handle the customization products. While process modularity is the segmentation of

Chiraga, N., Walker, A. and Bright, G.
Middleware Communication System for Fixture based Flexible Manufacturing Systems.
DOI: 10.5220/0005972201460152
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1*, pages 146-152
ISBN: 978-989-758-198-4

existing organizational and factory shop resources into modules that can be reused or recombined to fulfil differentiated customers' needs (Salvador et al., 2009).

To facilitate the flexible communications between the high-level software systems Enterprise Resource Planning (ERP), SCADA and Manufacturing Execution System (MES)) and users (control and industrial engineers), and the heterogeneous factory floor environment middleware technology is needed. This paper details an advanced Factory Middleware Communication System (FMCS) that uses a middleware communication system to allow for flexible control and information exchange in a factory environment driven by the dynamic customer needs for production execution. The aim of this study was to seamlessly integrate manufacturing floor communication with the customers' decisions in product configuration.

2 AN OVERVIEW OF THE FACTORY MIDDLEWARE COMMUNICATION SYSTEM (FMCS)

Traditional Supervisory Control and Data Acquisition (SCADA) systems collect data from various sensors nodes deployed in remote locations and then transmit it to a central controller which then manages and controls this data ("SCADA").While the Manufacturing Execution System(MES) tracks and documents the transformation of raw materials through finished goods("Manufacturing Execution System"). The Enterprise Resource Planning (ERP) tracks the factory's resources, i.e. raw materials supply and production capacity ("Enterprise Resource Planning"). This Factory Middleware Communication system is an advanced manufacturing strategy aiding the SCADA system, MES and ERP; and equipping them with intelligent handling capabilities for the rapidly changing customer needs, which in turn result in the production processes experiencing rapidly varying fixture and part demands. The figure below illustrates where the developed system exits in a manufacturing pyramid.

Figure 2 shows the general overview of the system. The system solution copes with a demand characterized by at least a subset of the following properties: low-mid demand volumes, mid-high variety of the part mix, short product lifecycle, and mid-high customization. The following sections explain each module in the system.

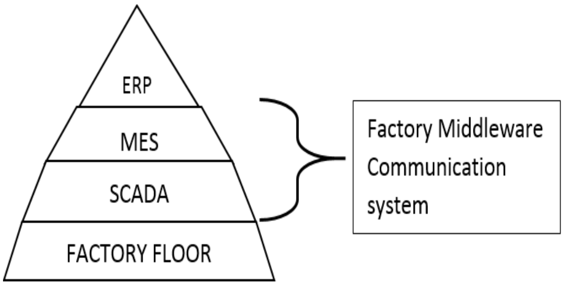

Figure 1: Manufacturing production levels.

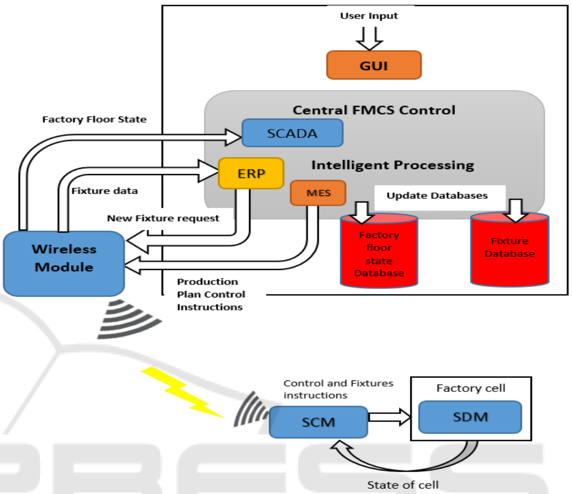

Figure 2: Factory Middleware Communication System (FMCS) Overview.

2.1 State Detection Module

The FMCS operates using state detection modules (SDM) attached to each factory cell to determine the state of each cell at any given time on the production floor. The distributed SDMs; microcontroller-based modules form a sensor network, with state detectation capabilities. In this research, we are primarily interested in the information presented at each factory cell. The aim is to create an information intense environment rather than just transmit instructions. Information is the reduction of uncertainty; it gives meaning and context on the state of each cell. In information theory Shannon describes entropy as the average value of information contained by a system (Lesne, 2011).

Suppose that a discrete variable X, has n possible outcomes; $x_1, x_2, x_3, \ldots\ldots, x_n$ the probability for each outcome being p_i. Shannon entropy is defined as:

$$E(X) = -\sum_{i=1}^{n} p_i \log_2 p_i \qquad (1)$$

Where:

$p_i \geq 0$ And $\sum_{i=1}^{n} p_i = 1$

The variable X denotes a system, which is the cell in our case, $x_i$and p_i $(i = 1,2, \ldots, n)$ are its n possible states and their probabilities in the cell respectively. The amount of information needed to describe the cell, X is $E(X)$ i.e. information entropy (Lesne, 2011). Considering a factory cell X on a factory floor, six states can be defined;

i. Optimal working state
ii. Running slow state
iii. Idle state
iv. Scheduled maintenance state
v. Unscheduled maintenance state
vi. Downtime state

Based on the probabilities of each state, using Shannon Theorem the entropy for each cell can be calculated and hence the overall entropy of the entire information system.

2.2 The State Communication Module (SCM)

The SCM facilitates the communication link between the state detection modules and the central control computer, which runs the Central FMCS Control (CFC) software.

Communication techniques range from wired to wireless communication protocols. Publications have shown emerging work ushering in the wireless communication protocols into industrial networks. (Buda et al., 2010). To keep up with technological trends wireless technology was used for the development of the communication network. Wireless technology offers flexibility; modules can be introduced to the system in a plug and play manner without having to worry about re-wiring. Factory propagation environments however, have many metallic surfaces and moving objects, which can be characterised as harsh conditions that can affect the operational quality of radio based communication systems (Buda et al., 2010). Wireless communication is implemented with ZigBee wireless protocol, a well-adapted technology for industrial applications. ZigBee operates at 2.4GHz frequency, which lays in the bandwidth of frequency not affected by the interference in factories. (Buda et al., 2010). ZigBee is based on an IEEE 802.15.4 standard. XBee Series 2 are used for the Zigbee wireless communication. The SCMs help us achieve process modularity; state information is continually transmitted wirelessly to the central control.

2.3 Central FMCS Control (CFC)

2.3.1 Middleware Communication Layer

Today's factory floors are a distributed system of heterogeneous components. There exist a wide range of applications and system platforms, making heterogeneity in factories predominant. Each SDM stands out as a heterogeneous component in a cellular configuration. This is where middleware technology comes in; middleware masks heterogeneity on the factory floor and allows for flexible control and exchange of information. Using a thick software-intensive middleware layer negates the need for a hardware supported middleware system in the control system. The objective is maintain optimal manufacturing operations on the shop floor during model definition changes of customer-desired products. Internet Communications Engine (Ice) is a modern object-oriented middleware platform that enables you to build distributed applications with minimal effort (Roulet-Dubonnet et al., 2013).

With Ice, there is no need to worry about details such as opening network connections, serializing and deserializing data for network transmission, or retrying failed connection attempts. Its features include location independence; the client does not need to know of the specifics of the target object's address, platform independence; multi-vendor manufacturing devices can integrate seamlessly, programming language independence; does not matter if client and server processes are implemented with the same language or different languages. The middleware layer works as the workflow manager, the nervous system of the network created allowing for interoperability and seamless flow of information.

Figure 3 shows the middleware client; sever structure (Roulet-Dubonnet et al., 2013).The CFC acts as the client and the SCMs as the server.

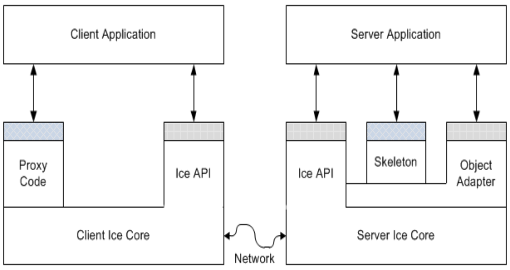

Figure 3: Ice Client and Server Structure.

When a client i.e. central control invokes an operation, the following steps take place:

i. Locates target object i.e. the SCMs located at each cell in the network

ii. Activates the server application in the SCMs, if the server is not already running

iii. Transmits any arguments for the call to the object via the ZigBee wireless network

iv. Waits for request to complete

v. Returns any out parameters or return value to the client when a call completes successfully. The return parameters in our case is the state information gathered by the SDMs, fixture information from the fixture reconfiguration system and machine orientation from the reconfigurable machine system.

vi. Returns an exception to the client when a call fails

2.4 Intelligent Processing and Program Assignment in the Central Control (CFC)

The CFC is the middleware and intelligence layer of the system, the workflow manager. It continually polls for information from the factory floor and stores it into a database. Based on the factory floor state, fixture configuration, availability, and the machine configuration an order is processed and control instructions are assigned to the work cells. Through this intelligence of raw factory floor state data fed by the SCMs, an information-based configuration is set-up. Sequences of job arrive, entering the shop floor input buffer and are logged in through the graphical user interface and order parameters are noted according to the customers' decisions in product configuration. Jobs are dealt with using the MIMO queue concept. The product specifications make up the procedural rules for the workflow decision tree. The ability to effectively communicate the factory floor state and the characteristic response of the system to a new product configuration are of utmost importance in this research. The EPR keeps track of the available raw materials i.e. fixture database on the factory, if a new product configuration requires a new fixture, instructions are sent to the 3D Printing cell to produce a new fixture. Once the fixture production is done, the fixture database is updated and the production plan and control instruction are sent to each cell on the factory floor. Figure 4 below explains the procedural steps in the intelligent process up until the order program is assigned to the factory floor for manufacture.

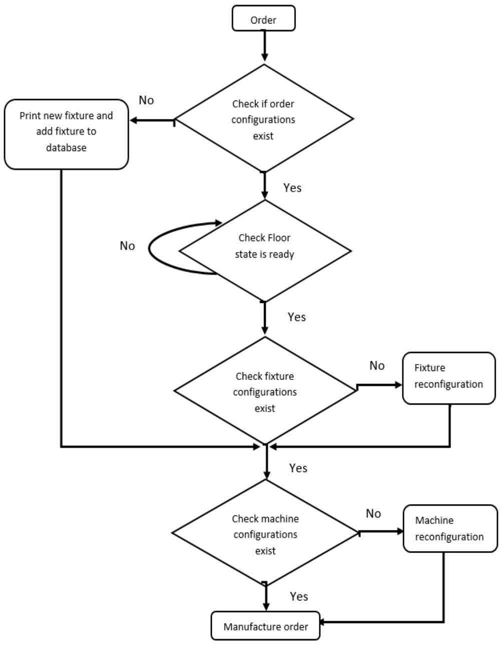

Figure 4: Procedural steps in the intelligent processing of an order.

3 EXPERIMENTATION ON THE SYSTEM

The system was implemented and tested in the UKZN Manufacturing and Mechatronics Laboratory. The laboratory layout depicts a factory floor environment; it is made up of a cellular configuration. The cellular make-up of the configuration includes; a Material handling cell, 3D Printer cell, Assembly cell, manufacturing cell and Quality control cell. State detection modules as well as state communication modules were placed in each cell, forming a star network. The CFC software layers links up with two pre-existing systems in the laboratory i.e.; the machine reconfiguration system and the fixture reconfiguration system. The machine reconfiguration system provides the CFC with the physical configuration of the machines on the factory floor and allows the machines to be reconfigured accordingly. While the fixture reconfiguration system provides the current fixture configuration at any given time and again allows for fixture reconfiguration.

The flexibility of this system is random-order, there are substantial variations in part configurations, new part designs are continually being introduced to the system. It is a recognition that a factory cannot do everything; it must limit its options to a certain scope of products activities in which it can best compete.

Table 1: Hypothetical initial scope of product variety.

Hard Product Variety	Soft Product Variety
Product 1	5 variations
Product2	3 variations

3.1 Aim

The experiment associated with the situation described above aimed to prove that it was possible, using the technology developed in this research, to rapidly respond to customer demands by intelligently and effectively communicating information on the factory floor. It aimed to increase the performance of downstream production instructions flow fed from parallel upstream flow of information on the factory state.

3.2 Method

To verify the success of the system, the system needed to respond rapidly to both the initial scope on product variations as well as the randomly introduced new customized product variations. In addition, the system needed to facilitate the appropriate fixture and machine configurations and assign program instruction to the factory floor.

The following method was followed:

1. Use the Central FMCS Control software to initiate a factory floor scan.
2. Input order's product configuration into the system using the graphical user interface. Randomly input new customized product specifics.
3. Check the response time, the time between lodging of order into system and dispatcher of order program instruction.
4. Repeat steps 2-3, varying the order specifics each time.
5. Check fixture database updates in the case of a new product configuration.

3.3 Results

As shown in Table 1, the system managed to successfully respond to new customized orders, seamlessly integrate systems and communicate information on the factory floor. The FMCS had an 80% success rate.

3.4 Discussion of Results

The experiment was successful in identifying the states of each cell at any given time. Each process, from the initialisation of the program to when the assignment of program production control instructions took place in real time with delays of less than 3mins. The system was able to obtain real time responses for the factory states data. The processes where a new product configuration was introduced responded within a 2min to 5min response time. Extra time was needed for the 3D printing of the fixture. The cellular configured SCMs were able to communicate within a radius of up to 1,2km approximately 99% of the time. This process is still very quick in industrial terms and is far more time and labour efficient than manual methods. The product variety was quantified using the equation given below.

$$P = \sum_{j=1}^{P_1} P_{2j} \tag{2}$$

Where P refers to the different product designs or types that are produced in the factory.P_1 the number of distinct products lines, hard product variety and P_2 the number of number of models in a product line, soft product variety. Subscript j identifies the product line (Groover, 2008).

Table 2: Results of FMCS Test.

Order #	Run #	Identified Factory floor state correctly	Communicated with Fixture reconfiguration system correctly	Communicated with machine reconfiguration system correctly	New Fixture Produced	Assigned Program instructions correctly	Time	Result
1	1	Yes	Yes	Yes	N/A	Yes	8:30	Pass
2	1	Yes	Yes	Yes	Yes	Yes	9:45	Pass
3	2	Yes	Yes	Yes	No	N/A	11:18	Fail
4	1	Yes	Yes	Yes	N/A	Yes	11:30	Pass
5	1	Yes	Yes	Yes	Yes	Yes	14:03	Pass

The graph below shows how the product variety of a factory overtime using the Factory Middleware Communication System (FMCS).

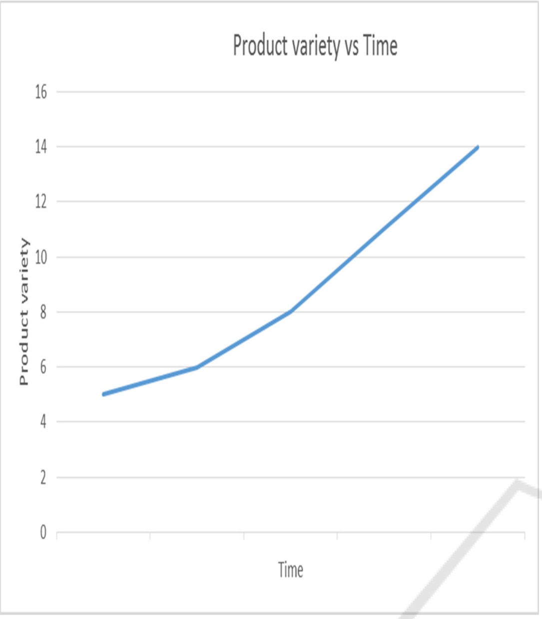

Figure 5: Graph of Product variety versus time.

From the graph, we can see that the system increase the scope of product variety of a system, making it well equipped to handle volatile changes driven by the market.

3.4.1 Performance and Scalability of Middleware Software

Further test were done to measure the performance of the middleware layer. A fundamental measure of middleware performance is latency. Latency is the time it takes for a two-way operation to be invoked between a client and server and obtain the results of the operation (Roulet-Dubonnet et al., 2013). When we run the client and server on the Core i7 machine, with the client and server running on different machines and communicating over a network. The latency is 2,500 messages per second (400µs per message).

4 CONCLUSIONS

The flexibility of the system means that users have an almost limitless expandability and engineers can adapt and upgrade the system's features and capabilities to meet immediate and future requirements. The middleware communication system allows for flexible control and information exchange in a heterogeneous factory environment driven by the dynamic customer needs for production execution. The performance of downstream production instructions flow fed from parallel upstream flow of information on the factory state that were increased with the use of this system.

REFERENCES

European Factories of the future assosiation EFFRA (2013). Factories of the future.

Bloem, J.,Van Doorn, M., Duivestien, S.,Excoffier, D., Mass, R., and van Ommeren, E.,(2014). The Fourth Industrial Revolution. In:Sogeti. Vint.sogeti.com/[Accessed 5 Mar. 2016].

Schwab, K., (2016). The Fourth industrial revolution. In: World Econimic Forum. http://www.weforum.org/ agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond.

Qiao, G., Lu, R., and McLean, C., (2000)"Flexible Manufacturing System for Mass Customization Manufacturing"

Walker, A., and Bright, G., (2013) Distributed Control Synthesis for Manufacturing Systems using Customers' Decision Behaviour for Mass Customisation.

Salvador, F., Martin, P., and Piller, F., (2009) Cracking the Code of Mass Customization. In MIT Sloan Management Review, Massachusetts.

"SCADA". Wikipedia. N.P., 2016. Web. 17 Mar. 2016.

D. Bailey and E. Wright, "Practical SCADA for Industry", (2003).

R. J. Robles, M. -k. Choi and T. -h. Kim, "The Taxonomy of SCADA Communication Protocols", Proceedings of the 8th KIIT IT based Convergence Service workshop & Summer Conference, Mokpo Maritime University (Mokpo, Korea), ISSN 2005-7334, pp. 23.

M Choi, "Wireless Communications for SCADA Systems Utilizing Mobile Nodes", International Journal of Smart Home Vol. 7, No. 5 (2013), pp. 1-8.

"Manufacturing Execution System". Wikipedia. N.p., 2016. Web. 17 Mar. 2016.

"Enterprise Resource Planning". Wikipedia. N. P., 2016. Web. 17 Mar. 2016.

Lesne, A., (2011) "Shannon entropy: a rigorous mathematical notion at the crossroads between probability, information theory, dynamical systems and statistical physics," Pierre and Marie Curie University, France.

Buda, A., Schuermann, V., Wollert, J., (2010) "Wireless Technologies in Factory Automation," University of Applied Sciences Bochum Germany.

Roulet-Dubonnet, O., Lund, M., and Skavhaug, A., (2013), "IceHMS, a Middleware for Distributed Control of Manufacturing Systems," in Industrial Applications of

Holonic and Multi-Agent Systems, 1 ed Berlin, Germany: Springer Berlin Heidelberg, pp. 95-105.

Mikell P. Groover. (2008). Automation, Production Systems and Computer-Integrated manufacturing, New Jersey: 3rd edition, Pearson International Edition.p. 55.