Journal of Intelligent Manufacturing, 17, 111–131, 2006

Development and applications of holonic manufacturing systems: a survey

RADU F. BABICEANU and F. FRANK CHEN*

Center for High Performance Manufacturing, Grado Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA
E-mail: ffchen@vt.edu

Received August 2004 and accepted June 2005

This paper surveys the literature in the holonic manufacturing systems area in an attempt to bring together the key issues in the development and applications of holonic systems. A brief introduction presents the characteristics of today's manufacturing environment and the requirements for next generation manufacturing systems. Then, starting with considerations about the origins of the holonic concept and its first applications in manufacturing, the paper presents the advances made in applying the holonic concept to manufacturing systems area. Several considerations for the development of holonic manufacturing systems and specific holonic system requirements are discussed. As holonic concept is considered a solution for next generation manufacturing systems, there is a significant number of applications and implementations of the holonic concept in manufacturing systems domain. The most important and relevant approaches developed so far are presented. Finally, a short conclusion and future research directions in the area are provided.

Keywords: Manufacturing control architectures, holonic manufacturing systems, autonomy and cooperation, holonic system development, holonic system implementation

1. Introduction

Over the last two decades, a vast amount of research has been carried out, in both academia and industry, to improve the performance of manufacturing systems and their responsiveness to changing customer requirements. In addition to improve manufacturing systems productivity and their capabilities to deliver quality and low-priced products, in recent years, new requirements have been imposed on the operation of manufacturing systems. Examples of such new requirements include capability to respond in real-time to any disturbance in the system, true fault-tolerance, and hardware/software reconfigurability in reaction to changing environment.

1.1. *Background information on manufacturing systems research and implementation*

To deal with the increasing challenges faced by manufacturing sector appeared after the mass production concept reached its peak and started to decline in the 1970s manufacturing research looked for solutions in the newly emerging information technology (IT) area. Using IT tools, the computer integrated manufacturing (CIM) concept attempted to integrate all activities within a manufacturing facility, and was the concept that was believed, in the 1980s and early 1990s, to be capable of delivering the best solution for all manufacturing problems. However in most CIM implementation projects, it was "wrongly assumed that the organizational structure and the procedures of a company can automatically be improved by the implementation of computer aided information systems" (Warnecke, 1993). So, contrary to what was expected, CIM implementations resulted in rigid centralized systems, incapable of delivering the expected flexibility in response to changes of any nature. As a consequence in the 1990s CIM ceased to be viewed as the answer to all operational improvements needed in manufacturing environments.

*Author for correspondence.

1.2. *Next generation manufacturing systems*

Large scale projects carried out during the 1990s, such as Next Generation Manufacturing (NGM) and Visionary Manufacturing Challenges for 2020 (VMC2020), aimed at giving a framework to be followed by manufacturers in the United States to become successful in the global economy of the 21st century. A NGM company should "implement reconfigurable, scalable, cost-effective manufacturing processes, equipment, and plants that adapt rapidly to specific production needs" (Agility Forum, 1997). One of the grand challenges identified in the VMC2020 report is to "reconfigure manufacturing enterprises rapidly in response to changing needs and opportunities" (National Research Council, 1998). To obtain such reconfiguration, the report mentions the need for development of self-organizing manufacturing systems based on autonomous modules, biotechnology, and holonic concept.

According to Shen *et al.* (2001), the NGM systems will be more strongly time-oriented while still focusing on cost and quality. Requirements for realizing the time-oriented manufacturing systems include enterprise integration and cooperation, agility in responding to customer requests, scalability and fault-tolerance capabilities. While the first two requirements are driven by the global competition environment, the other two are characteristics required by the NGM for coping with unexpected events.

2. Manufacturing control architectures

The architecture of a system is defined by Bauer *et al.* (1994), as "a framework or a set of rules and guidelines for managing the development and operation of complex systems." This definition applies very well to manufacturing control architectures where the objective is to develop a framework and associated set of rules, usually in the form of software programs, to operate a manufacturing system. The published literature in the manufacturing control area (Dilts *et al.*, 1991; Overmars and Toncich, 1996) considers four basic types, namely centralized, hierarchical, modified hierarchical, and heterarchical control architectures. The big drawbacks of the first three are the long response time when unexpected events occur, the lack of fault-tolerance, and the complexity of the control software for large systems. The main disadvantage of heterarchical architectures is the unpredictability in behavior due to their completely distributed structure.

Considering the deficiencies of the control architectures presented above and the ineffective implementations of CIM, in the last decade, the research community introduced a number of new concepts for the design of manufacturing systems, such as bionic, fractal or holonic systems. Each of these concepts attempts to model a manufacturing system based on some analogies with existing natural, theoretical or social organization systems (Tharumarajah *et al.*, 1996). Intelligent software agents, first introduced in distributed artificial intelligence (DAI) field, were developed "due to the difficulties that have arisen when attempting to solve problems without regard to a real external environment" (d'Inverno and Luck, 2001). There is an obvious similarity with the need for introduction of agents in DAI systems and the need that led to the use of intelligent agents in manufacturing area. In both cases the systems developed did not consider the impact of the environment changes to the overall system behavior, and therefore the results obtained were unrealistic. The distributed architecture of multi-agent systems (MAS) and the agents' characteristics of autonomy and cooperation make MAS a suitable tool for the implementation of the bionic, fractal and holonic manufacturing concepts.

3. Holonic manufacturing systems concept

This section presents the origins of the holonic concept, its first applications in manufacturing, and gives the definition of the holonic manufacturing systems (HMS) and its constituent entities as developed within the Intelligent Manufacturing Systems (IMS) program.

3.1. *Origins of the concept*

The holonic concept originated from the work of Hungarian author and philosopher Arthur Koestler who tried to capture the behavior of complex systems by considering its constituent entities as being both wholes and parts at the same time (Koestler, 1968). To describe a basic unit of organization in biological and social systems, Koestler invented the word "holon", which comes from the combination of the Greek word for whole, "holos", and the suffix "on" meaning a part or a particle. As, within a social organization, holons behave "partly as wholes and wholly as parts" according to the way you look at them, Koestler proposed also the concept of open-ended hierarchy (OEH) as the architecture formed out of holons, called holarchy, which is not bounded in both downward and upward directions.

3.2. *Early applications of the holonic concept in manufacturing*

Japanese researchers were the first to apply Koestler's holonic concept to manufacturing area during the 1980s, in the design and implementation of a so-called holonic manipulator controller. Hirose *et al.* (1986) presented the design of the holonic manipulator controller, and then in another paper (Hirose *et al.*, 1988) the software environment of the holonic manipulator is described. The prototype implementation for the manipulator was presented in 1990 (Hirose *et al.,* 1990). Reported advantages of applying the holonic concept in designing the manipulator were a more robust design due to a decrease in the wiring complexity and an increase in reliability of the manipulator. The control software for the manipulator requires coordination between the built-in controllers by using dedicated task managers and message exchanging, typical for holonic control software as presented in the following sections.

3.3. *Holonic Manufacturing Concept Defined*

The idea of using the holonic concept in the design of manufacturing systems emerged in the early 1990s in the IMS program as a solution to cope with the increased rate of changes that was affecting the entire business world including the manufacturing sector. A consortium with researchers from Australia, Canada, Europe, Japan and the United States was established within the IMS program with the goal of developing the tools and implementing the holonic concept in real-world manufacturing industry, and thus obtain the potential benefits offered by holonic organizations such as "stability in the face of disturbances, adaptability in the face of change and efficient use of available resources" (Van Brussel *et al.*, 1998). To guide the research in the area, the HMS consortium participants established a series of working definitions for the constituent entities of the holonic systems (Christensen, 1994):

- *Holon:* An autonomous and cooperative building block of a manufacturing system for transforming, transporting, storing and/or validating information and physical objects. The holon consists of an information part and often a physical processing part. A holon can be part of another holon.
- *Autonomy:* The capability of an entity to create and control the execution of its plans and/or strategies.
- *Cooperation:* A process whereby a set of entities develops mutually acceptable plans and executes these plans.
- *Holarchy:* A system of holons that can cooperate to achieve a goal or objective.
- *HMS:* A holarchy that integrates the entire range of manufacturing activities from order booking through design, production, and marketing to realize the agile manufacturing enterprise.

4. Considerations on the development of holonic systems

The characteristics of individual agents and their relationships within distributed agent architectures make MAS theory appropriate for the implementation of HMS. Thus, HMS research is strongly related to the MAS research within the DAI community. As Ulieru *et al.*(2001) stated "multi-agent systems paradigm seems to be well suited to implementing a holonic abstraction of a problem which is fundamentally distributed in nature." However, there are two differences between the constituent entities in MAS and holonic systems. The first one is related to the aggregation of holons in the holonic architecture. Paolucci and Sacile (2005) observed that "holons in a holarchy are quite similar to agents in MAS, if one disregards the fact that a holon can contain other holons." The other difference comes from the modeling abilities of the constituent entities of the two types of systems. While agents are pure software entities, holons can include both hardware and software parts of the modeled system. Still, MAS are the only platform identified as the modeling tool for developing holonic systems. Table 1 gives a classification of the papers reviewed in the holonic systems area based on the main developmental aspect considered by the authors.

4.1. *Holonic system design*

A holon is an autonomous entity considered a whole that can include sub-holons having inherited characteristics, and in the same time it could be part of a broader holon to whom it can pass on some of its characteristics. Based on Koestler's holonic and OEH considerations, the aggregation of holons and their relationships within a holonic structure can be modeled as in Fig. 1 where the basic entity is referred as reference holon. This structure permits a clear distinction between all entities included in the architecture and allows the possibility to investigate a particular part of it without considering the entire structure.

Typically, a holon is formed by a physical processing unit and a software control unit, though, for holons not associated with manufacturing resources

Table 1. Classification of the papers reviewed considering their developmental characteristics and their broad area of application

	Manufacturing enterprises	*Shop floor control systems*
Holonic system design	Christensen (1994); Mathews (1995); Gou *et al.* (1998); Toh *et al.* (1998); Chirn and McFarlane (2000a); Chirn and McFarlane (2000b); Rahimifard *et al.* (1999); Toh *et al.* (1999); Leitao and Restivo (2000); Ulieru *et al.* (2000); Cheng *et al.* (2001); Fletcher and Deen (2001); Ulieru (2001a); Ulieru (2001b); Ulieru and Norrie (2001); Ulieru *et al.* (2001); Huang *et al.* (2002); Christensen (2003); Monostori (2003)	Gou *et al.* (1994); Ramos (1996); Bongaerts *et al.* (1997); Bruckner *et al.* (1998); Gayed *et al.* (1998); Gou *et al.* (1998); Rannanjarvi and Heikkila (1998); Sousa and Ramos (1998); Valckenaers *et al.* (1998); Van Brussel *et al.* (1998); Chirn and McFarlane (2000a); Chirn and McFarlane (2000b); Monostori and Kadar (1999); Silva and Ramos (1999); Sousa and Ramos (1999a); Sousa and Ramos (1999b); Zhang and Norrie (1999); Balasubramanian *et al.* (2000); Bongaerets *et al.* (2000); Cheung *et al.* (2000); Fletcher *et al.* (2000); Hammerle *et al.* (2000); Liu *et al.* (2000); Lun and Chen (2000); Shu *et al.* (2000); Arai *et al.* (2001); Balasubramanian *et al.* (2001); Brennan and Norrie (2001); Brennan *et al.* (2001); Bussmann and Sieverding (2001); Fletcher *et al.* (2001); Giebels *et al.* (2001); Heikkila *et al.* (2001); Sugimura *et al.* (2001); Vrba and Hrdonka (2001); Wang (2001); Heragu *et al.* (2002); Hsieh (2002); Wullink *et al.* (2002); Deen (2003); Heikkila *et al.* (2003); Johnson (2003); Marik *et al.* (2003); McFarlane and Bussmann (2003); Monostori (2003); Neligwa and Fletcher (2003); Ritter *et al.* (2003); Tamura *et al.* (2003); Babiceanu *et al.* (2004a); Babiceanu *et al.* (2004b); Bussman *et al.* (2004); Hsieh (2004)
Fault-tolerance	Fletcher and Deen (2001); Ulieru and Norrie (2001); Heragu *et al.* (2002); Christensen (2003)	Jarvis and Jarvis (2003); Johnson (2003); Neligwa and Fletcher (2003)
Real-time control	Christensen (2003)	Balasubramanian *et al.* (2000); Balasubramanian *et al.* (2001); Brennan *et al.* (2001); Zhang *et al.* (2000)
Production planning and scheduling	Gou *et al.* (1998); Toh *et al.* (1998); Chirn and McFarlane (2000b); Toh *et al.* (1999); Fletcher and Deen (2001); Ulieru and Norrie (2001); Ulieru *et al.* (2001)	Gou *et al.* (1994); Overmars and Toncich (1996); Ramos (1996); Bruckner *et al.* (1998); Gou *et al.* (1998); Rannanjarvi and Heikkila (1998); Sousa and Ramos (1998); Chirn and McFarlane (2000b); Silva and Ramos (1999); Sousa and Ramos (1999a); Sousa and Ramos (1999b); Zhang and Norrie (1999); Bongaerets *et al.* (2000); Cheung *et al.* (2000); Lun and Chen (2000); McFarlane and Bussmann (2000); Shu *et al.* (2000); Arai *et al.* (2001); Brennan and Norrie (2001); Bussmann and Sieverding (2001); Heikkila *et al.* (2001); Schoop *et al.* (2001); Sugimura *et al.* (2001); Hsieh (2002); Deen (2003); McFarlane and Bussmann (2003); Neligwa and Fletcher (2003); Tamura *et al.* (2003), Babiceanu *et al.* (2004a); Babiceanu *et al.* (2004b); Bussman *et al.* (2004)

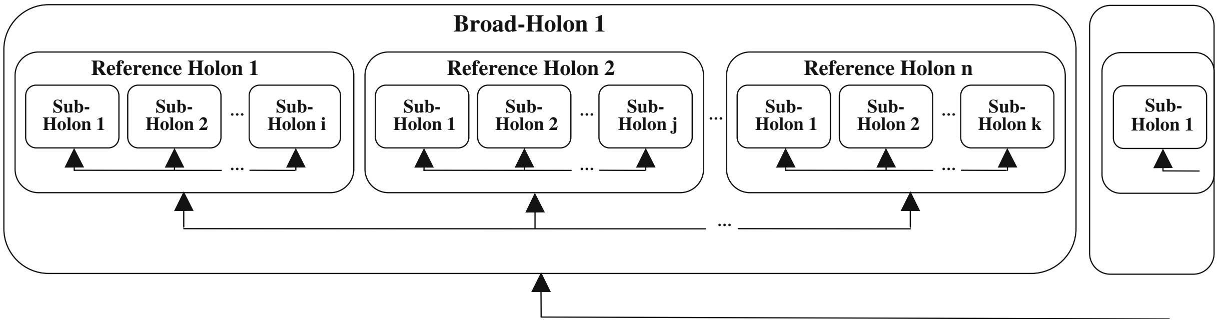

Fig. 1. The reference holon in a holonic architecture.

the holon comprises only the software control unit. To design the control architecture of holonic systems, two types of standards are proposed in the literature (Christensen, 2004). For the low-level control (LLC) architecture which refers to automation functions, it is proposed the International Electrotechnical Commission (IEC) 61499 series of standards for the use of function blocks in distributed industrial automation and control systems. Under the IMS project a significant progress was made for designing the holonic LLC architecture using the IEC 61499. This choice was made due to the capabilities offered by the above mentioned function blocks in terms of the “flexibility required in scheduling events, and especially in the case of hard real-time control” (Marik and Pechoucek, 2001). High-level control (HLC) architecture refers to inter-holon interactions and integration of the automation functions into holonic architecture. The collection of architectural standards for software agent systems developed under the Foundation for Intelligent Physical Agents (FIPA) are proposed in several papers as the solution to be used for designing the HLC architecture of holonic systems (Marik *et al.*, 2004).

Christensen (1994) developed a broader model of a holon which includes also a human unit functioning as a resource in the same way as the physical processing unit, but in the same time, it exchanges information with the environment and can act on the physical processing unit just like the software control unit. A more detailed view of a holon developed also under the HMS consortium program is presented by Fletcher *et al.* (2000). The software control unit, called intelligent control system (ICS) is further detailed to include all its building blocks, such as controllers and communication interfaces with other components of the holon. Based on the IEC 61499 function block standard the components of the ICS, their functions and interactions are presented in detail. The layered structure in the IEC 61499 function block provides the holons in the architecture the software portability, configurability and interoperability needed to act as autonomous and cooperative entities. The same approach of using function blocks to model the LLC architecture of holonic system appears in several other papers such as Wang *et al.* (1998b), Balasubramanian *et al.* (2000), Zhang *et al.* (2001), Christensen (2003), Deen (2003), and Neligwa and Fletcher (2003).

4.1.1. *Internal architecture of holons*

As stated above all holons have a control unit which is responsible for guiding holons towards accomplishing individual or group objectives. Based on the information and algorithms included in the control unit, a holon must be able to assess its status, react to changes in the environment, and decide the most appropriate actions to be performed which enhance its internal performance measure and move the holon closer to its objective. An internal database is used to store knowledge related to the holonic architecture, tasks to be performed, and manufacturing environment, needed to correctly evaluate and execute potential tasks. A schematic representation of the internal architecture of a holon in presented in Fig. 2.

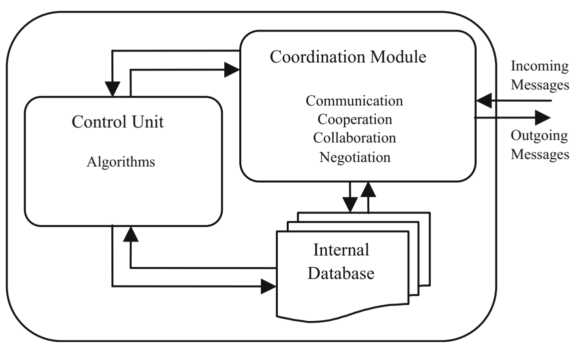

Fig. 2. Internal architecture of a holon.

4.1.2. *Autonomy and cooperation*

The HMS consortium definition considers autonomy and cooperation the two most important characteristics of holons. Glanzer *et al.* (2001) stated that "when consider the intelligent and cooperative part of a holon we find properties similar to software agents", thus the basic unit comprising MAS is a good choice to be used in the design of holonic systems. Moreover, Ulieru *et al.* (2001) noted that, "a system decomposition and analysis based on holonic principles naturally suggests a distributed software implementation, with autonomously executing cooperative entities as building blocks."

Autonomy allows holons to decide the actions needed to be taken such that their individual objectives are accomplished without consulting any supervisory entity. Cooperation is the characteristic that permits holons to agree on common plans and mutually execute them. It also aids holons to seek help in the case of a malfunction appeared after the start of the execution of the common plans.

4.1.3. *Relationships between entities in holonic systems*

The holons forming the holonic architecture need protocols and methodologies for exchanging information and coordinating their actions to accomplish individual or system wide objectives. Several terms are defined in MAS literature related to the potential relationships among entities in agent-based architectures, from which the most used are: coordination, communication, cooperation, collaboration, and negotiation. The definitions and the exact meanings of these four characteristics may differ from one work to another. Moreover, there is a clearly overlapping of these definitions as more than one paper is considered. As some of these relationships may need the accomplishment of others in order to take place, there could be also an overlapping of these relationships in the operation of MAS. A Venn diagram presenting these relationships and their overlapping characteristics is depicted in Fig. 3. The definitions and characteristics of these relationships presented below are developed after a comprehensive review of the available literature.

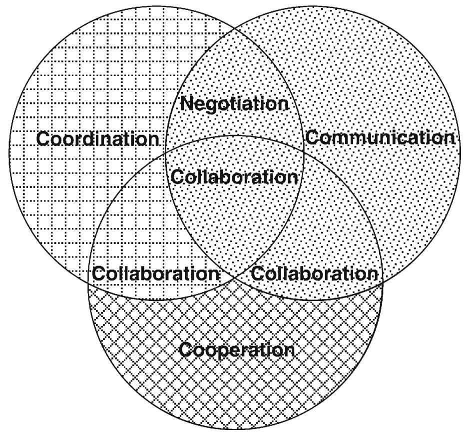

Fig. 3. Venn diagram showing the relationships among entities in a holonic system.

Coordination: Even a global optimal objective may not be achieved in a decentralized structure there might be several constraints that must be satisfied in order for the system to deliver a feasible solution. High-quality coordination moves the system towards satisfying these constraints. Thus, coordination primarily involves actions of the individual holons. If the overall objective function is time-dependent, the actions needed to be performed to achieve coordination become timed-actions. As d'Inverno and Luck (2001) stated, "coordination in MAS is necessary to avoid duplication of effort and obstruction in the actions of achieving common goals" and includes communication for exchanging information, cooperation between two or more holons for the purpose of achieving a common goal, and negotiation in the case of conflicting interests. Methods to achieve coordination in MAS that can be used when developing holonic systems include the use of entities having a global view of the system, use of direct supervisors having a much larger view of the system, mutual adjustments among agents in the architecture, or mediated coordination (Shen *et al.*, 2001).

Communication: Due to the decentralized architecture of the holonic systems, entities that comprise the architecture need to exchange information in order to satisfy system constraints and achieve system level goals. Communication is the principal means for exchanging information in MAS, and it mainly involves data. Communication enables coordination and cooperation, though both these processes can occur without communication, and it is a must in the negotiation process. Because MAS is the tool to implement holonic systems, communication between entities in holonic systems is based on the agent communication methodologies and protocols used in MAS. Methods of communication in MAS include message or plan passing, information exchanges through a shared data repository, or high-level communication (Shen *et al.*, 2001). Message passing methodology is based on exchange of messages among the entities in the architecture and is the most used approach for communication in MAS. Several protocols were developed for message passing from which the most known is the contract net protocol (CNP) developed by Smith (1980). CNP gives procedures for most communication needs within distributed agent architectures, such as task announcement and receiving, bidding mechanism, contract awarding and processing, and negotiations trade-offs (McFarlane and Bussmann, 2000).

Cooperation: Cooperation is the process, voluntarily or not, which results in common actions that move the entities involved in this process closer to a common goal. From this point of view, it can be considered that cooperation involves, primarily, common goal states. Doran *et al.* (1997) stated that cooperation in MAS means "to act with another or others for a common purpose or for common benefits." Tools to achieve cooperation, as presented in the DAI literature, include agent coalition and clustering, communication, sharing tasks and resources, and conflict resolution through negotiation (Shen *et al.*, 2001). When there is a voluntarily intent of exchanging data for the purpose of achieving a common goal, cooperation is sometimes referred as *collaboration* in the literature. But, basically, collaboration is a form of cooperation in which agents in MAS are voluntarily exchanging information for the purpose of achieving a common goal. Talukdar (1999) presented several collaboration rules among autonomous software agents, where collaboration is defined as "the exchange of data among information-processing agents, regardless of whether the exchange is productive or not," and a series of guidelines of how this rules can be extended in real-world real-time systems.

Negotiation: As the name implies, negotiation is the process of establishing an agreed set of actions that move a system or a part of it, from a conflicting situation towards a common goal. It principally involves intermediate or final system states. Depending on the architecture and the objective of the MAS, the negotiation process can be necessary between any types of entities in the architecture. In the holonic systems literature, negotiation is in some cases a term associated with task allocation. In this paper, negotiation refers strictly to conflict resolution, so negotiations protocols come into play when there is something to bargain about. Conflicts between holons can result from several reasons such as different objectives, different way of performing specific actions, or incorrect assessment of manufacturing resources capabilities. The conflict resolution techniques used for holonic systems are those used in MAS and include compromise negotiation where a solution is finally agreed by changing values along some dimension until a common point is achieved, integrative negotiation where a solution is found by identifying the common most important objectives of all agents, and game-theory based negotiations (Shen *et al.*, 2001).

4.2. *Task allocation*

The task allocation in holonic systems is the process of exchanging information among constituent entities and using it in internal algorithms and utility functions for the purpose of distributing tasks to the available resources. The most used task allocation approach for holonic systems as reported in the literature is the CNP, which uses a bidding process to assign each new task entering the system to the most appropriate resource. Fig. 4 presents the main steps of the CNP task allocation process.

As Smith (1980) stated, CNP is "a high-level protocol for communication among nodes in a distributed problem solver; it facilitates distributed control of cooperative task execution with efficient inter-node communication." The output of this methodology is a contract between the agent that is acting towards its goal, called manager agent, and the other agent, called contractor agent, which based on its internal configuration and set of goals is performing the actions requested by the manager agent. Probably the first to use CNP for task allocation in manufacturing was Parunak (1987) in his control architecture called YAMS (Yet another manufacturing system). The literature review identified several papers that employ CNP for distributing tasks in manufacturing applications in both holonic and agent-based systems (Gou *et al.*, 1994; Saad *et al.*, 1995; Ramos, 1996; Cantamessa, 1997; Shen *et al.*, 1997; Rannanjarvi and Heikkila, 1998; Sousa and Ramos, 1998, 1999 a,b; Ouelhadj *et al.*, 1999; Cardon *et al.*, 2000; Leitao and Restivo, 2000; Liu *et al.*, 2000; Unver and Anlagan, 2000; Usher and Wang, 2000; Vancza and Markus, 2000; Brennan and Norrie, 2001; Fletcher and Deen, 2001; Heikkila *et al.*, 2001; Mitidieri *et al.*, 2002; Hsieh, 2002, 2004; Neligwa and Fletcher, 2003; Tamura *et al.*, 2003; Babiceanu *et al.*, 2004a).

4.3. *Fault-tolerance in holonic systems*

When considering the design of manufacturing systems, Duffie (1990) noted that "one of the most difficult issues in the design of complex manufacturing systems is achieving fault-tolerance." This includes

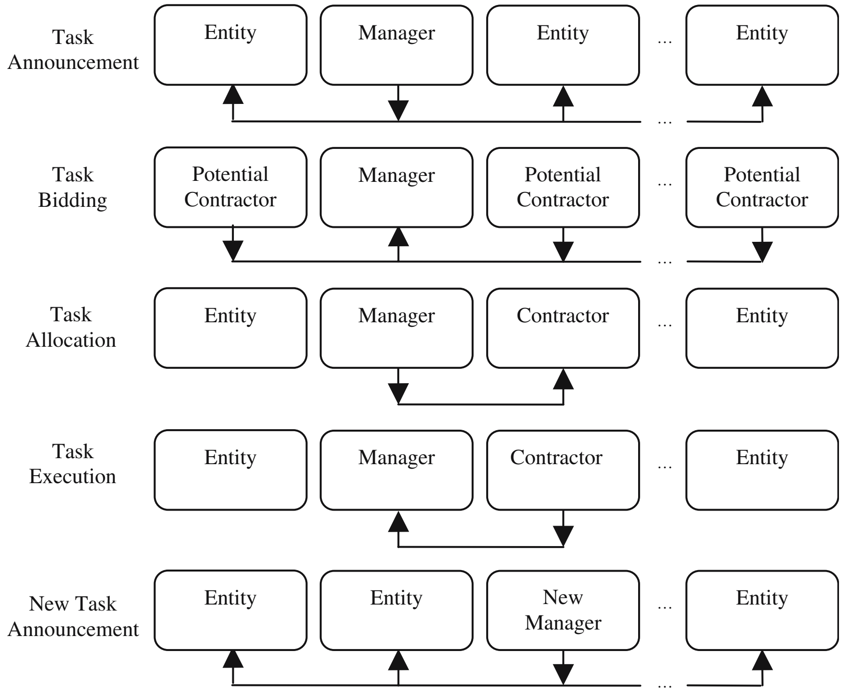

Fig. 4. The contract net protocol task allocation process.

automatic detection of failure situations, diagnosis of the cause of failure, and determination and implementation of appropriate recovery actions. Writing algorithms for every possible scenario is practically impossible. Some failure scenarios can be anticipated, and thus algorithms that take into account the status of the system after that particular failure can be written. But for complex systems, it is not possible to consider all the potential failures scenarios. Hatvany (1985) noted that "no algorithm can be written which foresee every possible failure mode of a highly complex system, nor can remedial strategies be deterministically designed for every situation."

Fault-recovery in HMS is addressed by a number of researchers using different methods. Ulieru and Norrie (2000) applied fuzzy modeling techniques to study the capabilities of holonic systems to recover in the event of occurring faults. The implementation of the proposed approach is capable of accomplishing fault-recovery by re-distribution of tasks in the case of resource failures among existing resources. Fuzzy logic techniques are applied also to study the self-organizing capabilities of holonic systems developed based on intelligent agents as presented in Ulieru *et al.* (2000). In the fault-tolerant HMS developed by Fletcher and Deen (2001), multiple entities use a cooperation framework for rescheduling affected tasks and recover the system from the effects of the unexpected event. Fault-tolerance is achieved by using error-generated information, a holon rescheduling mechanism, and a functional component failure recovery mechanism. There are other papers in the holonic manufacturing area (Christensen, 2003; Jarvis and Jarvis, 2003; Johnson, 2003; Neligwa and Fletcher, 2003) that consider the potential malfunctions that could appear during the operation of a manufacturing system. Such, Jarvis and Jarvis (2003) presented a model-based holonic diagnostic system for an automotive assembly line. Generally, in diagnostic systems the measures evaluated are the percent of the faults identified, also called fault coverage, and the execution time to identify and generate the diagnosis. By using heuristics to analyze the fault space, the fault coverage for the vehicle assembly station tested was reported at 95% with an execution time of less than one minute for each appeared fault.

4.4. *Software development*

Until a decade ago, software developers used a programming technique, called structural programming, having a pure hierarchical structure to develop computer programs in which master programs are calling up slave routines or subroutines (Mathews, 1995). More recently object-oriented programming (OOP) concept, instead of using a master-slave relationship to return data to the main program, uses objects that contain both data and functions not accessible from outside the object. With the aid of a coordinating program, the objects perform the required tasks (Mathews, 1996). There are many similarities between the OOP theory and the holonic approach for modeling manufacturing systems. Similar to the autonomy concept in holonic systems, the encapsulation concept in OOP provides autonomy for the objects defined within the software program. In OOP, there is no object or sub-program that can access all the data and variables within the classes. This is similar to the holonic architecture where there is no entity with a global view of the whole system that can have decision power over other entities in the architecture. The inheritance concept in OOP assures that objects in a class share common features and have access to the same external objects. In the holonic architecture, common features are specific for holons which are part of the same broader holon and all the holons part of the same broader holon can communicate with the others.

Even that entities in holonic systems can be modeled as objects, they are best portrayed using intelligent agents. Agents are a special type of software objects which have their own internal algorithms, use a common language for communication, and in contrast with objects, have the possibility to reason, interpret incoming messages, and take decisions according to its specific beliefs and objectives (Shen *et al.*, 2001). As software development is a continuously developing area, in the last years another software architecture concept emerged. The newly arrived concept of component-based development (CBD) tries to add reusability and reconfigurability into software modules. Using CBD concept and adapting it to the holonic systems theory, Chirn and McFarlane (2000a) presented an architecture called holonic component-based architecture (HCBA) that has the goal to cope with rapid changes in manufacturing environments.

Some of the software tools most used in the development and implementation of holonic systems are reported below. Unified modeling language (UML) is widely used to model individual holons and the interactions among them, examples of using UML can be found in Bruckner *et al.* (1998), Rannanjarvi and Heikkila (1998), Valckenaers *et al.* (1998), Van Brussel *et al.* (1998), Fletcher *et al.* (2000), Hammerle *et al.* (2000, 2001), Shu *et al.* (2000), Depke *et al.* (2002), and Huang *et al.* (2002). The CORBA architecture, a vendor-independent architecture that computer applications use to work together over networks appears also as a suitable modeling tool for holonic systems in papers such as that of Cheng *et al.* (2001). More recently a Java execution environment, JDPS, was adopted as a platform for developing holonic systems (Tamura *et al.*, 2003). Coming from the collection of FIPA standards, the agent communication language (ACL) is becoming increasingly used as the communication means in distributed architectures (Ritter *et al.*, 2003, Marik *et al.*, 2004). Besides using ACL, communication among entities in the holonic architecture is achieved also by using TCP/IP protocols (Shen *et al.*, 1997, Maturana *et al.*, 1999, Rannanjarvi and Heikkila, 1998, Arai *et al.,* 2001 and Heikkila *et al.*, 2001, Heikkila *et al*, 2003), Microsoft's COM/DCOM technology, which gives solutions for building entities that can communicate with each other (Brennan and O, 2000, Shu *et al.*, 2000, Unver and Anlagan, 2000, Schoop *et al.*, 2001), knowledge query and manipulation language (KQML), an agent communication language and protocol which has many similarities with the FIPA-ACL (Shen *et al.*, 1997; Maturana *et al.*, 1999; Leitao and Restivo, 2000; Heikkila *et al.*, 2001; Wang, 2001), or in the case of research systems, communication can be simulated using software objects residing in a single computer.

4.5. *Real-time issues in holonic systems*

According to Buttazzo (2002), real-time systems are defined as "systems that must react within precise time constraints to events in the environment." Thus, in real-time systems, not only the logical output has to be valid, but also the timing issues impose specific constraints. A real-time task is said to be hard when "missing its deadline may cause catastrophic consequences on the environment under control," whilst a real-time task is said to be soft when "meeting its deadline is desirable for performance reason, but missing its deadline does not cause serious damage to the environment and does not jeopardize correct system behavior." Both hard and soft real-time tasks appear in manufacturing systems. At the equipment level controllers, tasks such as positioning and transporting materials can be considered hard real-time since missing timing constraints could lead to damaging the entire system. The real-time tasks we deal in production planning and scheduling area are soft real-time tasks, since nothing catastrophic will occur by missing a deadline. Meeting deadlines for task scheduling in manufacturing systems is desirable and missing them only results in reduced solution quality.

A comprehensive discussion on the real-time requirements imposed on holonic systems can be found in Balasubramanian *et al.* (2001). Hard and soft real-time requirements are combined with other needed characteristics such as distributed, event-driven and intelligent control to develop a real-time distributed LLC system. Holonic systems are distributed by their nature, intelligent due to their software units, and event-driven as they react in response to changing environment, so these characteristics are appropriate to be included in the development of real-time holonic control architecture. Detailed design characteristics of the real-time distributed LLC system developed based on the IEC 61499 standards and its implementation can be found in Zhang *et al.* (2000).

Based on the OOP and agent-based approaches, another research undertaken by Brennan *et al.* (2001) considers the real-time issues in HMS and aims at developing reconfigurable distributed control systems capable of delivering real-time response to any service request coming in the system. Deadline control as an important evaluation of real-time capabilities is studied by Fletcher *et al.* (2001). They developed a holonic architecture with capabilities for meeting established deadlines and present several reconfiguration considerations for the design of manufacturing systems. A real-time control architecture viewed from the system, software, and functional architectures points of view is presented by Wang *et al.* (1998), followed by an event driven real-time distributed control system developed via using a combination of intelligent agents and IEC 61499 function blocks.

4.6. *Production planning and scheduling in holonic systems*

In holonic control systems, the computations needed for manufacturing scheduling applications are distributed among the control units of the holons comprising the architecture. This is in contrast to traditional manufacturing control where a single central processing unit (CPU) is performing all the calculations needed for a specific planning or scheduling application. Therefore, for complex problems, an improvement in the time necessary to obtain a valid schedule is obtained, due to the fact that the distributed holonic controllers are required to solve smaller problems compared to the complex problem need to be solved by the single CPU. Consequently, the holonic architecture allows for the possibility to attack large and complex problems which otherwise would be intractable in the traditional way. A valid solution is needed in many cases in real-time, a time-frame difficult to achieve when complex problems have to be solved on a single CPU. Even the solution obtained might not be optimal, by distributing the calculations over a number of controllers, real-time response can be achieved and complex problems become tractable.

There are two types of real-time scheduling. First uses the traditional scheduling theory and develops static schedules for all the jobs that need to be processed in a period. Then by using real-time information as it becomes available, works as a reactive system and revises the schedules to include the newly arrived information (Cowling and Johnson, 2002). The second approach is the application of software agent technology which constructs schedules using the existing resources whenever the status of the system changes. HMS is using this second approach for real-time allocation and scheduling of tasks.

A detailed description of a production planning and scheduling approach based on the CNP is presented by Vancza and Markus (2000). Starting with the order processing and task announcement, continuing with bidding mechanisms, task assignment and incentive-based mechanisms, and finishing with the final schedule generation and dispatching, all the steps of production scheduling are presented in detail. Many other papers (e.g., Gou *et al.,*1994, 1998; Ramos, 1996, Bruckner *et al.,*1998; Rannanjarvi and Heikkila, 1998; Sousa and Ramos, 1998, 1999a, b; Silva and Ramos, 1999; Zhang and Norrie, 1999; Bongaerets *et al.*, 2000; Cheung *et al.*, 2000; McFarlane and Bussmann, 2000, 2003; Shu *et al.*, 2000; Arai *et al.*, 2001; Brennan and Norrie, 2001; Heikkila *et al.*, 2001; Schoop *et al.*, 2001; Sugimura *et al.*, 2001; Hsieh, 2002, Huang *et al.*, 2002; Deen, 2003; Neligwa and Fletcher, 2003; Tamura *et al.*, 2003), considered the holonic approach to production planning and scheduling in manufacturing systems. Several of these papers are presented in the next section in the modeling of holonic enterprises and shop floor control systems discussion.

5. Applications and implementations of HMS concept

As holonic concept is considered a potential solution for NGM systems by many researchers in both academia and industry, there is a significant number of applications and implementations of the holonic concept in manufacturing systems. The most relevant ones as identified in the literature review are the focus of this section. Table 2 gives a classification of the papers reviewed based on the area of application and implementation of the holonic manufacturing concept.

Table 2. Classification of the papers reviewed based on the area of application of the holonic manufacturing concept

Applications of the holonic concept to existing manufacturing systems	*Holonic modeling of manufacturing enterprises*	*Holonic modeling of shop floor control systems*	*Holonic modeling of material handling and logistics systems*
Overmars and Toncich (1996); Gayed *et al.* (1998); Monostori and Kadar (1999); Chirn and McFarlane (2000a); Chirn and McFarlane (2000b); Lun and Chen (2000); Schoop *et al.* (2001); Jarvis *et al.* (2003); Monostori (2003)	Christensen (1994); Mathews (1995); Gou *et al.* (1998); Toh *et al.* (1998); Rahimifard *et al.* (1999); Toh *et al.* (1999); Leitao and Restivo (2000); Ulieru *et al.* (2000); Cheng *et al.* (2001); Fletcher and Deen (2001); Ulieru (2001a); Ulieru (2001b); Ulieru and Norrie (2001); Ulieru *et al.* (2001); Huang *et al.* (2002); Christensen (2003); Monostori (2003)	Gou *et al.* (1994); Ramos (1996) Bongaerts *et al.* (1997); Bruckner *et al.* (1998); Gayed *et al.* (1998) Gou *et al.* (1998); Rannanjarvi and Heikkila (1998); Sousa and Ramos (1998); Valckenaers *et al.* (1998); Van Brussel *et al.* (1998); Sousa and Ramos (1998); Silva and Ramos (1999); Sousa and Ramos (1999a); Sousa and Ramos (1999b); Zhang and Norrie (1999); Balasubramanian *et al.* (2000); Bongaerets *et al.* (2000); Cheung *et al.* (2000); Chirn and McFarlane (2000b); Fletcher *et al.* (2000); Hammerle *et al.* (2000); McFarlane and Bussmann (2000); Shu *et al.* (2000); Zhang *et al.* (2000); Arai *et al.* (2001); Balasubramanian *et al.* (2001); Brennan and Norrie (2001); Brennan *et al.* (2001); Cholski and McFarlane (2001); Fletcher *et al.* (2001); Giebels *et al.* (2001); Heikkila *et al.* (2001); Sugimura *et al.* (2001); Wang (2001); Heragu *et al.* (2002); Hsieh (2002); Wullink *et al.* (2002); Deen (2003); Heikkila *et al.* (2003); Jarvis and Jarvis (2003); McFarlane and Bussmann (2003); Neligwa and Fletcher (2003); Tamura *et al.* (2003); Hsieh (2004)	Cselenyi and Toth (1998); Liu *et al.* (2000); Bussmann and Sieverding (2001); Vrba and Hrdonka (2001); Heragu *et al.* (2002); Marik *et al.* (2003); Ritter *et al.* (2003); Babiceanu *et al.* (2004a); Babiceanu *et al.* (2004b); Bussman *et al.* (2004)

5.1. *Applications of the holonic concept to existing manufacturing systems*

Several attempts to apply the holonic principles to existing manufacturing systems such as flexible manufacturing systems (FMS) to improve their responsiveness to unexpected events and achieve fault-tolerance are presented in the literature. Overmars and Toncich (1996) suggested a method to apply the holonic approach to scheduling in FMS by dynamically selecting the appropriate manufacturing resources for any given workpiece coming into the system. The same research topic, applying the holonic concept for FMS scheduling, is considered by Cheung *et al.* (2000) and by Lun and Chen (2000). In the first paper a series of prototype holons are implemented for real-time scheduling tasks in an existing FMS, while in the second paper, the conceptual design developed based on the holonic approach is compared to the traditional FMS scheduling approach. Different disturbance scenarios are simulated and improvement results are reported. Jarvis *et al.* (2003) presented a migration process from an existing control system to an intermediate stage for a manufacturer of automotive engines and engine components. In the intermediate stage, the holonic framework and a part-oriented concept coexist with the existing infrastructure. The results obtained so far in implementing the intermediate stage show that more research is necessary in order to have all the tools needed for the migration to a true holonic system, the final objective of the research.

Various other strategies for migration of existing systems towards holonic systems are considered as a first and easier to implement solution and are presented in several papers. These strategies include the use of diagnostic capabilities and limited reconfiguration associated with existing systems (Gayed *et al.*, 1998), transitions of existing CIM systems to holonic systems using mapping of holons to existing CIM architecture (Chirn and McFarlane, 2000a), holonification of existing resources by incorporating control units for each resource (Monostori and Kadar, 1999; Monostori, 2003), or transitions of existing manufacturing cells to holonic cells (Chirn and McFarlane, 2000b).

5.2. *Holonic modeling of manufacturing enterprises*

Considering the definition of the holonic systems as holarchies in which holons are at the same time wholes and sub-wholes and that at any time a holon may be part of another broader holon, several papers in the literature cite the possibility to model the entire structure of an enterprise as a holonic system. There are also papers that present models of a chain of enterprises structured as a holarchy. This aspect can lead to the possibility to model the entire supply chain of an enterprise based on the holonic approach, for the purpose of developing holonic-based supply chain management systems.

Studies such as those of Ulieru (2001a, b) looked at the whole picture and illustrated manufacturing organizations as a network of enterprises grouped in a larger holonic enterprise. A shop floor is only a part of one of the enterprises involved in the larger holonic organization. This is a much larger view of the manufacturing picture and gives a good idea about the different levels of the holonic system. Even it looks like the holons are organized into levels of hierarchy the general master-slave relationship existing in proper hierarchies does not apply to these holonic organizations. Leitao and Restivo (2000) discussed the multi-enterprise model and present it in a structure similar to the general holonic architecture. The holon characteristic of being a part of another holon (i.e., a holon can be then broken into several other holons, which can be also broken into several other holons), allows for reduced system complexity, and consequently a multi-enterprise organization can be detailed in successive layers down to the equipment level of a single enterprise.

Toh *et al.* (1998) presented a solution to model small to medium enterprises (SME) using the holonic approach. Particularly, their model refers to a small metal-working company, and it included three main holons, the executive holon responsible for the most important decisions within the company, the business holon responsible for customer orders, inventory and other business related issues within the plant, and the manufacturing holon comprising the production resources and the control units associated with them and responsible for delivering goods to the shipping department. All these holons communicate and cooperate through the aid of the holonic information system support. Moreover, the main three holons are then composed of multiple sub-holons each of them having a well-defined responsibility in its area. A more detailed modeling of this particular enterprise functionality and behavior is presented by Toh *et al.* (1999) where holonic principles are applied for the small enterprise and also for the detailed model of three holons part of the holonic enterprise, the production planning, machining, and purchasing holons.

Another model for a holonic SME is presented by Rahimifard *et al.* (1999). Executive, inventory, scheduling and manufacturing holons together with orders and manufacturing databases form the holonic architecture of the SME. Additionally, a production planning and scheduling method and its software implementation called "Distributed Autonomous Real-Time (DART) planning and control" is presented in the paper. Other solution for modeling enterprises as holonic organization include the formation of dynamic clusters of software and hardware entities created each time new tasks need to be executed within the enterprise (Ulieru *et al.*, 2001). "Patterns of holonic collaboration" are studied starting with the inter-enterprise level, going through the intra-enterprise level, and down to the physical machine level of the holonic architecture. Mechanisms for holonic collaboration are presented for inter-enterprise level where the focus is to move from the actual closed supply chain model to a more open system that allows dynamic collaboration among entities. Intra-enterprise level focuses on task allocation using reconfigurable software, and achieves fault-tolerance, while at the machine level the focus is on using distributed intelligent control technologies.

A framework for modeling virtual enterprises (VE) using the holonic approach is presented by Huang *et al.* (2002), where the VE is formed by a group of distributed, autonomous and cooperative member enterprises (ME). Each ME holon has four corresponding sub-holons, planning, scheduling, task and resource holons. These holons can be further detailed in smaller holons, if needed. For example the resource holon is further comprising factory, shop-floor, workstation and cell holons. UML is used to present the interactions between holons in the proposed framework. Just like in the holonic shop floor resource allocation, where scheduling is performed among autonomous and cooperative holons with advices received from a global view entity, at the virtual enterprise level, resource sharing is performed among autonomous and cooperative ME holons with assistance from the VE holon. Table 3 presents a comparison of three of the holonic frameworks for modeling manufacturing enterprises presented above.

5.3. *Holonic modeling of shop floor control systems*

In a manufacturing facility the shop floor includes all the equipment, devices and controllers that aid to transform raw materials into finished products. To efficiently control the activities on the shop floor, the shop floor control system (SFCS) needs to take into account all the orders and corresponding process plans received form production planning department and execute them according to predefined objectives which can include: an acceptable level of quality, timing constraints, pre-established throughput levels, or resources utilization constraints. A traditional hierarchical SFCS cannot change the schedules it delivers when some predefined system status is modified. A holonic SFCS, on the other hand, can cope with these changes, by having the possibility to add and delete the holons in the architecture as imposed by the status of the system at each particular time.

The characteristics of the holons as potential part of two or more holons simultaneously is considered in the holonic shop floor control architectures developed at the Instituto Superior de Engenharia do Porto by Ramos (1996), and Sousa and Ramos (1998, 1999a, b). The resource holons are simultaneously part of three holons, the production planning, scheduling, and process planning holons, and the task holons are simultaneously part of other two holons, the production planning and the scheduling holons. The same idea appears in Sousa and Ramos (1999a) where several functional units of the shop floor are modeled as holons and there are multiple intersections across the functional units in the architecture. Silva and Ramos (1999) modeled a dynamic scheduling manufacturing system using the same approach with scheduling, production planning and process planning holons comprising common elements for more than one holon. All these papers make use of two important properties of holons, first a holon can be formed of other holons, and second, that a holon can be part of several larger holons.

Product Resource Order Staff Architecture (PROSA) is a HMS architecture developed at PMA-KULeven (Van Brussel *et al.,* 1998) and aims to be used as reference for future HMS implementations. It gives the basis of the architecture design, defines the terminology within the architecture, presents the component entities and their responsibilities within the architecture, and also gives examples of test-bed implementations. Three basic holons are the main components of PROSA reference architecture namely, product, resource and order holons. Their responsibilities are covering the main aspects of manufacturing control, product and process related planning, shop floor resources, and manufacturing tasks respectively, making it a good modeling tool in the design of SFCS. The other type of holon used in PROSA, the staff holon, aids the basic holons with its global view of the system, and can deliver optimal plans for particular manufacturing situations.

Table 3. Comparison of holonic frameworks for modeling manufacturing enterprises

	Holonic framework		
Holonic characteristic	Holonic enterprise model developed at the University of Calgary (Ulieru and Norrie, 2000; Ulieru *et al.*, 2000, 2001; Ulieru, 2001a, b,)	Holonic enterprise model developed for small to medium enterprises at Loughborough University, UK (Toh *et al.*, 1998, 1999)	Holonic enterprise model developed by Huang *et al.* (2002)
Scope	Inter-enterprise and enterprise-wide control	Enterprise-wide control	Inter-enterprise control
Reference holons (sub-holons) types	Enterprise (resource knowledge, design, product model, order, resource scheduling, resource control)	Executive Business (purchasing, production planning) manufacturing (machine)	Virtual Enterprise (member enterprise (planning, task, scheduling, resource))
Holonic system architecture	Three levels: inter-enterprise, intra-enterprise, physical	Holarchy: reference holons contain sub-holons	Holarchy: reference holons contain sub-holons
Decision-making	Mediator holons included in each of the levels	Individual holons or sub-holons based on their functions	Member enterprise holons with mediation from virtual enterprise holons

Valckenaers *et al.* (1998) argued that PROSA, aiming at being reference architecture, is a good "starting point for the design and development of holonic manufacturing control systems." And, as suggested by authors, PROSA was applied by other researchers to develop holonic manufacturing control architectures and implement them in test-bed systems. As an example, Liu *et al.* (2000) used PROSA to develop the control architecture for an automated guided vehicle (AGV) system which will be discussed in one of the next sections. Another PROSA-based system was developed under the Manufacturing Control Systems Capable of Managing Production Change and Disturbances (MASCADA) project for a car body painting at a German plant (Bruckner *et al.*, 1998). In the MASCADA project, the PROSA reference architecture is the starting point for a more detailed system, which includes all the specific aspects of the car body paint facility. As the holon and agent terms are used in many situations interchangeably in the distributed manufacturing control architecture literature, the paper adopts the term agent, instead of holon for the basic entities in PROSA. Related to the MASCADA project, the ManAge manufacturing control architecture (Heikkila *et al.*, 2001) uses agent technology to develop a distributed control architecture and provides a clear differentiation of agents and their functions in the architecture, enables scalability, and gives a detailed framework of inter-agent and inter-process communication. Specific to ManAge is the implementation of a belief database that is shared by the agents in the architecture, giving it an improved reaction to changing environments.

EtoPlan (Engineer-to-order Planning) is a holonic architecture for manufacturing planning and control developed at the University of Twente in Netherlands designed to work within a manufacture-to-order environment (Wullink *et al.* (2002). The EtoPlan architecture is based on multiple and temporary holarchies of applicable resources, called applicability groups (AG). Each activity on the shop floor has an AG associated with it, which groups all the resources needed to perform that specific activity. EtoPlan aims at dealing with large amounts of uncertainty, caused by incomplete and unreliable information. A prototype software implementation and more information about EtoPlan can be found in Giebels *et al.* (2001). Table 4 summarizes the main characteristics of the EtoPlan, PROSA, and the holonic framework developed at Instituto Superior de Engenharia do Porto.

Shop floor control is usually the first stage for the application and implementation of the holonic concept to industrial situations. Cholski and McFarlane (2001) presented their vision of implementing the holonic approach in the chemical process industries. The paper presents both the particular characteristics of this industry and specific ways to implement the holonic principles. Rannanjarvi and Heikkila (1998) applied the holonic concept in the

Table 4. Comparison of holonic frameworks for modeling shop floor control systems

	Holonic framework		
Holonic characteristic	The Holonic framework developed at Instituto Superior de Engenharia do Porto, Portugal (Ramos, 1996; Sousa and Ramos, 1998, 1999a, b)	The holonic reference architecture (PROSA) developed at Katholieke Universiteit Leuwen, Belgium (van Brussel *et al.*, 1998, Valckenaers *et al.*, 1998)	The holonic architecture (EtoPlan) developed at University of Twente, The Netherlands (Giebels, *et al.*, 2001; Wullink, *et al.*, 2002)
Scope	Production planning and scheduling	Entire manufacturing system	Production Planning and Control
Reference holons (sub-holons) types	Task manager Production planning Task Resource	Basic: Product Resource Order Assistant: Staff	Applicability Group (AG): group of resources that are applicable for the execution of an order
Holonic system architecture	Holarchy: reference holons include sub-holons; a holon can be part of several broader holons	Holarchy: reference holons include sub-holons	Temporary holarchy: AG, Parent-AG, Peer-AG, Resource, Child-AG. Resource can be member of multiple AGs
Decision-making	Task holons Task manager holon deals with uncertainties	Basic holons Assistant holon provides advice only	AG controller

development of the control software for surface treatment robots. They present a very detailed software development process and include models and tools used in analysis of the requirements for holonic systems, models for system design, models for structure design, models for design of behavior, and implementation models. Chirn and McFarlane (2000b) developed an architecture structured on a modular mix of standardized, autonomous, cooperative and intelligent components capable to cope with rapidly changing environments. They implemented the architecture for a robot assembly cell and report an encouraging preliminary performance evaluation, which include system integration and communication infrastructure.

5.4. *Holonic modeling of material handling and logistics systems*

Most of the research in the HMS area did not focus specifically on the material-handling task, so material movement within a manufacturing system is not much presented as potential implementation for holonic-based systems. However, several examples that consider the material handling aspect are presented here. The agent-based implementation for a material handling system (MHS) presented in Vrba and Hrdonka (2001), Marik and Pechoucek (2001) and Marik *et al.* (2003) is formed of a sorting system that includes several conveyors, diverters and storage units and supports two different types of dynamic reconfigurations. First one, called light-weight reconfiguration, when a resource failure is detected, and the second one, called heavy-weight reconfiguration, when the architecture of the systems is changing by adding or removing some of its physical elements.

Based on PROSA, the holonic reference architecture discussed above, a number of researchers developed holonic control frameworks to operate different parts of manufacturing systems, but only a few considered the MHS specifically. Liu *et al.* (2000) developed a holonic control architecture for an AGV system by further detailing PROSA to incorporate the AGV system, capable of being robust in the presence of disturbances. The architecture developed uses CNP as the communication methodology, and simulation tools for performance evaluation. The reported results show that the holonic system can offer considerable improvement compared to the traditional hierarchical control.

In the hybrid manufacturing control architecture, placed between hierarchical and totally decentralized architectures, developed at Rensselaer Polytechnic

Institute (Heragu *et al.*, 2002; Kim *et al.*, 2004) the entities comprising the architecture are referred as both agents and holons interchangeably. The particularity of this architecture resides in the fact that it has an intermediate agent responsible for the decision-making in the control architecture. A higher level global view agent has the ability to deliver optimal schedules, while the lower level agents are responsible for preparing the individual schedules, but they need to get permission form the intermediate level agent. The intermediate level agent, called middle level guide agent, has a larger system view than lower level agents, so it acts as a coordinator for the actions of lower level controllers. The authors argued that this architecture is more suited to satisfy the new requirements imposed on manufacturing system operations since it enables communication and decision-making in both horizontal (as in a heterarchical structure) and vertical (as in a hierarchical structure) directions among the entities in the architecture. The architecture was used to model a gantry robot system for order picking operations.

Within the HMS Project, the Holomobiles work-package (Bussmann *et al.*, 2004) analyzed the limitations of the material flow system in existing engine assembly systems. The aim of the Holomobiles work-package was to optimize the material flow in industrial manufacturing systems served by AGV systems. Using the holonic concept a new material flow design was proposed. The new design demonstrated improvements in the robustness and volume flexibility of the assembly process. Cselenyi and Toth (1998) presented a discussion of the logistics related issues in holonic systems. The paper gives the necessary features for holonic systems and the requirements that need to be satisfied from the logistics point of view. Collecting the transport demands, determining the delivery deadlines and sequences, and providing the transportation means are only a few of the capabilities of the system proposed. In the holonic transport system presented in Ritter *et al.* (2003), the holonic AGVs are capable of deriving a demand for orders individually and at the system level the cooperation mechanism prevents and resolves the potential deadlock conflicts. The holonic control architecture for automated MHS developed at Virginia Polytechnic Institute and State University (Babiceanu *et al.*, 2004, 2004b) addresses specific requirements for NGM systems that include reliability, real-time scheduling, fault-tolerance and material handling hardware reconfigurability. The holonic system is capable of delivering quality feasible real-time solutions when different types of changes occur during operation. A comparison of

Table 5. Comparison of holonic frameworks for modeling material handling systems

	Holonic framework		
Holonic characteristic	The FIPA-compliant system for material handling control developed at Czech Technical University and the Rockwell Automation Research Center, The Czech Republic (Marik *et al.*, 2003, Marik and Pechoucek, 2001, Vrba and Hrdonka, 2001)	The hybrid manufacturing control architecture developed at Rensselaer Polytechnic Institute, USA (Heragu *et al.* 2002, Kim *et al.*, 2004)	The holonic control architecture developed at Virginia Polytechnic Institute and State University, USA (Babiceanu *et al.*, 2004a, b)
Scope	Material Handling Control	Production planning, control and scheduling	Production planning, control and scheduling
Reference holons (sub-holons) types	Workplace agent (node, storage, machine) Crossing agent (diverter, Intersection) MHS agent (conveyor)	Order/Part Machine/MHD Middle level Cell/System	Order Resource (machine, material handling, equipment) Global scheduler
Holonic system architecture	Decentralized, composed of autonomous and cooperative agents	Low level: Machine, MHD Middle level: Guide Higher level: Cell/System	Holarchy: Reference holons include sub-holons
Decision-making	Crossing agent	Order holons with permission from middle level holons	Order holons Resource holons: re-scheduling

the basic characteristics of this holonic framework with the ones developed by Marik and Heragu, mentioned above, is presented in Table 5.

5.5. *Industrial applications of holonic systems*

Presently, the industrial acceptance of holonic systems is relatively low due to reasons inherent to the holonic systems development process and as a consequence of companies' business strategies. So, to date, there are only a few real-world implementations of holonic systems. The holonic shot-blasting system presented in Heikkila *et al.* (2003) is a demonstration of how separate robots can both act autonomously and cooperate with each other to complete the need-to-be-done tasks. The Holomobiles project mentioned in the previous section and developed for existing engine assembly systems in the automotive industry offered robustness and scalability as additional resources can be added easily to the system, unprecedented achievements in existing assembly systems (Bussmann and Sieverding, 2001). Two other holonic industrial implementations reported in literature were mentioned briefly in the previous sections. The first one is the model-based holonic diagnosis developed for an Australian automotive assembly plant (Jarvis and Jarvis, 2003), and the second one is the industrial automated warehousing system developed by Heragu *et al.* (2002) to which an intelligent holonic scheduling and control framework is applied for order picking and replenishing.

6. Conclusions and future research

Holonic control, as stated by Bongaerts *et al.* (2000) "tries to merge the best properties of both hierarchical and heterarchical systems, namely, a high and predictable performance with a high robustness against disturbances and unforeseen changes," and therefore it is appropriate for NGM systems. As holonic concept is implemented using MAS, advances in MAS theory can offer valuable tools for future development and enhancement of holonic systems. Agent technology is a vast area of research and parts of it are especially promising for future application in the holonic systems theory. Learning from experience and interaction with the environment, learning from future by using simulation, self-organization of societies of agents, or different types of emergent behavior are just a few promising research areas within DAI having potential applications in holonic systems. As an example, Cardon *et al.* (2000) suggested that incorporating genetic algorithms as the driver for the agents in MAS and considering the artificial intelligence behavior motivation for agents, newly formed agents become a completely autonomous genetic entity that "can lead to a drastically new kind of emergence phenomenon in self-organizing multi-agent systems."

In theory the holonic control approach is considered a viable solution for the NGM systems. While the holonic system theory evolves and new systems are being developed and implemented in research centers, a needed step forward is to refine and prepare the new control systems for the big real-world challenge in which holonic systems must demonstrate an improved performance beyond that of traditional control systems. In this regard there are some open issues in both holonic control system design and implementation that need to be addressed to obtain a larger industrial acceptance. For the holonic control system design, McFarlane and Bussmann (2003) pointed out the need for proven design methodologies that can provide consistency and reliability in holonic systems, clearly evidence of improvements delivered by the holonic control mechanisms and migration to full holonic control algorithms. On the other hand, for the holonic control system implementation, McFarlane and Bussmann (2003) indicated the need for adapting the holonic control systems to be used with the available computing systems, and the need for standardization of data exchange, internal algorithms and architectures of the holonic control systems.

Acknowledgment

The work presented in this paper has been mainly supported by a grant from the Virginia Commonwealth Technology Research Fund, Strategic Institutional Enhancement Program (CTRF Grant number: SE2002-03) and a grant from US National Science Foundation (DMI-9996417).

References

Agility Forum. (1997) *Next Generation Manufacturing Project: A Framework for Action*. Agility Forum, Leaders for Manufacturing, Technologies Enabling Agile Manufacturing.

Arai, T., Aiyama, Y., Sugi, M. and Ota, J. (2001) Holonic assembly system with plug and produce. *Computers in Industry*, **46**, 289–299.

Babiceanu, R. F., Chen, F. F. and Sturges, R. H. (2004a) Framework for the control of automated material handling systems using the holonic manufacturing approach. *International Journal of Production Research*, **42**(17), 3551–3564.

Babiceanu, R. F., Chen, F. F. and Sturges, R. H. (2004b) Reactive scheduling of material handling devices using the holonic control approach. *Proceedings of the Fourteenth International Conference on Flexible Automation and Intelligent Manufacturing*, Toronto, Canada, 1210–1217.

Balasubramanian, S., Zhang, X. and Norrie, D. H. (2000) Intelligent control for holonic manufacturing systems. *Proceedings of the Institute of Mechanical Engineers B*, **214**, 953–961.

Balasubramanian, S., Brennan, R. W. and Norrie, D. H. (2001) An architecture for metamorphic control of holonic manufacturing systems. *Computers in Industry*, **46**, 13–31.

Bauer, A., Bowden, R., Browne, J., Duggan, J. and Lyons, G. (1994) *Shop Floor Control Systems—From Design to Implementation*, Chapman & Hill, London.

Bongaerts, L., Monostori, L., McFarlane, D. and Kadar, B. (2000) Hierarchy in distributed shop floor control. *Computers in Industry*, **43**, 123–137.

Brennan, R. W. and O. W. (2000) A simulation test-bed to evaluate multi-agent control of manufacturing systems. *Proceedings of the 2000 Winter Simulation Conference*, 1747–1756.

Brennan, R. W. and Norrie, D. H. (2001) Evaluating the performance of reactive control architectures for manufacturing production control. *Computers in Industry*, **46**, 235–245.

Brennan, R. W., Fletcher, M. and Norrie, D. H. (2001) Reconfiguring real-time holonic manufacturing systems. *Proceedings of the 12th International Workshop on Database and Expert Systems Applications*, 611–615.

Bruckner, S., Wyns, J., Peeters, P. and Kollingbaum, M. (1998) Designing agents for the manufacturing control. *Proceedings of Artificial Intelligence and Manufacturing Research Planning Workshop—State of the Art & State of the Practice*, 40–46.

Bussmann, S., Jennings, N. R. and Wooldridge, M. (2004) *Multiagent Systems for Manufacturing Control: A Design Methodology*, Springer-Verlag, Berlin.

Bussmann, S. and Sieverding, J. (2001) Holonic control of an engine assembly plant—an industrial evaluation. *Proceedings of the 2001 IEEE Systems, Man, and Cybernetics Conference*, Tucson, AZ.

Buttazzo, G. C. (2002) *Hard Real-Time Computing Systems; Predictable Scheduling Algorithms and Applications*, Kluwer Academic Publishers, Norwell, MA.

Cantamessa, M. (1997) Hierarchical and heterarchical behavior in agent-based manufacturing systems. *Computers in Industry*, **33**, 305–316.

Cardon, A., Galinho, T. and Vacher, J.-P. (2000) Genetic algorithms using multi-objectives in a multi-agent system. *Robotics and Autonomous Systems*, **33**, 179–190.

Cheng, F.-T., Wu, S.-L. and Chang, C.-F. (2001) Systematic approach for developing holonic manufacturing execution systems. *Proceedings of the 27th Annual Conference of the IEEE Industrial Electronics Society*, 261–266.

Cheung, H. M. E., Yeung, W. H. R., Ng, H. C. A. and Fung, S. T. R. (2000) HSCF: A holonic shop floor control framework for flexible manufacturing systems. *International Journal of Computer Integrated Manufacturing*, **13**(2), 121–138.

Chirn, J.-L. and McFarlane, D. C. (2000a) A holonic component-based approach to reconfigurable manufacturing control architecture. *Proceedings of the 11th International Workshop on Database and Expert Systems Applications*, 219–223.

Chirn, J.-L. and McFarlane, D. C. (2000b) A component-based approach to the holonic control of a robot assembly cell. *Proceedings of 2000 IEEE International Conference on Robotics and Automation*, **4**, 3701–3706.

Chokshi, N. N. and McFarlane, D. C. (2001) Rationales for holonic manufacturing systems in chemical process industries. *Proceedings of the 12th International Workshop on Database and Expert Systems Applications*, 616–620.

Christensen, J. H. (1994) Holonic manufacturing systems: initial architecture and standards directions. *Proceedings of First Conference on Holonic Manufacturing Systems*, Hanover, Germany.

Christensen, J. H. (2003) HMS/FB Architecture and its implementation, in Agent-Based Manufacturing: Advances in the Holonic Approach, S. M. Deen (ed.), Springer-Verlag, Berlin, 53–87.

Cowling, P. and Johansson, M. (2002) Using real time information for effective dynamic scheduling. *European Journal of Operational Research*, **139**, 230–244.

Cselenyi, J. and Toth, T. (1998) Some questions of logistics in the case of holonic systems. *Journal of Intelligent Manufacturing*, **9**, 113–118.

Deen, S. M. (2003) An investigation into a computational model for HMS, *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, pp. 147–161.

Depke, R., Heckel, R. and Kuster, J. M. (2002) Formal agent-oriented modeling with UML and graph transformation. *Science of Computer Programming*, **44**, 229–252.

Dilts, D. M., Boyd, N. P. and Whorms, H. H. (1991) The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems*, **10**(1), 79–93.

Doran, J. E., Franklin, S., Jennings, N. R. and Norman, T. J. (1997) On cooperation in multi-agent systems. *Knowledge Engineering Review*, **12**(3), 309–314.

Duffie, N. A. (1990) Synthesis of heterarchical manufacturing systems. *Computers in Industry*, **14**, 167–174.

Fletcher, M., Garcia-Herreros, E., Christensen, J. H., Deen, S. M. and Mittmann, R. (2000) An open architecture for holonic cooperation and autonomy. *Proceesings of the 11th International Workshop on Database and Expert Systems Applications*, 224–230.

Fletcher, M., Brennan, R. W. and Norrie, D. H. (2001) Deadline control in holonic manufacturing using mobile agents. *Proceedings of the 12th International Workshop on Database and Expert Systems Applications*, 623–627.

Fletcher, M. and Deen, S. M. (2001) Fault-tolerant holonic manufacturing systems. *Concurrency and Computation: Practice and Experience*, **13**, 43–70.

Gayed, N., Jarvis, D. H. and Jarvis, J. H. (1998) A strategy for the migration of existing manufacturing systems to holonic systems. *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics*, **1**, 319–324.

Giebels, M. M. T., Kals, H. J. J. and Zjim, W. H. M. (2001) Building holarchies for concurrent manufacturing planning control in EtoPlan. *Computers in Industry*, **46**, 301–314.

Glanzer, K., Hammerle, A. and Geurts, R. (2001) The application of ZEUS agents in manufacturing environments. *Proceedings of the 12th International Workshop on Database and Expert Systems Applications*, 628–632.

Gou, L., Hasegawa, T., Luh, P. B., Tamura, S. and Oblak, J. M. (1994) Holonic planning and scheduling for a robotic assembly testbed. *Proceedings of the 4th International Conference on Computer Integrated Manufacturing and Automation Technology*, 142–149.

Gou, L., Luh, P. B. and Kyoya, Y. (1998) Holonic manufacturing scheduling: architecture, cooperation, mechanism, and implementation. *Computers in Industry*, **37**, 213–231.

Hammerle, A., Geurts, R. and Auinger, F. (2000) Internal communication architecture for resource holons. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, **3**, 1721–1726.

Hatvany, J. (1985) Intelligence and cooperation in heterarchical manufacturing systems. *Annals of CIRP*, **14**(1), 5–10.

Heikkila, T., Kollingbaum, M., Valckenaers, P. and Bluemink, G.-J. (2001) An agent architecture for manufacturing control: ManAge. *Computers in Industry*, **46**, 315–331.

Heikkila, T., Rannanjarvi, L., Sallinen, M. and Rintala, M. (2003) A holonic shot-blasting system, in*Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 255–302.

Heragu, S. S., Graves, R. J., Kim, B.-I. and St. Onge, A. (2002) Intelligent agent based framework for manufacturing systems control. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, **32**(5), 560–573.

Hirose, M., Ishii, T. and Ikei, Y. (1986) Development of holonic manipulator. *Proceedings of the Japan-USA Symposium on Flexible Automation*, 269–273.

Hirose, M., Ishii, T., Ikei, Y. and Amari, H. (1988) Software environment for holonic manipulator. *Advances in Flexible Automation and Roboticls*, **1**, 317–324.

Hirose, M. (1990) Development of the holonic manipulator and its control. *Proceedings of the 29th IEEE Conference on Decision & Control*, 91–96.

Hsieh, F.-S. (2002) Modeling and control of holonic manufacturing systems based on extended contract net protocol. *Proceedings of the American Control Conference*, 5037–5042.

Hsieh, F.-S. (2004) Modeling and control holonic manufacturing systems based on fusion of contract net and Petri nets. *Automatica*, **40**, 51–57.

Huang, B., Gou, H., Liu W., Li, Y. and Xie, M. (2002) A framework for the virtual enterprise control with the holonic manufacturing paradigm. *Computers in Industry*, **49**, 299–310.

d'Inverno, M. and Luck, M. (2001) *Understanding Agent Systems*, Springer-Verlag, Berlin.

Jarvis, D. H. and Jarvis, J. H. (2003) Holonic diagnosis for an automotive assembly line, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 193–206.

Jarvis, J., Jarvis, D. and McFarlane, D. (2003) Achieving holonic control—an incremental approach. *Computers in Industry*, **51**, 211–223.

Johnson, C. A. (2003) Towards a formalized HMS model, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 193–206.

Kim, B.-Y., Heragu, S. S., Graves, R. J. and St. Onge, A. (2004) Intelligent agent based control for warehouse control. *Proceedings of the IEEE 37th Hawaii International Conference on System Sciences*, 1–10.

Koestler, A. (1968) *The Ghost in the Machine*, Macmillan, New York.

Leitao, P. and Restivo, F. (2000) A framework for distributed manufacturing applications. *Proceedings of the Annual Conference of ICIMS-NOE, Bordeaux, France*.

Liu, S., Gruver, W. A., Kotak, D. and Bardi, S. (2000) Holonic manufacturing system for distributed control of automated guided vehicles. *Proceedings of the 2000 IEEE International Conference on Systems, Man, and Cybernetics*, **3**, 1727–1732.

Lun, M. and Chen, F. F. (2000) Holonic concept based methodology for part routing on flexible manufacturing systems. *The International Journal of Advanced Manufacturing Technology*, **16**, 483–490.

Marik, V. and Pechoucek, M. (2001) Holons and agents: recent developments and mutual impacts. *Proceedings of the Twelfth International Workshop on Database and Expert Systems Applications*, IEEE Computer Society, 605–607.

Marik, V., Pechoucek, M., Vrba, P. and Hrdonka, V. (2003) FIPA standards and Holonic Manufacturing, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 89–119.

Mathews, J. (1995) Organizational foundations of intelligent manufacturing systems—the holonic viewpoint. *Computer Integrated Manufacturing Systems*, **8**(4), 237–243.

Mathews, J. (1996) Orgnizational foundations of object-oriented programming. *Journal of Systems and Software*, **34**, 247–253.

Maturana, F., Shen, W. and Norrie, D. H. (1999) MetaMorph: an adaptive agent-based architecture for intelligent manufacturing. *International Journal of Production Research*, **37**(10), 2159–2174.

McFarlane, D. C. and Bussmann. S. (2000) Developments in holonic production planning and control. *International Journal of Production Planning and Control*, **11**(6), 522–536.

McFarlane, D. C. and Bussmann, S. (2003) Holonic manufacturing control: rationales, developments and open issues, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 303–326.

Mitidieri, C., Pereira, C. E. and Penz, L. L. (2002) Real time issues on coordinating cooperative autonomous objects. *Proceedings of the IEEE Seventh International Workshop on Object-Oriented Real-Time Dependable Systems*, 131–138.

Monostori, L. and Kadar, B. (1999) Agent-based control of manufacturing systems. *Proceedings of the Second International Conference on Intelligent Processing and Manufacturing of Materials*, **1**, 131–137.

Monostori, L. (2003) AI and machine learning techniques for managing complexity, changes and uncertainties in manufacturing. *Engineering Applications of Artificial Intelligence*, **16**, 277–291.

National Research Council (1998) Visionary Manufacturing Challenges for 2020 Report, *Committee on Visionary Manufacturing Challenges, Board of Manufacturing and Engineering Design, Commission on Engineering and Technical System of the National Research Council*, National Academy of Sciences.

Neligwa, T. and Fletcher, M. (2003) An HMS operational model, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 163–191.

Ouelhadj, D., Hanachi, C., Bouzouia, B., Moualek, A. and Farhi, A. (1999) A multi-contract net protocol for dynamic scheduling in flexible manufacturing systems. *Proceedings of the 1999 IEEE International Conference on Robotics & Automation,* 1114–119.

Overmars, A. H. and Toncich, D. J. (1996) Hybrid FMS Control Architectures Based on Holonic Principles. *The International Journal of Flexible Manufacturing Systems*, **8**, 263–278.

Paolucci, M. and Sacile, R. (2005) *Agent-Based Manufacturing and Control Systems: New Agile Manufacturing Solutions for Achieving Peak Performance*, CRC Press, Boca Raton, FL.

Parunak, V. D. (1987) Manufacturing experience with the contract net. *Distributed Artificial Intelligence*, M.H. Huhns (ed.), Pitman, London. 285–310.

Rahimifard, S., Newman, S. T. and Bell, R. (1999) Distributed autonomous real time planning and control of SME's. *Journal of Engineering Manufacture: Proceedings of the Institution of Mechanical Engineering Part-B*, **213**, 475–489.

Ramos, C. (1996) A Holonic Approach for task scheduling in manufacturing systems. *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, 2511–2516.

Rannanjarvi, L. and Heikkila, T. (1998) Software development for holonic manufacturing systems. *Computers in Industry*, **37**, 233–253.

Ritter, A., Braatz, A., Hopf, M., Schaeffer, C. and Westkamper, E. (2003) Transport agents - specification and development, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 241–253.

Saad, A., Kawamura, K., Biswas, G., Johnson, M. E. and Salama, A. (1995) Evaluating a contract net-based heterarchical scheduling approach for flexible manufacturing. *Proceedings of the IEEE International Symposium on Assembly and Task Planning*, 147–152.

Schoop, R., Neubert, R., Colombo and A. W. (2001) A multiagent-based distributed control platform for industrial flexible production systems. *Proceedings of the 27th Annual Conference of the IEEE Industrial Electronics Society*, **1**, 279–284.

Shen, W., Xue, D. and Norrie, D. H. (1997) An agent-based manufacturing enterprise infrastructure for distributed integrated intelligent manufacturing systems. *Proceedings of the Third International Conference on the Practical Application of Intelligent Agents and Multi-Agents*, 1–16.

Shen, W., Norrie, D. H. and Barthes, J.-P. A. (2001) *Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing*, Taylor & Francis Inc., London.

Shu, S., Wilkes, M. and Kawamura, K. (2000) Development of reusable, configurable, extensible holonic manufacturing system. *Proceedings of the 2000 IEEE International Conference on Systems, Man, and Cybernetics*, **3**, 8–11.

Silva, N. and Ramos, C. (1999) Infrastructures and scheduling method for holonic manufacturing systems. *Proceedings of the 1999 IEEE International Symposium on Assembly and Task Planning*, 442–447.

Smith, R. G. (1980) The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers*, C-**29**(12), 1104–1113.

Sousa, P. and Ramos, C. (1998) A dynamic scheduling holon for manufacturing orders. *Journal of Intelligent Manufacturing*, **9**, 107–112.

Sousa, P. and Ramos, C. (1999a) Distributed task scheduling. *Proceedings of the 1999 IEEE International Symposium on Assembly and Task Planning*, 436–441.

Sousa, P. and Ramos, C. (1999b) A distributed architecture and negotiation protocol for scheduling in manufacturing systems. *Computers in Industry*, **38**, 103–113.

Sugimura, N., Hino, R. and Moriwaki, T. (2001) Integrated Process Planning and Scheduling in Holonic Manufacturing Systems. *Proceedings of the 4th IEEE International Symposium on Assembly and Task Planning*, 250–255.

Talukdar, S. N. (1999) Collaboration rules for autonomous software agents. *Decision Support Systems*, **24**, 269–278.

Tamura, S., Toshibumi, S. and Hasegawa, T. (2003) HMS development and implementation environments, in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen (ed.), Springer-Verlag, Berlin, 209–240.

Tharumarajah, A., Wells, A. J. and Nemes, L. (1996) Comparison of the bionic, fractal and holonic manufacturing system concepts. *International Journal of Computer Integrated Manufacturing*, **9**(3), 217–226.

Toh, K. T. K., Newman, S. T. and Bell, R. (1998) An information systems architecture for small metal-working companies. *Journal of Engineering Manufacture: Proceedings of the Institute of Mechanical Engineers*, **212**(2), 87–103.

Toh, K. T. K., Harding, J. A. and Thompson, D. (1999) An holonic approach for the modeling of enterprise functionality and behavior. *International Journal of Computer Integrated Manufacturing*, **12**(6), 541–558.

Ulieru, M. and Norrie, D. (2000) Fault recovery in distributed manufacturing systems by emergent re-configuration: A fuzzy multi-agent modeling approach. *Information Sciences*, **127**, 101–123.

Ulieru, M., Stefanoiu, D. and Norrie, D. (2000) Holonic self-organization of multi-agent systems by fuzzy modeling with application to intelligent manufacturing. *Proceedings of the 2000 IEEE International Conference on Systems, Man, and Cybernetics*, **3**, 1661–1666.

Ulieru, M. (2001a) The holonic enterprise as a collaborative information ecosystem, *FIPA Meeting*, London, UK.

Ulieru, M. (2001b) Web-centric diagnosis and prediction system for global manufacturing. *Proceedings of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference,* Vol. 1.

Ulieru, M., Walker, S. S. and Brennan, R. W. (2001) The holonic enterprise as a collaborative information ecosystem. *Proceedings of the Workshop on Holons: Autonomous and Cooperating Agents for Industry*, Montreal, Canada.

Unver, H. O. and Anlagan, O. (2000) Design and implementation of an agent-based shop floor control system using Windows-DNA. *International Journal of Computer Integrated Manufacturing* (available at http://www.me.metu.edu.tr/imtrg/Papers/ijcimpaper.pdf) **15**(5), 413–427.

Usher, J. M. and Wang, Y.-C. (2000) Intelligent agents architectures for manufacturing control. *Proceedings of the 2000 Industrial Engineering Research Conference*, Cleveland, OH.

Valckenaers, P., Van Brussel, H., Wyns, J., Bongaerts, L. and Peeters, P. (1998) Designing holonic manufacturing systems. *Robotics and Computer-Integrated Manufacturing*, **14**, 455–464.

Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L. and Peeters, P. (1998) Reference architecture for holonic manufacturing systems: PROSA. *Computers in Industry*, **37**, 255–274.

Vancza, J. and Markus, A. (2000) An agent model for incentive-based production scheduling. *Computers in Industry*, **43**, 173–187.

Vrba, P. and Hrdonka, V. (2001) Material handling problem: FIPA compliant agent implementation. *Proceedings of the 12th International Workshop on Database and Expert Systems Applications*, 635–639.

Wang, L., Balasubramanian, S. and Norrie, D. H., Brennan, R. W. (1998) Agent-based control system for next generation manufacturing. *Proceedings of the 1998 IEEE International Symposium on Intelligent Control*, 78–83.

Wang, L., Balasubramanian, S. and Norrie, D. H., (1998b) Agent-based intelligent control system design for real-time distributed manufacturing environments. *Proceedings of the Agent-based Manufacturing Workshop—Autonomous Agents '98*, 152–159.

Wang, L. (2001) Integrated design-to-control approach for holonic manufacturing systems. *Robotics and Computer Integrated Manufacturing*, **17**, 159–167.

Warnecke, H. J. (1993) *The Fractal Company*, Springer-Verlag, Berlin.

Wullink, G., Giebels, M. M. T. and Kals, H. J. J. (2002) A system architecture for holonic manufacturing planning and control (EtoPlan). *Robotics and Computer Integrated Manufacturing*, **18**, 313–318.

Zhang, X. and Norrie, D. H. (1999) Dynamic reconfiguration of holonic lower level control. *Proceedings of the Second International Conference on Intelligent Processing and Manufacturing of Materials*, **2**, 887–893.

Zhang, X., Balasubramanian, S., Brennan, R. W. and Norrie, D. H. (2000) Design and implementation of a real-time holonic control system for manufacturing. *Information Sciences*, **127**, 23–44.

Zhang, X., Brennan, R. W., Yuefei, X. and Norrie, D. H. (2001) Runtime adaptability of a concurrent function block model for a real-time holonic controller. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, **1**, 164–168.