INT. J. PROD. RES., 2001, VOL. 39, NO. 10, 2155–2176

An agent-based production control framework for multiple-line collaborative manufacturing

TA-PING LU† and YUEHWERN YIH†*

This research focuses on constructing an agent-based collaborative production control framework that is capable of conducting scheduling and dispatching functions among production entities, as well as within them, in a collaborative manufacturing environment. The proposed framework utilizes autonomous agent and weighted functions for distributed decision-making while all agents work in active and collaborative ways to help each other make decisions. This collaborative control framework is capable of realizing and seeking balances among heterogeneous objectives of the production entities within a collaborative manufacturing system. The agents in this control framework were constructed with an object-oriented prospective so that a production entity can join or depart from the control scheme without affecting the rest of the framework. Simple index values, instead of detailed data, were used for information exchange among agents. This can greatly reduce the communication and computation load of the control system and keep detailed production information confidential while the agents in the system could belong to different companies. In this research we created a simulation model of a real-world multi-line elevator manufacturing system as the test bed to evaluate the performance of the proposed control framework. Two other control strategies with different levels of collaboration were applied to the simulation model to compare and evaluate the performances of the proposed control strategy. Results of the simulation show that multiple objectives of the production entities can be realized.

1. Introduction

Driven by factors such as global competition, downsizing, low volume and high variety production, and advances in computing coupled with communications technologies, manufacturing industries have a new emphasis on partnerships and alliances. It is our belief that collaborative arrangements will become an increasingly necessary means of meeting the needs of manufacturing companies in the future. According to the findings of the Collaborative Manufacturing Center at Purdue University, virtually everyone spoken to in manufacturing companies agrees with this premise.

As a currently ongoing research topic, 'collaborative manufacturing environment' has no generally accepted definition at this time. Our definition of the collaborative manufacturing environment states that it is an environment where production entities—which can be a manufacturing site, line, cell or machine-work collaboratively on a short-term basis to produce a product that cannot be produced by any individual entity within the environment. The production entities

Revision received October 2000.
† School of Industrial Engineering, Purdue University, West Lafayette, Indiana 47907-1287, USA.
* To whom correspondence should be addressed. e-mail: yih@ecn.pudue.edu

International Journal of Production Research ISSN 0020–7543 print/ISSN 1366–588X online © 2001 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/00207540110038478

in the environment are individual profit centres that can belong to different companies. In this environment, the production entities gathered to form a team as required by a project, which is an order of a particular product. After the project is completed the team will dissolve. Each production entity can, and usually will, join multiple projects at the same time. The highly dynamic nature of a collaborative manufacturing environment makes the traditional production control approaches inadequate because these approaches only work when the domain of entities is fixed.

The term *agent* is often used in the artificial intelligence domain and refers to an entity that can perform a task continuously and autonomously in a non-determinant environment, where other processes and entities exist. Although the term, agent, is used frequently, it has no unified meaning, definition or structure (Lei *et al.* 1998). It can be a hardware or software component (Shoham 1993), or it can be a combination of human users and software tools (Genesereth and Nilsson 1987, Khedro 1996). However, many researchers in different fields have defined agents as possessing certain fundamental characteristics and capabilities (Crowston and Malone 1988, Lei *et al.* 1998, Shoham 1993, Tseng *et al.* 1997). First, an agent must possess and maintain certain information. Second, it must be able to interact with its environment and extract knowledge from it. Third, the agent must be able to communicate with other agents for information and knowledge exchange. Last, it must be able to process information and make decisions autonomously. For the purpose of this research we define an agent as a software program that has the above fundamental characteristics and capabilities.

This research focuses on constructing a agent-based collaborative production control framework that is capable of conducting scheduling and dispatching functions among production entities as well as within them in a collaborative manufacturing environment. This framework utilizes autonomous agent and weighted functions for distributed decision-making while all control entities, namely agents, work in an active and collaborative way to help each other in making decisions. This collaborative control framework is capable of realizing and seeking balances among heterogeneous objectives of the production entities within a manufacturing system. The control framework consists of autonomous agents with object-oriented prospective so that an entity can join or depart from the control scheme without affecting the rest of the framework.

In this research we created a simulation model of a real-world multi-line elevator manufacturing system as a test bed. In the elevator production system, all sub-assemblies belonging to the same order must be gathered in a grouping buffer at the end of the production line and loaded into the truck in a particular sequence. Applying traditional production control methods will result in high inventory and handling costs at the grouping buffer due to long matching times, which refers to the time from the first sub-assemblies of an elevator order arriving at the grouping buffer to the last one arriving. The penalty from poor on-time delivery rate will also be high. Control strategies within three different levels of collaboration were applied to the simulation model to compare and evaluate the performances of the proposed control strategy.

2. Literature review

2.1. *Agent-based production control strategies*

Since the late 1980s, a number of researchers have applied agent technology to perform several part production control tasks on the shop floor level

(Balasubramania and Norrie 1995, Sikora and Shaw 1997). While conventional part manufacturing control systems have centralized and top-down control with scheduling and control decisions made sequentially ahead of time, agent control systems, also refereed to as responsible agent systems, are autonomous control with scheduling and control decisions made in real-time. Sluga *et al.* (1998) proposed a multi-agent approach to the development of a distributed manufacturing architecture and attempted to define the autonomous building blocks of the system. Sikora and Shaw (1997) presented the coordination mechanisms needed for ensuring the orderly operations and concerted decision making among agents in a multi-agent scheduling system.

A number of research studies have been completed with attempts to use a multi-agent system to integrate certain design and control functions that were traditionally performed separately (Balasubramanian and Norrie 1995, Gu *et al.* 1997, Kempenaers *et al.* 1996). Balasubramania and Norrie presented a multi-agent architecture for the integration of design, manufacturing, and shop floor control activities. It is based on cooperating intelligent entities in the sub-domains, which make decisions through negotiation, using domain-specific knowledge both distributed among the entities and accessible to them. In 1997, Gu *et al.* presented a bidding-based process planning and scheduling system to integrate design, process planning and scheduling in a market-like environment.

2.2. *Production control strategies for collaborative manufacturing*

Kempenaers *et al.* (1996) proposed an integrated automatic process planning and scheduling system based on the concept of nonlinear process plans, which is the same concept as flexible routing in Lin and Solberg's (1992) work. Instead of using agents, Kempenaers *et al.* proposed adding an evaluation module to each of the following: the process planning module, the scheduling module and the workshop resource module in this control system. Feedback information from all modules in this system was used by these evaluation modules for the collaboration among these traditionally separated control functions.

Certain researchers (Lee *et al.* 1996) proposed a collaborative scheduling system for made-to-order manufacturing, in which the details about the product specification become gradually known in phases from development, design and process planning. This production control system consists of three scheduling modules: presumptive scheduling modules, predictive scheduling modules, and reactive scheduling modules. These modules perform scheduling functions based on different data collected at different phases of production. By collaborating with each other, these scheduling modules contribute to the operation schedule that was actually implemented on the shop floor. Data collected from implementing this scheduling system in a real-world made-to-order factory for more than a year showed a reduction in manufacturing lead time of 25% on average and an improvement of machine utilization of 10 to 20%.

Nof and Weill (1992) proposed a collaborative coordination control structure for distributed multi-machine manufacturing. It focuses on collaboration among machines, in sharing tasks and resources, and with human supervisors in order to obtain global information, external knowledge and plan revisions. This control structure was different from the above stated past researched methodologies by providing an interface between control modules and human supervisors/operators for human-machine cooperation in decision making. This is required for new or unusual situations that have not yet been learned or programmed in the control computers. Two case studies showed this control structure has the potential to deliver shorter lead-time, higher production rate, and lower cost through machine collaboration.

2.3. *Agent-based production control strategies for collaborative manufacturing*

Certain researchers (Lin and Solberg 1992, Tseng *et al.* 1997) proposed an agent-based production control framework that utilizes distributed decision-making and distributed information flow in a market-like environment based on a price and objective mechanism. This framework supports heterogeneous job objectives, admits job priorities, recognizes multiple resource types, and allows multiple step negotiation between parts and resources. Under the framework, each job enters the system with some currency and tries to purchase processing services. On the other hand, resources determine their service charge based on demands and try to sell their service to maximize their profit. In this way, the complicated coherent manufacturing control problem is decomposed into a collection of independent agents' decision-making problems. The global states and entity interactions are reflected and controlled by a price system. Lin has proposed two different negotiation schemes, namely the part-initiated negotiation scheme and the resource-initiated negotiation scheme. Tests have been conducted to compare the effectiveness of these two schemes under different manufacturing environments. Tseng was aiming at the flexibility that the agent-based control system can provide and proposed a collaborative control system for made-to-order manufacturing.

It was believed that autonomous agent controlled architecture was one of the concepts that would greatly improve a shop's flexibility and responsiveness because the dynamic scheduling under the architecture can adapt to the changes in the shop floor in real time (Herrin 1994). It was also suggested that collaboration among system resources could serve to increase the system's flexibility, reliability and production rate (Nof and Weill 1992). It is our observation that while the term *collaboration* was not mentioned in certain agent-based control strategies, the concept of collaboration among system resources was actually implemented in the control system by the communication or negotiation among agents. On the other hand, in certain production control frameworks, intelligent and autonomous entities were used to perform the control functions, although the term *agent* was not used. Large portions of the researches we have found in the field of agent-based collaborative control were for made-to-order or job-shop type manufacturing. Therefore, it is our belief that the agent-based collaborative control scheme has great potential for the manufacturing environments of the future, with an emphasis on flexibility and fast responses in order to meet the ever-changing customer requirements.

These control systems contribute to the delivery of higher production efficiency, more flexible systems, better on-time delivery rates, higher machine utilization, and lower inventory. However, up-to-date research has focused on production control tasks at the shop floor level and the products of these systems were individual parts. These control systems have a domain of a company or a profit centre and the collaboration was among system resources within the domain. Our research, on the other hand, focuses on higher-level production control functions, which ensure good parts synchronization and on-time delivery rates of complicated multi-part products. The control framework has a flexible domain with manufacturing entities belonging to different companies or profit centres working collaboratively to produce the product.

3. Problem definition

3.1. *Problems associated with traditional production control approach*

The collaborative manufacturing environment is a highly dynamic environment. At the current time, no production control framework can be successfully applied to the collaborative manufacturing environment due to its constantly changing nature. While traditional production control approaches, such as MRP or just in time can only be applied to the manufacturing environment where long-term relationships among manufacturing entities exist and the domain of the control functions are fixed; the short-term characteristic of the collaborative manufacturing environment makes traditional production control approaches obsolete.

There are other problems associated with applying traditional production control approaches to the collaborative manufacturing environment, including the following.

(1) Traditional production control approaches usually make scheduling/dispatching decisions for sites/lines using information within them (Amrine *et al.* 1987). The decisions were made without a global perspective, which would have considered the production status of all parts of the final product as a whole. When using traditional production control approaches in a collaborative manufacturing environment, the lack of coordination among production entities can result in low on-time delivery rates, long production lead-time, high inventory/WIP level and, most importantly, a long matching time.
(2) Production control methods with pre-set schedules are not flexible. The MRP system typically runs between once a day to once a week. It has limited ability to adapt to contingencies on the shop floor, such as major machine breakdowns or rush orders (Brown *et al.* 1996). The result of MRP can be even worse when the production sites/lines in the collaborative manufacturing environment are of different types, such as flow line and job shop.
(3) The just-in-time approach proved to be efficient for production systems where there was a constant demand for products with limited varieties. The central-satellite factory relationship is very rigid in such systems (O'Hare, 1990). These limitations make the just-in-time approach obsolete while the manufacturing environment in the future moves towards a high variety of products and short term collaboration, as revealed by the concept of collaborative manufacturing and virtual cooperation.

3.2. *The elevator manufacturing industry*

In this research we used a simulation model of a real-world multi-line elevator manufacturing system in Taiwan as a test bed. An elevator is a complicated product. It has an average of 20 000 parts existing in the bottom level of its bill of material. Figure 1 shows the major sub-assemblies in the top three levels of the bill of material (BOM) of an elevator. These three levels will be taken into consideration in this research.

The elevator production system under investigation is shown in figure 2. It consists of five major production lines to produce sub-assemblies with very different processing requirements. Each line has four to five production cells, each of which

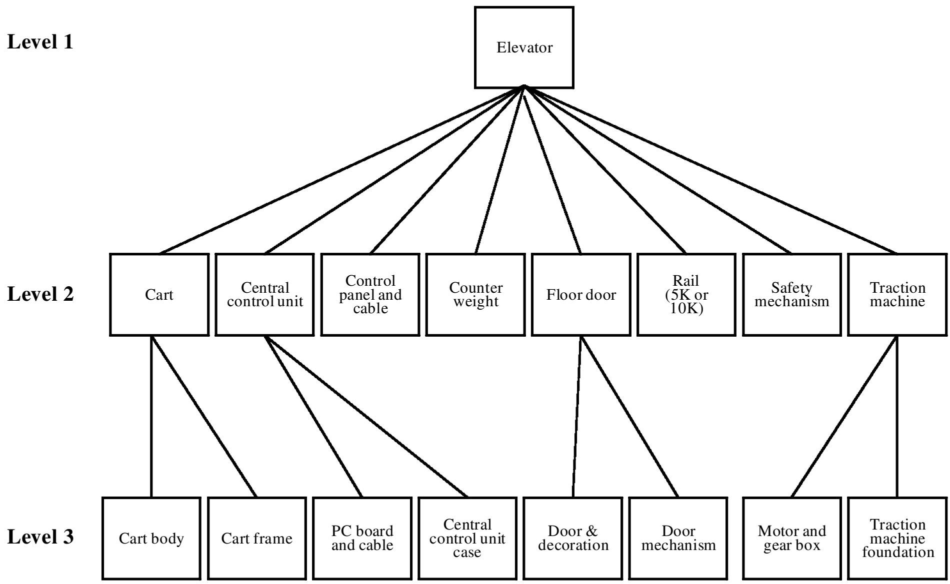

Figure 1. Top three levels bill of material of an elevator.

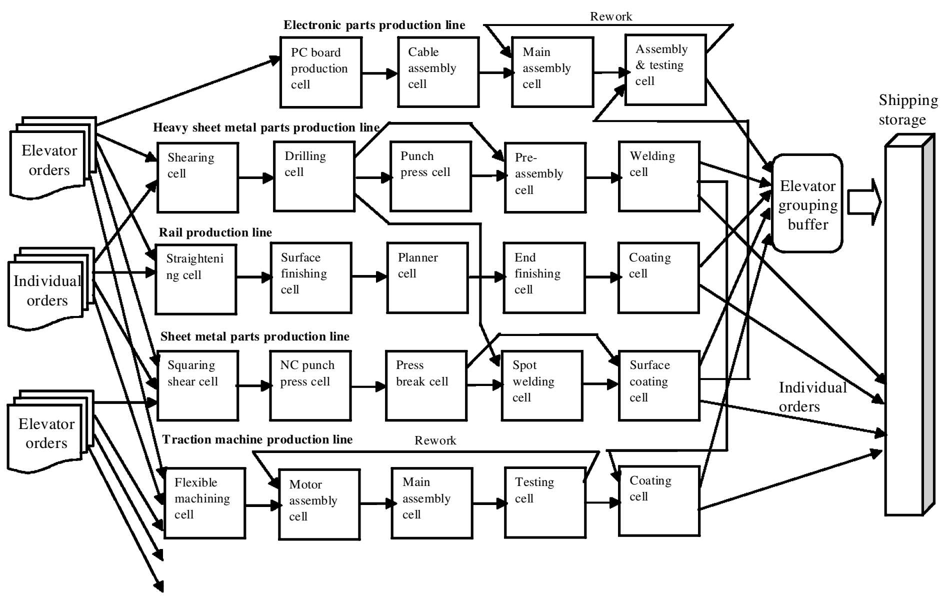

Figure 2. The elevator production system.

consists of a group of machines and a dedicated storage buffer. Certain machines are capable of producing multiple parts at the same time.

There are two types of orders from different customers. The first type is the elevator order, usually from a construction company. The second type is the individual sub-assembly order from other elevator manufacturers or from service divisions to satisfy maintenance requirements of previously sold products. Within these two order types there are five product types. They are: elevator in elevator order type and counterweight, floor door, rail and traction machine in individual order type. All products were made-to-order to match the number of stories, the size, and the styling design of the building, therefore each order only consists of a single product. All five lines process sub-assemblies in batches while one batch consists of only one sub-assembly type within a single order. However, to certain sub-assemblies, such as rail and floor door, the actual number of sub-assemblies in a batch varies mainly according to the number of stories and the complicity of the design of the order. While this system is capable of producing elevators from 2 storeys to 20 storeys, the processing times of batches even for the same sub-assembly type varies greatly. Certain sub-assemblies require sub-assemblies from more than one production line before they can be assembled in one of the lines. The final assembly of the elevator will not be performed until the elevator gets to the construction site. This elevator manufacturing system is a typical example for collaborative manufacturing because each of the five production lines is a profit centre and works on multiple projects, namely processing multiple orders, at the same time. Certain individual sub-assembly orders accepted are part of an elevator order outside the system.

One unique characteristic of this system is that all finished sub-assemblies of an elevator must be gathered in the grouping buffer to be grouped into a certain number of groups and then sent to the storage area waiting for shipping. There are several reasons for grouping the parts. First of all, there are a large number of parts in each elevator and grouping the parts prevents mismatch among orders, which is very costly. Secondly, these parts vary greatly in size, shape, weight and handling requirements, so they have to be grouped for fast and easy handling. In addition, there is a certain sequence for loading these groups of parts onto the shipping truck, so that they can fit into the limited space of the truck while considering the weight balancing of the truck, the protection of the parts, and the unloading process at the construction site, where only limited types of material handling equipment are available.

Due to the required grouping process, the matching time of an elevator order at the grouping buffer becomes a crucial factor for the performance of this production system. Long matching time can result in a large number of parts in the grouping buffer, waiting for the parts that are still missing. If the buffer is full, certain groups of parts have to be temporally stored in the shipping storage, for retrieval later for regrouping. This is a very length and costly procedure.

Short matching time is difficult to achieve by using current production control approaches such as MRP. This is because these approaches have limited abilities to adapt to system contingencies, such as machine breakdown and rush orders, while any delay of a sub-assembly of the elevator order in any cell will prolong the matching time. In addition, traditional production control approaches require a good estimation of part process time while the processing times for parts varies greatly depending on the number of stories of the building and the complicity of the customer design.

4. The simulation model

A simulation model of this elevator manufacturing system was built on a PC using the Extend V4 plus manufacturing module. Most of the data used in this simulation system were from the real-world factory while certain detailed numbers were not provided to ensure the confidentiality of the company. This system can produce an average of ten elevators per shift, not including individual orders, which are a significant portion of the total production. There are a total of 46 machines in the system. For all machines, scheduled maintenance and random breakdowns, which represent 10-15% of the time, were take into consideration. Certain parts will require rework after being inspected. Set-up time for machines and transportation time to the next production cell was incorporated.

Penalty functions, shown in tables 1–3, were designed to evaluate the effectiveness of different production control strategies used in this research. These functions were designed to reflect the real-world situation in the manufacturing system. Two types of penalties were taken into consideration. The first type is the matching penalty associated with the matching time, which represents the time duration from the first part of an order entering the grouping buffer to the final part entering the buffer and the grouping processes being finished. The second type is the deviation time penalty associated with the time deviation between the due date of an order and the actual finished time of the order. Here, we define deviation time to have a negative value when the order is finished before the due date and a positive value when the order is late. Both types of orders can suffer from a deviation time penalty while only elevator orders can suffer from a matching time penalty. Deviation time penalty functions are different for elevator orders and individual sub-assembly orders because it costs a lot more to store and handle an elevator than to store and handle an individual order, and the actual penalty from customers for a late elevator order is higher.

Matching time (M in minutes)	Elevator order matching time penalty function
$0 \leq M \leq 240$	Penalty $= 0$
$240 \leq M \leq 720$	Penalty $= 250 + 0.3(T - 240)$
$720 \leq M$	Penalty $= 250 + 144 + 0.5(T - 720)$

Table 1. Matching time penalty function for elevator order.

Deviation time (D in minutes)	Elevator order deviation time penalty function
$D \leq -240$	Penalty $= 25 + 0.1(T - 240)$
$-240 \leq D \leq 240$	Penalty $= 0$
$240 \leq D \leq 720$	Penalty $= 500 + 0.4(T - 240)$
$720 \leq D$	penalty $= 500 + 192 + 0.6(T - 720)$

Table 2. Deviation time penalty function for elevator order.

Deviation time (D in minutes)	Individual order deviation time penalty function
$D \leq -240$	Penalty $= 10 + 0.5(T - 240)$
$-240 \leq D \leq 240$	Penalty $= 0$
$240 \leq D \leq 720$	Penalty $= 200 + 0.2(T - 240)$
$720 \leq D$	Penalty $= 200 + 96 + 0.1(T - 720)$

Table 3. Deviation time penalty function for individual sub-assembly order.

5. Agent-based production control framework

5.1. *Architecture of the agent-based production control system*

The proposed production control framework is shown in figure 3. All production control issues in this control framework were conducted by intelligent software agents, which represent entities within this system. An entity in the system can be a sub-assembly, an order or a production entity. In the proposed framework, there are four types of agents: a line agent associated with each production line, a cell agent associated with each production cell, a sub-assembly agent associated with each sub-assembly, and an order agent associated with each elevator order.

Upon receiving of an elevator order, an order agent specifically for this order will be generated. The order agent will then generate sub-assembly agents for all sub-assemblies required in this order. These sub-assembly agents will inherit certain important information, such as the order number and which production entities will be used to produce the sub-assembly. An individual order with only one sub-assembly to be produced will be represented by its associated sub-assembly agent. Communications among agents shown in figure 3 will be established. After the order was finished, the order agent and sub-assembly agents will leave the system.

5.2. *Index values*

All agents in this system are capable of making decisions and determining certain indexed values according to the information they are carrying, the real-time information from the system, and index values from other agents. These simple index values—cell status (CS), matching status (MS), slack time (ST) and front loading time (FT)—were used to reflect certain real-time statuses of the manufacturing

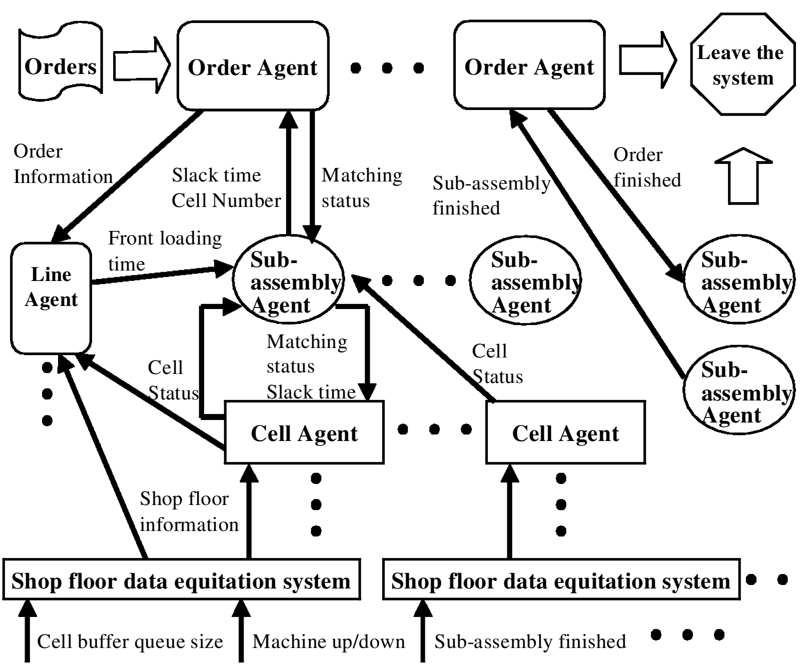

Figure 3. The agent-based production control framework.

system and used for information exchange among agents instead of using detailed data. CS combined the real time information of queue size and machine up/down in the cell to provide an index of how long a sub-assembly can expect to wait in the cell before it can be processed. MS represents the relative earliness or lateness of a sub-assembly to other sub-assemblies within the same elevator order. ST, as commonly defined, is the difference between the due date and the estimated finishing time of a sub-assembly. FT represents the timing of a sub-assembly to be loaded to the front of the production system.

One important reason for using index values is that many agents in the manufacturing environment can, and should, belong to different companies. It is impractical for these agents to exchange detailed production information, which is confidential, within a company. Detailed production information can potentially improve the quality of decisions in this system but considering the large number of agents in the system, the communication and computation load for both the agents and the network can be heavy. Using simple index values can greatly reduce the communication load and it is much easier for new members to join this production control framework while the problems of data formats and programming the agent were greatly reduced. Similar index values were also used for communication between an agent and other intelligent systems or devices in the environment.

5.3. *Structure and functionality of autonomous agents*

All agents in the system consist of three modules. First is the data module, which carries certain information in the data file for the use of the agent. Secondly is the communication module, which enables the agent to communicate with other agents and the data acquisition systems in the manufacturing environment. Lastly is the decision module, which makes decisions according to the information from the data module and communication module. Certain agents can reside in the same computer while other agents reside in remote computers, assuming that there are computer networks to provide all necessary communication functions required by these computers. It is also assumed that a data acquisition system is available to provide agents with real-time shop floor information in the system. The structure of agents and the structure of the proposed production control system are shown in figure 4. The functionality of the four types of agents is described in the following.

5.3.1. *Cell agent*

A cell agent represents a production cell in the system. It carries the following information in its data file: the cell identification number, the number of machines in the cell and their up/down status, current sub-assemblies in the cell and their time of entering the cell, current queue size (CQS), average queue size (AQS), and average waiting time (AWT). AQS is the average number of sub-assemblies in the storage buffer of the cell and AWT is the average time of a sub-assembly spent in the storage buffer before it will be processed.

The communication module of the cell agent will accept queries from sub-assembly agents and line agents and return the CS and AWT values to them. This module will keep communicating with the shop floor data acquisition system and will constantly monitor the current queue size in the storage buffer and up/down status of machines within the cell.

The decision module uses the information from the data module and communication module to make decisions. The cell agent decision module determines the

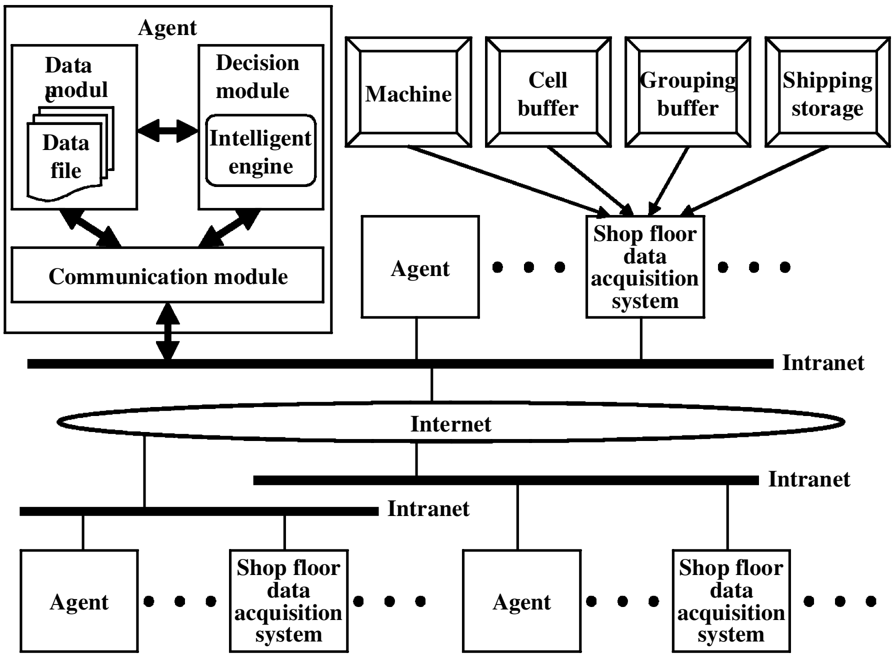

Figure 4. Structure of agents and the proposed production control system.

CS value for the cell, which provides a reference value for the sub-assembly agents and the line agents concerning how 'busy' this cell is. The CS value was determined by the following equation

$$\mathrm{CS} = \mathrm{A} \times \frac{\mathrm{CQS}}{\mathrm{AQS}} + \mathrm{B} \times \mathrm{MHS},$$

where A and B are constant and

$$\text{Machine status (MHS)} = \frac{\text{Total number of machines currently down in the cell}}{\text{Total number of machines in the cell}}.$$

The decision module also determines a priority value for all sub-assemblies in the storage buffer of the cell and this value will only be used within the cell. The priority was used to determine the sequence of these sub-assemblies being processed by the cell where the one with the lowest priority value will be processed first. The priority value will be determined for a sub-assembly under two circumstances: when it enters the storage buffer of the cell and when it has been waiting in the buffer for a time longer than AWT of the cell and still has not been processed. The cell agent will query the part agent for its ST and MS and then determined the priority as follows

Elevator order parts priority $= \mathrm{ST} + \mathrm{E} \times \mathrm{MS}$
Individual order parts priority $= \mathrm{ST}$
Reworked elevator order parts priority $= -10000$

where E is a constant.

5.3.2. *Sub-assembly agent*

A sub-assembly agent represents a sub-assembly in the system while this sub-assembly can be in the second level or in the bottom level of the BOM, as shown in figure 1. An individual order can be placed through the sub-assemblies on the second or on the bottom level of the BOM. Therefore, a sub-assembly agent can be instantiated by ether an elevator order or an individual order, while this sub-assembly that has been represented can be part of an elevator order, or part of an individual order, or it can be an individual order itself. As two of the sub-assemblies in the bottom level of the BOM merge into one sub-assembly in the second level of the BOM along the progress of production, the two sub-assembly agents will also merge.

The sub-assembly agent carries the following information in its data file: order number, sub-assembly type, sub-assembly specifications, matching status, remaining process time (RPT), and a list of subsequent cells in which it will be processed. RPT is an estimation of the total time remaining for the sub-assembly to be processed by subsequent cells.

The communication module of a sub-assembly agent will communicate with the line agent, the current and all subsequent cell agents, and its corresponding order agent. The sub-assembly agent accepts queries of its ST and MS values from the cell it is currently in. The sub-assembly agent will contact the cell agents of current and all subsequent cell agents to get their CS and AWT values and determine the ST value using the following formula:

$$\mathrm{ST} = \mathrm{TTDD} - \mathrm{ETTF},$$

where

$$\text{ETTF (Estimated Time To Finish)} = \sum_{\text{Current and all subsequent cells}} (\mathrm{AWT} \times \mathrm{CS}) + \mathrm{RPT}$$

and time to due date (TTDD) is the amount of time from now to the due date.

The MS value of the sub-assembly is determined by its parent order agent. When a sub-assembly agent has been queried for its MS value, it will send its ST value together with the query for the MS value to the order agent. The MS value will be determined by the order agent using this ST value. The sub-assembly agent will get the MS from the order agent and then pass it to the cell agent that needs it. If this sub-assembly belongs to an individual order, it will give an MS value of 0.

5.3.3. *Order agent*

An order agent represents an elevator order in the system. It was instantiated by a customer order of an elevator. The order agent carries the following information in its data file: order number, order specification, list of all sub-assemblies in the order and their specifications, list of cells in which these sub-assemblies will be processed, list of ST of all its sub-assemblies.

The communication module of the order agent will communicate with the line agent and its entire child sub-assembly agents. The order agent accepts queries of MS values from sub-assembly agents. The order agent will determine an MS value for one of its child sub-assembly agents upon a query. The MS value was determined using the following:

$$\text{MS} = \begin{cases} -15 & \text{if any one sub-assembly in the order has finished} \\ \sum_{\text{All other sub-assemblies in the order}} \text{MSI} & \text{if none of the sub-assemblies in the order has finished,} \end{cases}$$

where the matching status index (MSI) is a value resulting from comparing the slack time of current querying sub-assemblies (STC) with the ST of each of the other sub-assemblies in the order. MSI values were determined using the following:

$$\text{MSI} = \begin{cases} 0 & \text{if } \text{C} \leq (\text{STC} - \text{ST}) \leq \text{D} \\ 1 & \text{if } (\text{STC} - \text{ST} > \text{D} \\ -1 & \text{if } (\text{STC} - \text{ST}) < \text{C}, \end{cases}$$

where C and D are constant.

5.3.4. *Line agent*

A line agent represents a production line in the system. The order agent carries the following information in its data file: list of cells in the line with specification of the cells, queue size and machine up/down status in all these cells, and list of orders that will be processed in the line and specifications of the orders.

The communication module of line agents will communicate with order agents, sub-assembly agents, cell agents, and shop floor data acquisition systems to keep the information in the data file up to date. The decision module of line agents will determine the FT of a sub-assembly. FT represents the timing of a sub-assembly to be loaded to the front of the production lines. If an order was accepted with a TTDD shorter than the required time in system (RTS) of that type of order, it will be loaded to the production line immediately. If the TTDD of an order is longer than the RTS, the order will be held for a period of time before it will be loaded to the production lines. The value of RTS was found from historical data of the real-world factory, and the holding time (HT) was determined by the following:

$$\text{HT} = \begin{cases} \text{TTDD} - \text{RTS} & \text{if TTDD} > \text{RTS} \\ 0 & \text{if TTDD} \leq \text{RTS}. \end{cases}$$

In future studies, the FT should be determined according to the real-time information from the orders and the shop floor. The potential benefits could include better matching time, lower inventory levels, and higher on-time delivery rates.

5.4. *The agent-based production control methodology for a collaborative manufacturing environment*

The basic concept of the agent-based production control methodology is simple. The shorter matching time and better on-time delivery rate was achieved through using ST as priorities of sub-assemblies in a cell, while the ST was determined by real-time shop floor information. The priority was adjusted by considering the real-time status of all the other sub-assemblies within the same order.

The sub-assemblies in the storage buffer of a cell will be processed according to the sequence of their priority values, where the one with the smallest priority value will be processed first. For a sub-assembly of an elevator order, its priority in a cell will be adjusted by the MS, which represents the relative earliness or lateness of a sub-assembly in relation to other sub-assemblies within the same elevator order.

That means the progress of the sub-assembly will be sped up or slowed down according to the real-time progress of the other sub-assemblies in the same elevator order. The priority value of a sub-assembly will be determined every time it enters the storage buffer of a cell. Since the waiting time in a storage buffer of a cell can be relatively long, this priority value can become out-of-date due to the changes of many factors in the entire manufacturing system. Therefore, this priority value will be re-calculated according to the most up-to-date information if the subassembly has waited in the buffer for an amount of time longer than the AWT of the cell.

Basically, the cell agent serves to provide real-time cell information, determine priorities for sub-assemblies in the cell, sequence the sub-assemblies to be processed by the cell, keep track of all sub-assemblies in the cell and make sure their priority values are up to date. The sub-assembly agent serves to determine its ST and provide its current status in relation to the order agent for updating. The order agent keeps track of the current status of all sub-assemblies in the order and provides them with up-to-date matching statuses. The line agent considers the real-time status of the production line and determines the time for a sub-assembly to be loaded into the line. The following sections will describe the control methodology in a sequential manner, from accepting the order to finishing the production.

5.4.1. *Order accepting*

Every time an order is accepted, certain agents representing this order will be instantiated depending on the type of order. If an elevator order is accepted, an order agent for that order and all sub-assembly agents for the sub-assemblies to construct the elevator in the order will be instantiated, if it was an individual order for a sub-assembly in level 2 of the BOM in figure 1, which means this sub-assembly could consist of multiple sub-assemblies in level 3 of the BOM. In that case sub-assembly agents will be instantiated for only sub-assemblies in level 3. These sub-assemblies will merge into a sub-assembly in level 2 at certain points along the progress of the production and these sub-assembly agents will also merge into one agent to represent the sub-assembly after merger. If the order was an individual order of a sub-assembly in level 3 of the BOM, this order would contain only a single sub-assembly and, respectively, only the sub-assembly agent for this sub-assembly would be instantiated. All these agents will store certain detailed order information from the order in their data files. These sub-assembly agents will contact the line agents in which they will be processed and provide them with production requirements information. The line agent, after receiving production requirements information from the sub-assembly agents, will determine the FT for the sub-assemblies to be loaded to the production lines.

5.4.2. *Priority assigning*

Once a sub-assembly is loaded on to the production line, its priority will be determined every time it enters a production cell. The following steps will determine the priority of the sub-assembly within a cell.

Step 1. The cell agent will query this sub-assembly agent for its ST and MS value.

Step 2. After receiving the query, the sub-assembly agent will query cell agents of the cell it currently resides in and all the subsequent cells in which it will be processed through for CS values.

Step 3. All cell agents receiving the query will determine a CS value according to real-time information from the shop floor data acquisition system and return the value to the querying sub-assembly agent.

Step 4. After receiving all the required CS values, the sub-assemblies will determine the ST value. If this sub-assembly is part of an individual order, step 5 and step 6 will be skipped.

Step 5. This sub-assembly agent will send its ST value, together with a query for an MS value, to its parent order agent.

Step 6. The order agent uses this ST value to update its data file and determine an MS value for the sub-assembly agent. This MS value will be returned to the sub-assembly agent that asked for it.

Step 7. If this sub-assembly is part of an elevator order, the sub-assembly agent will return the MS values from the order agent together with the ST value to the cell agent that asked for them. If this sub-assembly is part of an individual order, the MS value will be 0.

Step 8. The cell agent determines the priority of the sub-assembly according to the ST and MS values returned by the sub-assembly agent.

Step 9. The priority value will be sent to the sub-assembly agent, and the sub-assembly agent keeps this value in its data file.

Step 10. The cell agent keeps track of all sub-assemblies in the cell. If a sub-assembly is in the storage buffer of the cell for an amount of time longer than the buffer's AWT, then the priority value will be re-determined by repeating step 1 through step 9.

To strengthen further the ability of the control function in pursuing shorter matching times and higher on-time delivery rates, the following adjustments to priorities will be made. First, if a sub-assembly needs to be reworked, that means it requires an unexpected amount of extra processing time. Therefore, this sub-assembly agent will be given a positive rework index value in its data file. All cell agents will give all sub-assemblies with this positive rework index value a priority value of −10 000, which is the highest priority value. Secondly, if any sub-assembly in an elevator order has been finished and is in the matching buffer waiting for the other sub-assemblies in the same order to arrive, it is necessary to speed up these unfinished sub-assemblies. Therefore, a positive sub-assembly finished index value will be send by the agent of the finished sub-assembly to its parent order agent. The order agent will give all other unfinished sub-assemblies in the same order an MS value of −15, which is the highest value of MS. This MS value can result in higher priorities for these unfinished sub-assemblies and hence speed them up.

5.4.3. *Order finishing*

After a sub-assembly of an individual order has finished and entered the shipping storage, the sub-assembly agent will leave the control system. When a sub-assembly of an elevator order has finished and entered the grouping buffer, a positive sub-assembly finished index value will be sent to its parent order agent. After the order agent finds that all sub-assemblies in this order are finished, it will send a positive order finished index value to all its child sub-assembly agents and all these agents for the finished order can leave the control system.

6. Control strategies applied to the simulation model

Control strategies with three different levels of collaboration were applied to the simulation model to compare their performances while being applied to the collaborative manufacturing environment. The first level is distributed control with no collaboration, the second level is centralized control with passive collaboration, and third level agent-based collaborative control with active collaboration. Level one and two reflect control strategies that have been applied in the real-world collaborative manufacturing environment under certain conditions, which will be discussed in the following sections. Level three is the control strategy proposed in this research.

6.1. *Level one: distributed control with no collaboration*

In the first level there were no collaborations among production entities in the system. This control strategy reflects the real-world manufacturing environment where each production line in the system is a company or profit centre itself with a goal to maximize its own profit. Within a production entity, production control issues were conducted without collaboration considerations, which consider the performance of producing the final product as a whole. Under this control strategy all orders will be loaded into the production line as soon as they were accepted. The priority of a sub-assembly in the storage buffer of a cell was assigned using the earliest due date (EDD), which means assigning the highest priority to the sub-assembly with the earliest due date.

6.2. *Level two: centralized control with passive collaboration*

In the second level there were passive collaborations among production entities. This control strategy reflects the situation where all five production lines belong to the same company and the company has control over the production lines with a goal to maximize the profit of the company as a whole. Without actively participating in making control decisions the production entities collaborate in a passive why by accepting decisions made by the centralized control mechanism. Under this control strategy the company will assign an FT value and a priority value to all accepted orders. All sub-assemblies within the same order will have the same FT and priority values. Only two different priority values, high and low, will be assigned to a sub-assembly in this control system and this priority value will be used in all cells within which the sub-assembly will be processed. The FT is an amount of time that has a constant value for each product type and is determined by the historical data of the real-world factory. If an order was accepted with TTDD shorter than FT, it would be assigned a high priority value and loaded to production lines immediately. If an order has a TTDD longer than FT, this order would be held for an amount of time before it would be loaded on to production lines with a low priority. According to the historical data of the real-world factory, there were about 30% of the total elevator orders and 50% of the individual orders that were assigned a high priority value. The HT value was determined by the same equation shown in section 5.3.4. By loading all sub-assemblies within an order on to the production lines at the same time and assigning them with the same priority values to be used throughout the manufacturing processes, the centralized control strategy serves to synchronize the sub-assemblies within an order and hence reduce the matching time penalty.

6.3. *Level three: agent-based collaborative control with active collaboration.*

The control strategy in the third level is what we proposed in this research. In this level the agents provided a means for production entities actively to join the decision-making processes with multiple goals to improve the performances of the entire system as well as the their own performances. As these goals might be contradictory to each other under certain circumstance, one goal might compromise others. For example, to reduce the matching time penalty of an elevator order, a production line might decide to give a sub-assembly in this order a higher priority, which results in other individual orders in this line being late. We shall see later in this research that we can satisfy certain multiple goals at the same time but the relationship among multiple goals and how to compensate one for the other in this system needs further research.

7. Simulation results

A simulation model of this elevator manufacturing system was built on a PC using Extend V4 plus a manufacturing module to evaluate the performance of the three different control strategies. Each of these three control strategies belongs to a different level of collaboration, as discussed in section 6, and was applied to the same model. Six random number seeds were used for each of the three control strategies to create a total of 18 simulation runs. The run time for each run is 57 600 minutes of simulation time. The statistics from the first 14 400 minutes, which is considered the warm-up period of the system, was disregarded.

7.1. *Performance index*

Under the same random number seed, different control strategies could result in slightly different total production quantities over the simulation period. Hence, using the total penalties accumulated by products being finished during a simulation run as performance indexes would be inappropriate. While trying to evaluate the performance of the system as a whole, it would also be biased to use the sum of the average penalty per order of the two types of orders because different order types account for different portions of the total production of the system. Therefore, in the following sections, we use total penalties of elevator orders per day (TPEO), total penalties of individual orders per day (TPIO), and total penalties of the system per day (TPOS) as performance indexes of a simulation run to compare the system performances among different control strategies. The TPEO, TPIO and TPOS were determined as follows:

$$\text{TPEO} = \frac{\text{Total penalty acquired by elevator orders in a simulation run}}{\text{Total number of elevator orders produced in a simulation run}} \times \text{OREO}$$

$$\text{TPIO} = \sum_{\text{All } i \text{ of individual order type}} \text{TPPT}(i)$$

$$\text{TPOS} = \text{TPEO} + \text{TPIO},$$

where

$$\text{OREO} = \text{order receiving rate of elevator orders per day.}$$

$$\text{TPPI}(i) = \text{Total penalty acquired by product type } i \text{ per day}$$

$$= \frac{\text{Total penalty acquired by product type } i \text{ in a simulation run}}{\text{Total number of product type } i \text{ being produced in a simulation run}} \times \text{ORIO}(i)$$

$$\text{ORIO}(i) = \text{order receiving rate of product type } i \text{ per day.}$$

$$i = \text{all product types within individual order type.}$$

OREO and ORIO(i) are constants for all simulation runs. TPEO, TPIO and TPOS are appropriate performance indexes because they represent the penalties that the system acquired by different order types or by the entire system within a day, while the daily production rates were constant for all simulation runs.

7.2. *Simulation results*

TPOS from the 18 runs are shown in figure 5. We can see that TPOS values were reduced for all six seeds while the level of control collaboration increases. The average TPOS reduction for the six seeds from no collaboration control to agent-based collaboration control was 53% and the reduction from passive collaboration control to agent-based collaboration control was 20%. A paired-sample t test has been performed to ensure that the difference between passive collaboration control and agent-based collaboration control was significant with 0.005 level of significance.

It is our observation that the agent-based collaboration control strategy can adapt to system contingencies and can serve to control the system to deliver a steady performance. We consider this ability to reduce variations in performances

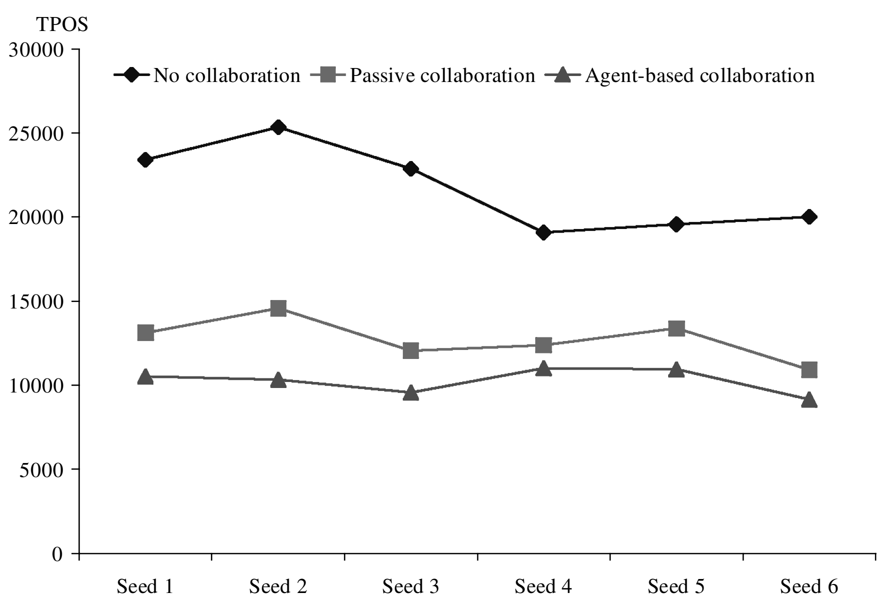

Figure 5. Total penalties of the system per day.

one of the important benefits of the proposed control strategy. From figure 5 we can see that the TPOS values of no collaboration control strategy vary greatly with the six random number seeds. The maximum TPOS value from seed 2 is 33% higher than the minimum value from seed 4. It implies that under the no collaboration control strategy the system performance is unstable and is subject to changes caused by random factors of the system, such as machine breakdowns, which can be as long as 3000 minutes in certain simulation runs. On the other hand the six TPOS values of agent-based collaboration control strategy form a much smoother line in Figure 5 where the maximum value from seed 4 is 20% higher than the minimum value from seed 4.

The TPEO values of the 18 simulation runs are shown in figure 6. These values showed a very similar pattern to TPOS in figure 5, in terms of both penalty reduction and performance variation reduction. That implies that the agent-based collaborative control strategy can conduct effective production control functions for elevator products.

The TPIO values of the 18 simulation runs are shown in figure 7. It shows that TPIO values were reduced for all six seeds while the level of control collaboration increases but we can see that the values have patterns that are contrary to TPOS and TPEO in terms of reducing performance variation. In figure 7, the TPIO values of no collaboration control strategy form a smoother line than those values of the agent-based collaboration control strategy. For the no collaboration control strategy, the maximum TPIO value from seed 6 is 6% higher than the minimum value from seed 3, while for the agent-based collaboration control strategy, the maximum value from seed 2 is 26% higher than the minimum value from seed 3.

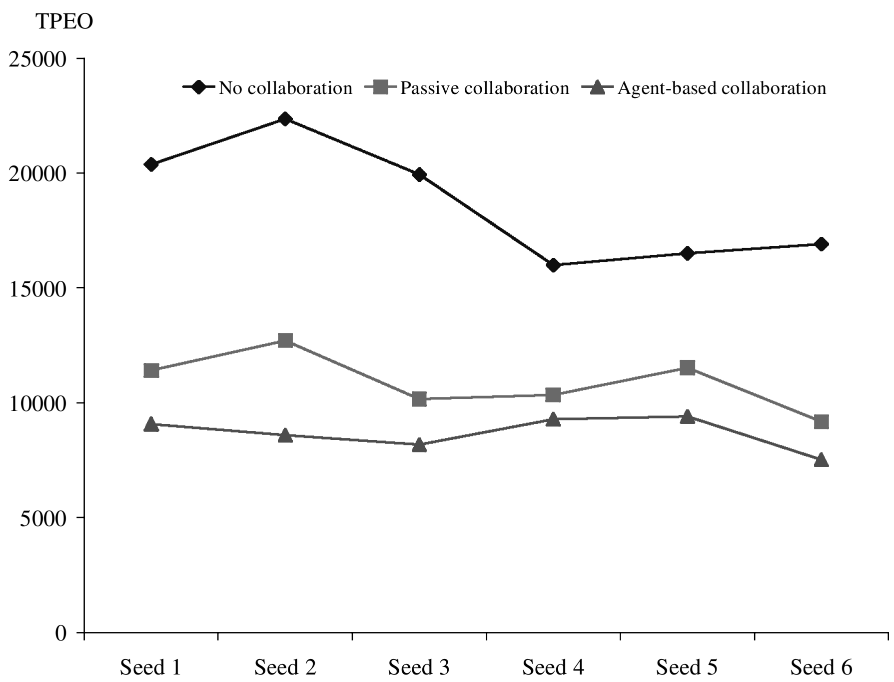

Figure 6. Total penalties of elevator orders per day.

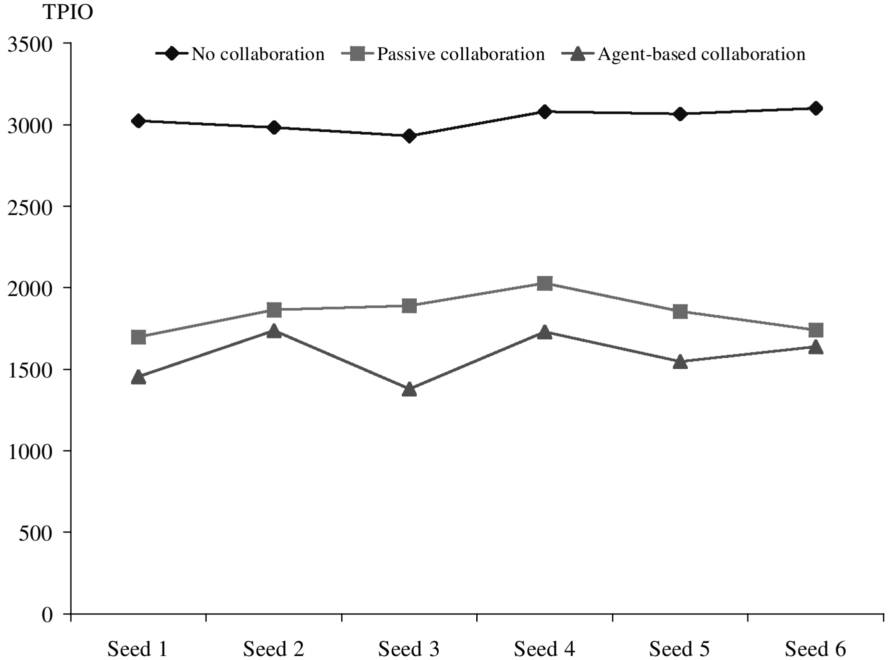

Figure 7. Total penalties of individual orders per day.

It was our belief that the reasons for the above observed patterns were that individual orders and elevator orders have different requirements for collaboration among production entities. An elevator order consists of a large number of sub-assemblies and requires collaboration among production entities to deliver lower total penalties and steady performance. However, within the individual order type a product type consists of only 1 or 2 sub-assemblies and hence requires little or no collaboration among production entities. The no collaboration control strategy does not consider collaboration among production entities; therefore, the performance of elevator orders was unstable due to system contingencies but the production of individual orders was left undisturbed. On the other hand, the agent-based collaboration control strategy enforced collaboration among production entities and took the matching status of elevator orders into consideration to adjust the priorities of sub-assemblies. The production of the sub-assemblies in these two types of orders will compensate each other's needs to deliver a better overall system performance. Therefore, the collaboration among production entities delivers a more steady performance for elevator orders but the performance for individual orders was subject to change. It should be emphasized here that under the agent-based collaboration control strategy, the fluctuation of TPIO values were compensated by TPEO values to deliver a better and more stable overall system performance, which was shown in figure 5.

8. Concluding remarks

This research constructed an agent-based control framework, which can be used in the collaborative manufacturing environment that has already been recognized and is practised in many sectors of manufacturing industry. This control framework can serve to provide better and more stable overall system performances, while the manufacturing facilities served to produce these parts can be different companies or profit centres in a collaborative manufacturing environment. This control framework is able to satisfy multiple goals, which are to deliver lower penalties for the whole system and the two different order types simultaneously.

This framework can be applied to any multi-part small batch discrete event manufacturing, while the information that agents contain and functions they perform can be different, depending on the nature of the manufacturing industry. More research should be performed in order to realize the objectives of an individual sub-assembly or a production entity. Certain functions can be added to the agents in the future, including the ability to communicate with human operators and to negotiate with each other to form an appropriate project team. Still, we consider the simplicity of the framework to be one of the major beauties of the proposed system. That is because, in practice, a company might have to join and leave a project in the collaborative manufacturing environment frequently. With these simple agents, the company can join the system without too much effort to program the agents and adapt to the control framework. Simple indexed values used in this framework serve to reduce communication and the computational load of the control system and keeps detailed production information confidential. The simplicity also serves to make this control framework more adaptable to different manufacturing industries.

Acknowledgements

This paper is based upon work supported by the Purdue Research Foundation and the National Science Foundation under NSF Young Investigator Award (Grant No. 9358158-DMI).

References

Amrine, H. T., Ritchey, J. A. and Moodie, C. L., 1987, *Manufacturing Organization and Management* (Prentice-Hall).

Balasubramanian, S. and Norrie, D. H., 1995, A multi-agent intelligent design system integrating manufacturing and shop-floor control. *First International Conference on Multi-agent Systems*, San Francisco, pp. 3–9.

Brown, J., Harhen, J. and Shivnan, J., 1996, *Production Management Systems—an Integrated Perspective*, 2nd edn (Addison-Wesley).

Genesereth, M. R. and Nilsson, N. J., 1987, *Logical Foundations of Artificial Intelligence* (Morgan Kaufman).

Crowston, K. and Malone, T. W., 1988, Intelligent software agents. *Byte*, **13**, 267–270.

Gu, P., Balasubramanian, S. and Norrie, D. H., 1997, Bidding-based process planning and scheduling in a multi-agent system. *Computers & Industrial Engineering,* **32**(2), 477–496.

Herrin, G., 1994, CIM perspectives—autonomous agent architectures for manufacturing. *Modern Machine Shop*, **67**(August) 64–65.

Ishida, T., 1995, Parallel, distributed and multi-agent production systems—a research foundation for distributed artificial intelligence. *First International Conference on Multi-agent Systems, San Francisco*, June, pp. 416–422.

Kempenaers, J., Pinte, J., Detand, J. and Kruth, J.-P., 1996, A collaborative process planning and scheduling system. *Advances in Engineering Software*, **25**(1), 3–8.

Khedro, T., 1996, Distributed problem-solving approach to collaborative facility engineering. *Advances in Engineering Software*, **25** (2–3), 243–252.

Lee, K. I., Kang, M. and Park, J. H., 1996, A collaborative scheduling system for made-to-order manufacturing. *CIRP Annals*, **45**(1), 461–464.

Lei, M., Yang, X., Tseng, M. M. and Yang, S., 1998, Design an intelligent machine center—strategy and practice. *Mechatronics,* **8**(3), 271–285.

Lin, G. Y. and Solberg, J. J., 1992, Integrated shop floor control using autonomous agents. *IIE Transactions*, **24**(3), 57–71.

Nof, S. Y. and Weill, R., 1992, Collaborative Coordination Control (CCC) of distributed multimachine manufacturing. *CIRP Annals*, **41**(1), 441–445.

O'Hare, G. M. P., 1990, Designing intelligent manufacturing systems: a distributed artificial intelligence approach. *Computer in Industries*, **15**(October), 17–25.

Shoham, Y., 1993, Agent oriented programming. *Artificial Intelligence*, **60**(1), 51–92.

Sikora, R. and Shaw, M. J., 1997, Coordination mechanisms for multi-agent manufacturing systems: applications to integrated manufacturing scheduling. *IEEE Transactions on Engineering Management*, **44**(2), 175–187.

Sluga, A., Butala, P. and Bervar, G., 1998, A multi-agent approach to process planning and fabrication in distributed manufacturing. *Computers & Industrial Engineering*, **35**(3–4), 455–458.

Tseng, M. M., Lei, M. and Su, C., 1997, A collaborative control system for mass customization manufacturing. *CIRP Annals*, **46**(1), 373–376.