To Appear in Proceedings of HolonMas00, London, September, 2000

A Holonic Component-Based Approach to Reconfigurable Manufacturing Control Architecture

Jin-Lung Chirn, Duncan C. McFarlane
Institute for Manufacturing, University of Cambridge
Mill Lane, Cambridge, CB2 1RX, UK

Abstract

Holonic Manufacturing Systems have emerged over the last seven years as strategy for manufacturing control system design. A new approach called Holonic Component-Based Architecture (HCBA) to establish a manufacturing control system as to cope with rapid changes in manufacturing environment is presented in this paper. Intelligent building blocks in terms of resource and product are proposed to dynamically form a virtual controller via a computer network, and to perform co-operative control execution and diagnosis operations. This concept enables the design, operation and maintenance of the manufacturing controller to be performed in a distributed manner, which can increase the agility and responsiveness of an integrated system. This flexible structure has been implemented in a robot assembly cell to show its plug-and-play capability via an Internet-based infrastructure.

1. Introduction

The theme of this study is to develop a reconfigurable manufacturing control architecture which can adapt to continuous changes in the manufacturing environment. Essentially, past research and practice associated with these problems has primarily focused on "operational performance issues", ignoring aspects related to the characterisation and analysis of the system structure and its effects on the system (logical) behaviour [7]. Such an attitude has been possible in the past, since the presence of the human operators on the manufacturing shop floor has allowed for the last-minute ad hoc solutions to make the shop-floor production activity meet. Automated production, and rapid change requirements however, cannot not afford this luxury.

A systematic reconsideration of past practices is detailed in this study, and a new integrated architecture is then developed. Its scope is mainly focused on shop floor for manufacturing execution and diagnosis. With the aid of advanced computer communication technology, this approach has been applied to the design of an Internet-based manufacturing controller which can implement the concept of distributed design, integration, operation and maintenance. Furthermore, this research result encourages the future integration of manufacturing operations with an *E-commerce* environment.

2. Research methodology

The research methodology of this study is mainly derived from component-based development (from the software perspective) and holonic manufacturing systems (from the manufacturing perspective).

Component-based development (CBD) is associated with a shift from statement-oriented coding to system building by plugging components together. It is now heralded as the next wave to fulfil the promises that object technology could not deliver [9]. The idea can be traced back to the introduction of the concept of the software integrated circuit (SIC) [3] to develop and package software components for later use, just as hardware components are packaged for convenient use in integrated circuits. The CBD approach focuses much on developing reusability and reconfigurability in view of the architecture rather than the individual software modules. This idea provides valuable experience when examining the underlying system components, interface and infrastructure for a manufacturing control architecture, and can provide a guide when constructing a scalable and extendible system from the view of a bottom-up approach.

An approach called *holonic manufacturing systems (HMS)* is introduced. This approach is based on the concept of a *holon* as originally coined by Koestler [6]. This system concept was later developed in a manufacturing context by Suda [8] and Christensen [4] as a means of providing a building block or "*plug and play*" capability for developing and operating a manufacturing system in the factory of the future. The ultimate aim of this approach is to develop an architecture for highly decentralised manufacturing systems, built from a modular mix of standardised, autonomous, co-operative and intelligent components, in order to cope with rapidly changing environments.

The manufacturing control architecture proposed in this study is the so called *Holonic Component-Based Architecture (HCBA)*, which is derived from the concepts of CBD and HMS. CBD provides a guideline for HCBA when constructing a reconfigurable structure from the basic building block, whereas HMS identifies the underlying attributes of the building blocks within a dynamic architecture. Although this structure of CBD provides a potential advantage for ease of plug-and-play capability, the integration of components may encounter problems in dynamic and changing manufacturing environments. The concept of HMS, which identifies the requirements of the behaviour of a system component in a very dynamic environment can support that of CBD so as to enhance the control architecture. In this study, these two concepts are combined to derive the concept of a *holonic component*, by which a flexible architecture can be dynamically formed and reconfigured.

3. Holonic component-based architecture

In this section, the development of a holonic component-based architecture (HCBA) is outlined, including the introduction of different types of holonic components, the integration of these components, and the migration of a conventional control architecture to HCBA.

3.1 Identification of system components

According to the distributed and bottom-up design concept from CBD, the composition of the basic units in the HCBA (i.e. holons or holonic components) is identified initially according to the fundamental elements of a physical plant. The physical objects of a manufacturing plant can be categorised into two general groups in terms of their properties. One is the resource which performs the manufacturing operations and the other is the product which accepts the manufacturing treatments.

The *resource component* or *resource holon* is a self-contained system component which can perform operations on works in process (WIP), such as fabrication, assembly, transportation, and testing. Besides the visible physical part, a resource component contains an invisible control part which can perform its operations, decision-making and communication ability by aid of its local database. Similarly, the *product component* or *product holon* also contains a physical part and a control part. The physical part may include raw material, parts and pallet/fixtures. A control part may contain routing control, process control, decision-making and production information. The classification of these system components provides the potential to reconfigure a control system readily. Further explanation can be found in [1, 2].

3.2 Operation of HCBA

The control software of HCBA is different from that of conventional architecture in structure. The traditional design results in a single bulk of software which is usually provided from a unique system provider. Whereas this new approach establishes an architecture which can dynamically form a virtual manufacturing controller across the computer network according to the immediate requirement. Furthermore, the system components can be provided from different sources to implement the concept of concurrent and distributed design and integration. A diagram showing a possible operation is depicted in Fig. 1.

Initially, a pool of separated and unorganised resource software components in HCBA may be supplied by machine providers or equipment designers in the designer centre. These components have a one-to-one link with their associated physical equipment in the factory. Thus, a resource holon contains these two main parts: a software part in the computing environment for control and decision-making, and a physical part in the physical plant for actual fabricating. As a result, the complete decoupled property of a resource holon can be obtained. The formation of the resource pool is called *static integration* because there is no interaction between holons at this stage. Hence, a resource holon can be replaced or added readily, without causing a global effect in this architecture.

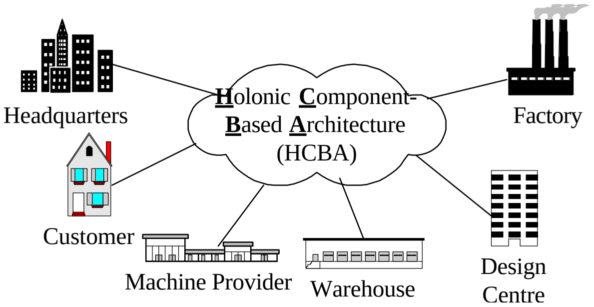

Fig. 1. Operation of HCBA

Dynamic interaction between holons in HCBA is only initiated once product holons are introduced. A product holon which may be provided by the product/process designers in the design centre, or OEM customers, carries a detailed process plan. The bill of material (BOM) may pass to the logistic team in the factory for them to prepare the associated raw material. The software part of the product holon can then be plugged into the architecture. It then tries to make use of resource holons by proceeding with a series of negotiations. This stage is called *dynamic integration*. The holonic controller returns to the state of static integration if all product holons have finished their tasks; this is because no interaction is now generated.

3.3 Migration approach of existing architecture

An approach to apply the principle of HCBA to an existing CIM-based hardware -architecture is now explained. The hierarchical structure of CIM is depicted in Fig. 2, and is composed of machine, cell, and factory levels. Different functions are provided in each level to control the manufacturing activities in the plant. The machine level at the bottom performs real-time control and operation. Execution and monitoring functions are situated at the cell level to co-ordinate the lower level operations. At the factory level, the function is more information-oriented where - planning and scheduling functions also run to decide a global but not necessarily a real-time control strategy. According to the different requirements of timing, and data, different communication devices are employed between each of the two levels to integrate the CIM architecture. This architecture encourages each level to become more centralised and horizontally integration. Basically, this structure is not compatible with the ideal HCBA infrastructure as we will now discuss.

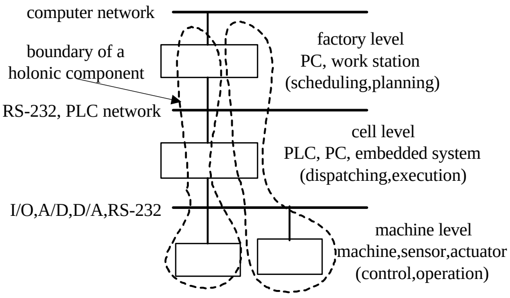

Fig. 2. Mapping of Holons to the CIM Architecture

Sections of holons or holonic components in HCBA can be easily identified at the bottom level of CIM. However, the functional modules of CIM at higher levels cannot be fitted in the holon framework directly. Bottom-up development is employed to identify the boundary of a holon beginning from the machine level. A holon therefore can grow up gradually from the bottom to higher levels in CIM by investigating the necessary functions that a holon should possess. Fig. 2 shows the boundary of a holonic component in the CIM architecture. A holonic component has broken the traditional level delimitation and straddles traditional architecture levels. The functional modules of CIM, such as planning and scheduling, no longer exist. These functional modules are decomposed and re-organised across a number of separate holonic components. Although no single functional module exists to control the integrated manufacturing operations, overall control behaviour can be managed by co-operation between holonic components. Holonic design in fact reinforces the vertical integration of an existing CIM-based architecture (within the holon), while also providing a more physically-oriented distribution of control functions and while also supporting horizontal integration (between holons).

A software infrastructure has been developed to support the migration of a traditional manufacturing control architecture into the HCBA. Two major communication mechanisms: the *Black Board System* (BBS) and the *Message Broker* (MB) are introduced as the communication agents to integrate holonic components within the software environment. The detailed introduction to the migration approach and software infrastructure is given in [2].

4. Manufacturing co-ordination in HCBA

HCBA is inherently distributed in terms of system structure and design philosophy. There is no central mechanism to co-ordinate the overall factory behaviour. Under the constraint of the distributed architecture, an individual holonic component should possess distributed decision making and communication ability to implement the required global behaviour with the aid of the other holonic components. In this section, the development of co-ordination in manufacturing execution and diagnosis respectively is presented.

4.1 Co-operative Execution

As shown in Fig. 3, a holonic manufacturing system initially consists of a pool of separated and unorganised resource holons. The co-ordinator component of a product holon is also situated in this system waiting for the manufacturing orders. Typical information in an order includes the quantity of products to be produced, the due date for delivery and the specification, such as quality criteria and variety type. After a new order is released, the co-ordinator component updates its production plan immediately, and dynamically generates WIP (work in process) agents from time-to-time to gradually accomplish the tasks of the order. A dynamic interaction is therefore generated.

The mission of a WIP agent is to escort a unit of the physical part to pass through the manufacturing plant by following a given goal which is proposed by the co-ordinator of the product holon. The function of the co-ordinator is to track the status of WIPs within the plant, as provided by the WIP agents it creates, so that the progress of orders can be monitored. The co-ordinator component manages the group behaviour of a batch of productions, while WIP agents control the detailed execution of each physical part. Their co-operation constitutes the behaviour of a product holon.

A WIP agent is responsible for negotiating with the resource community to decide the best manufacturing treatment during its life cycle. Requirements, such as high quality or high throughput, are decided by negotiation with resource holons. Different methods of request of a WIP agent to resource holons are described below:

(1) Single and fixed routing: Only one specific resource is available and pre-set in a WIP agent's process plan. A WIP agent will send its request to that resource directly and wait for the agreement of that resource.

(2) Flexible routing table: More than one resource is available and pre-set in a WIP agent's process plan. A WIP agent will send its request to all the possible

resources and wait for bids from them. A WIP agent will choose the most suitable bid and execute it.

(3) Resources are unknown: The resource information is not mentioned in a WIP agent's process plan. In this situation, a WIP agent will broadcast its request to all resources and then wait a certain time for the bids from resources which are able to perform that request. Similarly to (2), a WIP agent will choose the most suitable bid and execute it.

On the other hand, resource holons try to optimise their utilisation during their negotiation. At the end of the negotiation, the WIP agent obtains an *execution table* which contains a specific operation sequence, equipment to be used, and the expected execution time. This execution table can be regarded as a specification for a dynamic and virtual manufacturing line which is formed by the participating resources holons (i.e. a manufacturing holarchy). Each WIP agent which is created from the same product holon will produce different execution tables because of the different conditions of each physical part and arrival time. The execution table is intended to implement the function of real-time dispatching. This can reduce the impact of disturbances as well as implement short-term scheduling to obtain a better global performance.

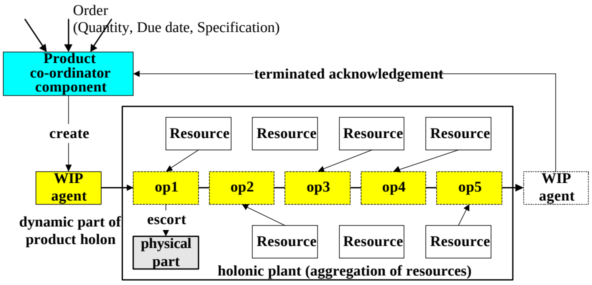

Fig. 3. Interaction between products and resources

4.2 Co-operative diagnosis

In manufacturing diagnosis, a significant amount of work is focused on the development of the operations of a machine or of a section of the plant. Fault diagnosis has, in general, been treated as an isolated activity within process operations, rather than in the context of the overall system. An integrated approach has the potential to make use of the diagnosis information to improve the diagnosis capability, and therefore to eliminate unnecessary stoppages. In recent years, research has paid more attention to making the best use of available information in performing diagnosis and in co-ordinating diagnosis with which linked process operations [5].

In HCBA, *co-operative diagnosis* is proposed to implement integrated diagnosis, by co-ordinating the diagnosis capability of individual holons in a co-operative manner. The basic concept of co-operative diagnosis can be described using the diagram in Fig. 4. Besides diagnosing themselves, holons are able to report their errors to other related and relevant holons. Some cases are interpreted below to introduce the co-operative manner and potential benefits of this approach.

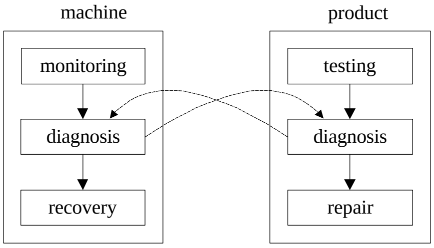

Fig. 4 Co-operative diagnosis between resource and product holons

Case (1): An error occurs within a machine. The resource holon sends the error information to the product holon which is escorting the associated product in this machine. Thus, the product holon is able to take an action to check the potential influence on itself in advance and prevent the propagation of the fault.

Case (2): A defect is found in a product. The defect can be observed either by testing the product directly or by fault information reported from a resource holon (see case (1)). The associated product holon reports this defect information to the related resource holons which may have caused this product defect in the previous operation. This notice provides extra diagnosis information for a resource holon to examine its implicit error. The diagnosis capability of a resource holon can be extended by the diagnosis information of other holons without adding new sensing devices.

Case (3): Similar to case (2), a defect is found in a product. A product holon provides the fault information to a resource holon which will perform a task on this product in the future. This warning information tells the resource holon that an error may occur due to the side effect of a fault in the product. Therefore, this machine may provide a more conservative procedure or request human attendance. This proactive information may reduce the risk of a new fault and stop the propagation of an old fault.

Due to the procedure of co-operative diagnosis, a holon can obtain more diagnose information than it has individually. The on-line and real-time diagnosis capability of holons can be increased without adding new sensors or diagnosis algorithms. As a result, failure impact and propagation can be reduced. In addition, the quality of the product and the utilisation of the machine can be improved with the development of co-operative diagnosis in HCBA.

5. Implementation

The concept of HCBA has been implemented successfully for a robot assembly cell. A holonic-like system is derived from the conventional PLC-based cell control by the migration approach mentioned above. This testbed consists of four pieces of equipment. Three kinds of parts annotated A, B, and C are used to produce two kinds of products. (Part colour variations also increase the potential product mix) The control architecture is depicted in Fig. 5. The cell PLC is employed to integrate machines in this cell and then link upwards to the resource controller. The detailed design of this testbed is discussed in [1].

The primary means of demonstrating the effectiveness of the HCBA approach to date has been in its successful application to the robot assembly cell. In particular, we have verified the testbed results in the following ways:

(1) Separated system integration: In order to implement the system development procedure, the design of the testbed is deliberately separated into two groups to test the new approach to system integration. The first group is to develop resource holons and static integration in the *resource controller*. The second group is to develop product holons and dynamic integration in the *production controller*. The complete integration is achieved after both groups have provided their final design. This approach verifies that the integration of HCBA which is not necessarily achieved by one designer or a specific team is possible. System users are able to reconfigure the system by "plugging" the self-contained and ready-to-run holonic components which could be provided by different experts, into this architecture.

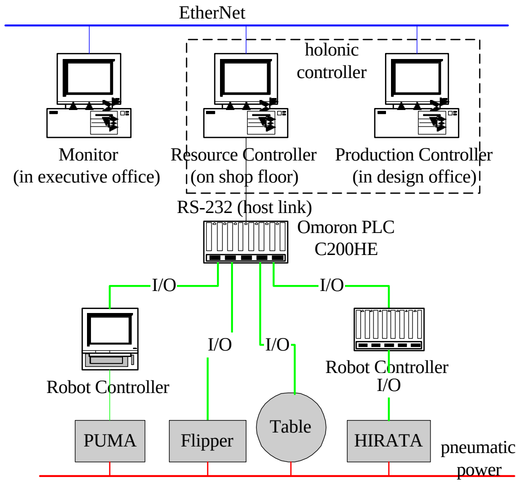

Fig. 5 Control Architecture for a Robot Assembly Cell

(2) Reconfigurable and scalable system: A scenario is designed to test the system reconfigurability when the manufacturing equipment or the production process is changed. The testbed is initially operated to assemble the Product AB by using the Puma robot, the rotary table and the Hirata robot. Later, a new Product ABC is introduced into the testbed. The flipper unit is employed to perform a new operation to achieve this plan. The change implemented by the holonic controller is to plug the software component of the flipper unit into the resource controller and to plug the software component of Product ABC into the production controller instead of the software component of Product AB. The manufacturing plan is immediately changed without causing any side effects to the other parts of the system.

(3) Virtual distributed manufacturing controller: Unlike the monolithic application in traditional design, the communication infrastructure builds a virtual control system to accommodate system components which could be run anywhere using the Internet as a means of connection. That is, here is no physical boundary for a holonic controller. Because of the concept of HCBA, with the aid of Internet, no fixed cell controller exists to integrate a physical plant. Software components of resources and products, each of which may be provided from different sources, can dynamically and temporarily form a holonic controller to perform a specific operation in this physical cell. This architecture provides a transparent plant for machine vendors and customers to remotely exploit its functionality, and has potential to link with the emerging technology of *E-commerce*.

This designed test bed has been mainly implemented for manufacturing execution and dispatching. Co-operative diagnosis is being developed to integrate with co-operative execution. In the future, scheduling issues will also be investigated to improve global performance based on this framework. In addition, due to the rapid growth of E-commerce, we hope to integrate this new manufacturing control architecture with the E-commerce environment by a standard commercial infrastructure, such as DCOM or CORBA.

6. Reference

[1] Chirn, J-L and McFarlane, D. C., "A Component-Based Approach to a Holonic Control of Robot Assembly Cell", *IEEE ICRA*, 2000.

[2] Chirn, J-L and McFarlane, D. C., "A Migration of Holonic Approach to Production Control Systems", *1st IFAC Workshop on MAS'99*, Vienna, 1999.

[3] Cox, B.J., *Object-Oriented Programming - An Evolutionary Approach*, Addison-Wesley Publishing Company, 1986.

[4] Christensen, J., "Holonic Manufacturing Systems - Initial Architecture and Standards Directions", *1st Euro. Conf. on HMS*, Hanover, Germany, 1994.

[5] Karcanias, N., and McFarlane, D., " System Design Issues for Integrated Diagnostics ", *Proceedings of ASI'95*, 1995.

[6] Koestler, A., *The ghost in the machine*, Arkana Books, 1989.

[7] Reveliotis, S. A., "Production Planning and Control in Flexibly Automated Manufacturing Systems: Current Status and Future Requirements", *IEEE ICRA*, p.p.1442-1449, 1999.

[8] Suda, H., "Future Factory System Formulated in Japan", *Japanese Journal of Advanced Automation Technology*, Vol. 1, pp. 67-76, 1989 (1) and Vol.2, No.1, pp. 58-66, 1990 (2).

[9] Szyperski, C., *Component software: beyond object-oriented programming*, New York: ACM Press; Harlow: Addison-Wesley, 1997