EVALUATING HOLONIC CONTROL SYSTEMS: A CASE STUDY

Jin-Lung Chirn*, Duncan McFarlane**

TTP Com, Melbourn Science Park, Cambridge Road, Melbourn, Royston, Herts, SG8 6HQ, UK
*** Institute for Manufacturing, University of Cambridge, Mill Lane, Cambridge, CB2 1RX, UK*

Abstract: In the last 10 years many designs and trial implementations of holonic manufacturing systems have been reported in the literature. Few of these have resulted in any industrial take up of the approach and part of this lack of adoption might be attributed to a shortage of evaluations of the resulting designs and implementations and their comparison with more conventional approaches. This paper proposes a simple approach for evaluating the effectiveness of a holonic system design, with particular focus on the ability of the system to support reconfiguration (in the face of change). A case study relating to a laboratory assembly system is provided to demonstrate the evaluation approach.

Keywords: manufacturing, control, agent, holonic manufacturing system, evaluation, case study

1. INTRODUCTION

1.1 Aims

In the last 10 years many designs and trial-implementations of holonic manufacturing systems have been reported in the literature. Few of these have resulted in any industrial take up of the approach and part of this lack of adoption might be attributed to a shortage of evaluations of the resulting designs and implementations and their comparison with more conventional approaches. This paper proposes a simple approach for evaluating the effectiveness of a holonic system design, with particular focus on the ability of the system to support reconfiguration (in the face of change). The approach proposed is most appropriate in the evaluation of a physical implementation rather than at the conceptual level.

1.2 Background to Holonic Manufacturing Systems Design

The concept of holonic manufacturing was introduced by Suda in the early 1990s (Suda, 1989) to address emerging challenges for manufacturing in the 21st century. It is intended to enable a ‘‘plug and play’’ approach to designing and operating a manufacturing system. Holonic manufacturing systems (HMS) represent a methodology, tools and accompanying standards for the design of flexible, reconfigurable control systems in the manufacturing supply chain. Importantly, HMS represents a methodology for control system design and operation that manages disruptions and changes as *business as usual.* The approach is based around the concept of automated, distributed or decentralised decision making which enables multiple entities in the manufacturing environment to be able to influence manufacturing control decisions. The reader is referred to the recent book by Deene (Deene, 2003) for a collection of introductory papers on this topic.

1.3 Evaluating Holonic Manufacturing

A barrier to the deployment of holonic based systems in industry has been the limited literature produced in the area of the evaluation of designs and/or the subsequent performance of the resulting control system, and in particular the lack of comparison with conventional based control systems. We acknowledge evaluation and comparison work related to holonic manufacturing

in (Fletcher et al, 2003), (Bussmann, 2004), (Leitao, 2004), and the ongoing benchmarking work in the EU Intelligent Manufacturing Systems Network as examples of current efforts in this area, and we speculate that some of the reasons for the lack of evaluation research in the past are:
1. A lack of common base to evaluate because of different interpretations of the holonic manufacturing approach.
2. A lack of appropriate performance criteria and quantitative measures to evaluate them
3. A lack of standard scenarios and benchmarks
4. A research field where there is less emphasis on proving or verifying performance compared to producing new designs and methods.

This paper will directly address issues 2. and 3. and will contribute in part to the others. We begin the paper by proposing our approach to evaluating holonic manufacturing system designs and then devote the remainder of the paper to evaluating a case study on the control design for a simple assembly cell.

2. EVALUATION APPROACH

In setting out to assess the performance of a industrial control system design, there are two distinct areas that can be addressed.
1. System Performance: the assessment of the development and performance of the control system itself
2. Operational performance: the assessment of the performance of the operations influenced by the control system
The evaluation approach developed here is focussed only on the former, and is particularly concerned with reconfigurability of the control software environment in the face of changes to product or resources. We also emphasise that the evaluation approach is intended to be used at an implementation level – rather than as a means of evaluating a generic design approach

We next discuss the three key elements to the approach, which are
- Specification a representative set of evaluation scenarios within the selected problem domain
- A representative set of performance criteria and measures which reflect the evaluation goal (system design).
- A set of candidate designs to be evaluated through their implementation and testing

2.1 Control System Evaluation Scenarios

Because holonic systems are understood to be most applicable in situations of disruption and / or change – the target control system design application should be selected on this basis. Then, a series of test scenarios must be selected in order to capture the key areas to be evaluated. For example
1. *base case*
2. *product change (mix, specifications)*
3. *resource change (availability, configure)*

In each case it is important that the scenario conditions be repeatable and reflect the operating envelope of the system to be controlled, and clearly that they reflect the main *change* conditions.

2.2 Performance Criteria for Assessing Industrial Control Systems

I Strategic Complexity of Control System Software – indicates the level of complexity of the control system in design phase in terms of the specified sequences, concurrencies and dependencies.

II Operational Complexity of Control System Software – indicates the level of complexity of the control system in implementation phase in terms of the number of different commands and steps required. The effort of implementing the manufacturing control can be measured by examining the complexity of the source code underlying the implementation. A large number of measures for software complexity have been proposed in the past three decades (Zuse, 1991). These measures can be classified in terms of the different attributes of the software, such as specification, design and code length metrics (Itzfeldt, 1990) to provide a complexity index covering different perspectives.

III Reconfigurability of the Control System Software – indicates the degree of reconfigurability and reusability afforded over time through the evolution of the control system. We further define two indexes to examine the system reconfigurability according to the change from an existing application to a new application based on the two measures introduced above.
- *Extension rate*: This index represents the growth rate of the scale or complexity of a new scenario compared with that of the existing scenario. A higher extension rate means that the system is less reconfigurable.
- *Reuse rate*: This index is defined as the percentage of the existing design or codes used in a new scenario. A higher reuse rate means that the existing system takes less effort to reconfigure.

Defining measures for each of these criteria needs to be done on a case by case basis. We note that in Neely et al (1995) a set of criteria have been developed for the general selection of performance measures, and in particular it is required that measures be measurable, representative, quantitative and objective. These criteria need to be applied in the specific case (see Section 3.)

2.3 Selecting Comparative Designs

To evaluate a new control system design it is necessary that a conventional design be available for comparison. However, It is very difficult to provide a true comparison for a practical industrial implementation and many factors which at best can only be partially accounted for. Some typical

requirements for an alternative control design to be used for comparison are:
- the design addresses the same goals, problem domain
- the performance measures are meaningful
- expertise exists to perform the alternative design
- infrastructure exists to implement the alternative design

We also note that generally there is some compromise in such a comparison and it is unlikely that evaluation is truly objective. Hopefully however, the comparison leads to meaningful insights, and provides indications of strengths / weaknesses of proposed approach.

3. CASE STUDY OVERVIEW

In this section, a physical case study is presented to evaluate a holonic design. A holonic design approach (Chirn, McFarlane, 2001) is used to develop a control system for an assembly cell and performance measures are established to assess reconfigurability compared with a conventional approach. We emphasise that this case study and hence the evaluation is limited only to control and execution functions at cell level without any consideration of scheduling or planning functions.

3.1 Problem Domain and Evaluation Scenarios

The testbed used for the implementation of both a holonic and conventional control systems was a simple robotic assembly cell. The job of the robot assembly cell is to complete the assembly of a simple electrical meter box. Details of the operation are given in (Chirn, 2003). The layout of the cell is shown in Figure 1.

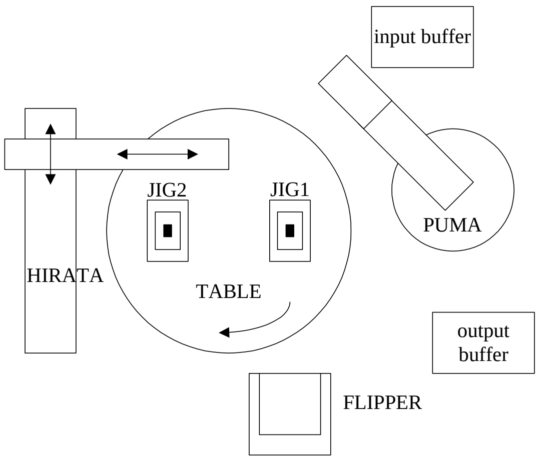

Figure 1 Layout Of The Robot Assembly Cell

Three kinds of parts, annotated as Parts A, B, and C, are used to assemble the two products. One is denoted as *Product AB*, and the other product, referred to as *Product ABC.* The two products require a different resource set to perform their respective assembly needs. The goals of this implementation were to develop both holonic and conventional control solutions with existing hardware and for the system to be able to construct parts AB and ABC either separately or simultaneously in finite time.

Test Scenarios for the Robotic Assembly Cell

Five test scenarios were designed to represent changes made to the cell over time. These scenarios test the relative reconfigurability of the two control system design approaches by changing the specifications of the required resources, or products or both. The full set of scenarios is described in Chirn and McFarlane (2005). For space reasons, only three scenarios are introduced in this paper:

I Base Case - Assembly of product AB: This scenario is to measure the “design cost” to build a new cell controller given required hardware and fundamental execution functions.

II New Product Introduction - Assembly of new product ABC: The cell is redesigned to assemble a new product ABC which consists of parts A, B, and C. A new piece of equipment (*flipping unit)* is also required to do this.

III New Product Mix: The testbed is modified to perform mixed assembly of both products *AB* and *ABC* at the same time. The control should be able to perform the concurrent assembly operations of two products, either product *ABC* or product *AB*, continuously and randomly.

3.2 Performance Measures

We now introduce the specific measures used in this study to evaluate the designs against the criteria set in Section 2.2.

(i) Strategic Complexity of Control System Software: By measuring the complexity of the Petri-nets describing the overall control strategy, it is possible to obtain an indication of the cost of the design, testing and maintenance of the control system. According to measures proposed in (Hura, 1981; Lee, 1992; Venkatesh, 1994), the complexity of Petri-net can be measured by the following index.

$$C_{PN} = N_{OP} + N_{OT} + N_{OA} \quad (1)$$

Here N_{OP} is the number of places, N_{OT} the number of transitions and N_{OA} the number of arcs.

(ii) Operational Complexity of Control System Software: A simple, *weighted lines-of-code* measure (C_{LOC}) is used to indicate the complexity of the design in our study. (See also Beck, 2000; Reagin, 1999).

$$C_{LOC} = N_C + 0.5N_d \quad (2)$$

where N_c is the number of lines of conditional and control codes while N_d is the number of lines of data processing codes. Refer to (Chirn, McFarlane, 2005) for more details.

(iii) Reconfigurability of the Control System Software: indicates the degree of reconfigurability and reusability afforded over time through the evolution of the control system. We further define two indexes to examine system reconfigurability

according to change from one scenario to the next based on the two complexity measures introduced above.

- *Extension rate*: The index can be defined based on designed Petri-net or source codes respectively:

$$(E_{PN})_{i+1} = \frac{(C_{PN})_{i+1}}{(C_{PN})_i} ; (E_{LOC})_{i+1} = \frac{(C_{LOC})_{i+1}}{(C_{LOC})_i} \quad (3)$$

where i is the scenario number.

- *Reuse rate*: The reuse rate can be calculated from the measures of source codes in the old scenario i and the new scenario i+1 as follows.

$$(R_{LOC})_{i+1} = \frac{(C_{LOC})'_{i+1}}{(C_{LOC})_{i+1}} \quad (4)$$

where i is the scenario number, $(C_{LOC})'_{i+1}$ is the subset of $(C_{LOC})_i$ which is reused from $(C_{LOC})_I$, i.e. $(C_{LOC})'_{i+1} \in (C_{LOC})_i$)

The performance measures described above are not intended to be definitive and by no means will apply to all situations relating to performance measurement of holonic systems. They are however quantitative and can be used to reflect properties of both holonic and conventional system designs which is an important feature.

3.3 Candidate Designs

Holonic Control System Design

The manufacturing control architecture proposed in this study is the so-called Holonic Component-Based Architecture (HCBA) which has been outlined in detail in (Chirn and McFarlane, 2001), (Chirn 2003). In this approach an underlying system building block - a holonic component - and an infrastructure for the HCBA are introduced.

Holonic components are self-contained and ready-to-run entities possessing autonomous and co-operative properties. In this design they represents the different physical resources and also the products to be made. The resource component or resource holon is a self-contained system component which can give treatments to works in process, such as fabrication, assembly, transportation, and testing. Typical resource components are machines, robots, etc. Besides the physical part, a resource component contains an invisible control part, which can perform its operations, decision-making and communication ability with the aid of its local database. On the other hand, the product component or product holon also contains a physical part and a control part. A physical part may include raw material, parts or pallet/fixture. A control part may contain routing control, process control, decision-making or production information. Unlike conventional plant control, there is no central controller to manipulate the overall manufacturing operations in this new architecture. The control is generated by the interaction of manufacturing holons.

In addition, the *holonic infrastructure* provides an open platform to accommodate these holonic components, which can be provided from varied sources. Thus, these holonic components can be "plugged in" to this platform and used to build a modular and reconfigurable manufacturing control system. Two major communication mechanisms: a *Blackboard System* (BBS) for intra-holon communications and a *Message Broker* (MB) for inter-holon communications were developed. The detailed design of these can be referred in (Chirn, 2001).

Conventional Design

Although there is no universal standard, the reference conventional control design used here has features which have been applied to a many control applications in the past such as *top-down design, central process-based development, functional decomposition*, and *system designer-dominated integration*.. We acknowledge that our design selection is subjective in nature and that the results that follow should be viewed in this light. The conventional reference approach adopted here is referred to as the Conventional Centralised Approach (CCA). The CCA controller is designed and implemented based on a Petri-net design, which is a common approach in the design of manufacturing control systems (DiCesare 1993; Van Brussel, 1993; Narahar, 1985). More details are in (Chirn, 2003)

4. CASE STUDY: RESULTS

In this section we provide the main results of the evaluation of the two control designs for the disassembly cell and then provide an interpretation of the results in terms of the relative differences of a holonic and conventionally based design.

4.1 Results of Evaluation

The three tables that follow in Tables 1-3 represent the main results from the evaluation of the system performance of the holonic control design against the conventional control design. In each case we provide measures for the three performance criteria outlined in the previous section.

Table 1: Scenario I Evaluation: Base Case

	Holonic Control	Conventional Control
Complexity of Strategy	83	143
Complexity of Development	2055 (1507)	315
Reconfigurability: Extension	n/a	n/a
Reconfigurability: Reuse	n/a	n/a

Table 2: Scenario II Evaluation: New Product

	Holonic Control	Conventional Control
Complexity of Strategy	103	220
Complexity of Development	2407 (1585)	497
Reconfigurability: Extension	Strategy: 1.24 Develop: 1.17	Strategy: 1.54 Develop: 1.62
Reconfigurability: Reuse	0.95	0.4

Table 3: Scenario III Evaluation: Mixed Products

	Holonic Control	Conventional Control
Complexity of Strategy	103	252
Complexity of Development	2407 (1585)	595
Reconfigurability: Extension	Strategy: 1.00 Develop: 1.00	Strategy: 1.15 Develop: 1.20
Reconfigurability: Reuse	1.0	0.9

We make some simple observations and comments before examining the evaluation criteria in detail:

- The amounts in brackets in the complexity of holonic design reflect the value of the measure were a series of identical templates used in designing each holon not counted multiply.
- Note that the target value for reconfigurability in each case is 1.00 which reflects a 100% reconfigurability measure in some sense.
- In Scenario I there is clearly no reconfigurability data as this represents the base case. In fact the reconfigurability data for scenarios II and III represents more accurately reconfigurability from I -> II and II -> III respectively as the reconfigurability measures are relative measures, depending on initial conditions.

In the following sections we provide comments relating to complexity and reconfigurability of the designs respectively.

4.2 Strategic and Operational Complexity

In this section, referring to Tables 1-3 above, the strategic and operational complexity of the control architecture for both holonic and conventional approaches are compared. Referring to the strategic complexity data for Scenario I in Table 1 and comparing with the single bulk of the conventional Petri Net design in Figure 2 with the Petri-net models for resource and product holons in HCBA in Figure 3 it is clear that the latter appear much simpler, although we acknowledge that this is only indicative and that the Petri nets do not allow for any quantitative comparison in this form. (In fact, the overall manufacturing operations in HCBA can be represented by the separated Petri-net designs which are distributed across four Petri Net based system components - 3 resource holons and 1 product holon). These Petri-net models that can implement the individual execution behaviour of resource and product holons. However, the distributed Petri-nets for each of the four resource holons in HCBA are not able to implement the required functions and extra design effort (and PN modelling) is needed to integrate the communication infrastructure of the message broker, and the communication and decision-making capability in each system component. These extra designs l increase the design cost. From Table 1, it is clear that the scale of the software development of HCBA is much bigger than that of CCA. Part of the reason for this is that the initial software overhead is higher in each component in order to maintain the minimum requirements of communication functions in HCBA. This property will be discussed later.

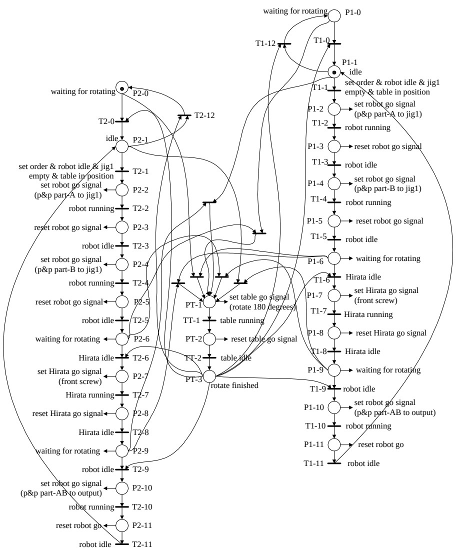

Figure 2. Petri-net model of a conventional approach for Scenario I

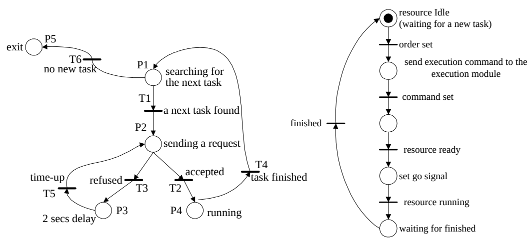

Figure 3 Petri-net model of a resource holon (left) and a product holon (right)

Scenario II involves the further assembly of a part C on the existing product AB. A flipper machine is added in this cell to implement a flipping function. As a new machine is added and the new process is longer than that of the previous scenario, the control system becomes more complex. Basically, the topology of the new Petri-net model in the CCA is the same as the one in the previous scenario but need to be extended. Alternatively, in HCBA, all system components designed in Scenario II are retained. Since a new machine has been introduced and the process extended in this scenario, both approaches have to be redesigned to meet the new requirements. In CCA, it is necessary to go back to modify the original Petri-net model. As the process of product AB is part of the process of product ABC, a new Petri-net model can be revised based on the previous model. In Scenario III (Refer to Table 3) the introduction of a two product types instead of one has no effect on the existing complexity as there is effectively no change to the operating environment. In contrast the conventional controller needs to be adjusted to account for the possibility of simultaneous products being made.

4.3 Reconfigurability

As introduced in section 2, extension rate and the reuse rate can be examined to measure the reconfigurability of the test bed. The figures are obtained according to the change of the strategic and operational complexities from one scenario to another. The results for both HCBA and CCA are given in Tables 1-3 but are also are summarised in Table 4.

Table 4. Reconfigurability Analysis

	Extension-Strat		Extension-Dev		Reuse	
	HCBA	CCA	HCBA	CCA	HCBA	CCA
II-III	1.24	1.54	1.17	1.62	0.95	0.4
III-IV	1.0	1.15	1.0	1.20	1.0	0.9

It is shown that the complexity of the CCA increases significantly more than that of HCBA from the measures of both Petri-net and source code. Furthermore, the code reuse rate in HCBA is comparatively higher than that of CCA. Obviously, the HCBA has significantly less impact to these test scenarios comparing with the CCA.

5. CONCLUSIONS

The paper has proposed an evaluation approach for comparing holonic control system designs with other competing designs. It has been trialed in a study which compares holonic and conventional control systems designs for an assembly operation. With limitations, the evaluation has shown that the holonic approach can provide the potential advantages in reconstructing the control structure over a predefined set of different scenarios. Furthermore, the holonic design indicates lower extension rate and higher reuse rate which is an indicator that the design approach has higher reconfigurability and modularity when facing a series of design changes.

6. REFERENCES

Beck, J. E., Reagin, J. M., and Sweeny, T. E., (2000) "Applying a Component-Based Software Architecture to Robotic Workcell Applications", IEEE Transactions on Robotics and Automation, **Vol. 16, No. 3,** pp. 207-217.

Bussmann, S, (2004) An Agent Oriented Design Methodology for Production Control, PhD Thesis, Univ. of Southampton, UK

Chirn,J-L, McFarlane, D, (2001) "Building Holonic Systems in Today's Factories: A Migration Strategy", J Applied Systems Studies, Special Issue on "Industrial Applications of Holonic and Multi-Agent Systems", **Vol. 2, No. 1**

Chirn, J-L, (2003) "Developing a Reconfigurable Manufacturing Control System - A Holonic Component-Based Approach", PhD thesis in Department of Engineering, University of Cambridge

Chirn, J-L, McFarlane, D, (2005) Evaluation of Holonic Manufacturing Systems, *paper under preparation,*

Deene, M, ed, (2003) "Agent Based Manufacturing – Advances in the Holonic Approach", Springer, Berlin Heidelberg,

DiCesare, F., Harhalakis, G., Proth, J. M., Sliva, M., and Vernadat, F. B., (1993) Practice of Petri Nets in Manufacturing, Chapman & Hall

Fletcher, M, Brusey, J, Thorne, A, McFarlane, D, (2003) Holomas 03, Evaluating a Holonic Packing Cell, Proceedings of Holomas 03, Prague, Czech Rep.

Hura, G. S., Singh, H., and Nanda, N. K., (1981) "Petri Net Approach to the Evaluation of the Complexity of a Program", International Journal of Electronics, **Vol. 51, No. 1**, pp. 79-85.

Itzfeldt, W. D., (1990) "Quality Metrics for Software Management and Engineering", Managing Complexity Software Engineering, IEE computing series 17, Chapter 6, pp. 127-151

Lee, G. S. and Yoon, J-M., (1992) "Empirical Study on Complexity Metrics of Petri Nets", Microelectronics and Reliability, **Vol. 32, No. 9,** pp. 1215-1221.

Leitao, P, (2004) An agile and adaptive holonic architecture for manufacturing control, PhD Thesis, Universidade do Porto, Portugal.

Narahar, Y. and Visvanadham, N., (1985) "A Petri Net Approach to the Modelling and Analysis of Flexible Manufacturing Systems", Annals of O.R., **Vol. 3**, pp. 449-472.

Neely, A, Platts, K, Gregory, M, Performance Measurement System Design, IJOPM, **Vol.15, No. 4,** pp. 80-116

Reagin, J., et al, "A Component-based Software Architecture for Flexible Manufacturing Systems", IEEE Transactions on Electron, Packaging Manufacturing, **V 20**, pp.85-94, 1999.

Suda, H., (1989) "Future Factory System Formulated in Japan", Japanese Journal of Advanced Automation Technology, **Vol. 1,** pp. 67-76.

Van Brussel, H. V., Peng, Y., and P. Valckenaers, (1993) "Modelling Flexible Manufacturings Based on Petri Nets", Annals of the CIRP, **Vol, 42, No. 1,** pp. 479-484.

Venkatesh, K., Zhou, M., and Caudill, R., (1994.) "Evaluating the Complexity of Petri Nets and Ladder Logic Diagrams for Sequence Controllers Design in Flexible Automation", IEEE Symposium on Emerging Technology and Factory Automation, pp. 428-435,

Zuse, H., (1991) Software Complexity Measures and Methods, Walter de Gruyter.