

An Approach for An Architecture to Embodied Procedural Reasoning

Fernando Orduña Cabrera and Miquel Sànchez-Marrè

Technical University of Catalonia (UPC)

Campus Nord-Building Omega Software Dept. (LSI)

Jordi Girona 1-3, 08034, Barcelona, Spain

{forduna, miquel}@lsi.upc.edu

Abstract. Some research in intelligent manufacturing systems summarizes the importance of developing new methods and techniques that should be more knowledge intensive, applied at the level of embedded devices. To bring a solution for this demand we propose an embedded architecture for micro-controllers based on the hypothesis is it possible to introduce intelligence in microcontrollers by applying some solutions from the area of Multiagent Systems and in particular Belief-Desires-Intentions (BDI) agents to model intelligent computational units that are physically embedded in the world. This approach was first formulated by Deepak Kumar and it is adopted as the basis of our research. This research is focused in the development of a BDI architecture which could provide flexible reasoning capabilities wich can cope with complicated tasks executed by an embodied system. The intelligent part is based in procedural reasoning -Belief, Desires and Intentions- (BDI).

Keywords. Intelligent Mechatronic System (IMS), Holonic Manufacturing System (HMS), Extended Machine State (EMS), RT-MESSAGE (Real Time MESSAGE), JADE, JADEX

1. Introduction

Europe may have a chance to stay competitive by exploiting emerging innovation in embedded systems to maintain manufacturing of high added value. As computer technology (embedded systems in particular) is becoming mature. Several works of research like [1,2,3,4,5,6] and some research reports like [7], formulated the need of develop new techniques for embedded intelligent methods.

Addressing the need for more agile and reconfigurable production systems has let to growing interest in new automation paradigms which model and implement production systems as sets of production units interacting in a collaborative manner in order to achieve a common goal.

The umbrella paradigm, encompassing this general form of system design, is to consider the set of intelligent production units as a conglomerate of distributed, autonomous, intelligent, pro-active, fault-tolerant and reusable units, which operate as a set of cooperating entities. Each entity typically constitutes hardware (the mechatronic part), control software and embedded intelligence.

Embedded systems for sensing, data acquisition, communication (wireless in particular) and control that establish the backbone for process control and machine automation, are key for the European process industries. Today, this industry is confronted with various challenges: Firstly, innovation in embedded systems and its seamless integration into new systems for process control and automation. Secondly, increasing amounts of process data arise from ever higher amounts of sensing and control data, mainly due to the increased complexity of novel products and production, [7].

Real-time control is becoming increasingly sophisticated, regulatory demands are becoming more exacting, and increasing expertise is needed beyond simple, reactive strategies. The main problem in adapting these solutions is low-level control is hard real-time nature. One solution is to locate both the realtime control and the deliberative higher-level decision making on the same computer-controlled device, thereby avoiding communication delays and networking problems, [4].

Other trends in integrating real-time control and operations management are aimed at top-down approaches starting from the operations management level, bottom-up approaches beginning with the physical operations and equipment and the building of simple, common strategies for agent deployment at any level. Proactive, deliberative approaches to realtime control. It is desirable to extend the deliberative, knowledge-intense agent-based solutions developed for the operations management level to real-time control, [4].

Customers want to be able to connect all their subsystems and equipment into the same, easily configurable, information system hence the need for standard protocols and real-time information flow at the lowest, i.e., embedded device, levels of a system. An intelligent distributed system is needed capable of integrating a variety of heterogeneous devices into an inter-operable network of resources.

To address this need control component vendors could develop smarter embedded devices and diversify their business operations in offer extended maintenance services via suitable information systems. There is the need for a strategic change from a one-way essentially product-oriented approach (e.g., the initial provision of control systems and machines) to two-way service-oriented offerings, including remote management, diagnostics, maintenance and even reconfiguration of complete automation systems.

Collaborative automation requires the use of distributed systems organised as dynamic peer-to-peer infrastructures, capable of autonomously restructuring in response to changes in business requirements. Such systems are required to modify their operational data and knowledge constantly in response to real-time changes in schedule, goals, and decision criteria.

Applied at the level of embedded devices, this approach enables entirely new automation architectures based on peer to peer interactions between autonomous devices. Incorporating such high-level technologies into low-level devices will gradually become feasible as the horsepower of embedded devices further increases, [1].

In order to overcome the interoperability challenges of the future manufacturing plant, a new paradigm is needed in terms of how intelligent components represent and process knowledge about themselves, about other components, and about the environment, [8].

This research is dived in different parts to imply the most important commentaries related from important issues which help us to identify the principal problem in the industry, and help us to find opportunities specially in resent research and try to do something

to solve it, just like a justify for this research. The last part we bring a short description about a new architecture to be research.

2. MAS and previous research

Agent based software systems are becoming a key technology for smart manufacturing control systems. A multi-agent based software platform can offer distributed intelligent-control functions with communication, cooperation, and synchronization capabilities; it also provides for the mechatronics components behavior specifications and the manufacturing system is production specifications, [3].

Manufacturing systems will constantly modify their operational data and knowledge in response to real-time changes in schedule, goals, and decision criteria, [2].

Increasingly intelligent embedded devices indicates how this approach can benefit the manufacturing industry and outlines the issues and approaches for cohesively coordinating manufacturing services at various levels of the manufacturing device hierarchy, [2].

The holonic-manufacturing community has been considering the problems of applying agents to decentralized, highly distributed control problems since 1993. In holonic-manufacturing-systems research, the vision of a "plug and operate" manufacturing unit led to the specification of holons as complete autonomous, cooperative manufacturing units. These units are intended to not only undertake tasks but also to accept, plan, schedule, and control tasks assigned to them. So, for example, a CNC (computer numeric control) machine holon would address the real-time control needs of a machine as part of a production cell and would support functions beyond the real-time domain. Such an environment, which requires distributed, self-contained entities to make decisions, is a logical application for agents. Most reported developments of holonic systems either explicitly or implicitly integrate agents into decision making (at the real-time control level or the operations management level). In a holonic environment, holonic elements might themselves comprise a set of simpler holons. So, a holon representing a specific CNC machine might consist of holons for product fixturing, part handling, and so on. These holons would coordinate to ensure the machine performs its assigned tasks effectively. Operations management decision-making problems at this level usually are not hard real time in that the decisions are not safety critical. However, a scheduling or planning decision is timeliness can influence production delays. The decision scope is reasonably broad, relating to production management, materials planning, and resource scheduling across a factory. Decision robustness is of fundamental importance for these areas no factory is immune from disruptions, whether from external or internal sources. Additionally, the computational complexity of decisions is a critical consideration for large operations. These two factors drive much of the research in this area, which is also often called intelligent scheduling. Intelligent-agent technology has been applied in this domain to problems such as distributed order preprocessing, production planning and scheduling processes, shop floor and workshop decision making, and financial management and billing, [4].

Future knowledge-based manufacturing. In order to overcome the aforementioned challenge, a new paradigm is needed in terms of how intelligent components represent and process knowledge about themselves, about other components, and about the envi-

ronment. The mandate for this new paradigm is that knowledge be made explicit and be given machine interpretable semantics. Through machine-based reasoning and inference, components previously unknown to each other can gain understanding about their respective skill sets, goals and interaction models, and therefore interact intelligently, [2].

2.1. Ontologies

Because of the need of MAS to manage increasing amounts of data, another crucial question is how individual agents can store and use knowledge locally?. For this purpose, agents wrappers include different acquaintance models. These organize, maintain, and explore social knowledge about other agents (their addresses, capabilities, load, reliability, and so on). To keep the temporary and semipermanent knowledge fresh, researchers have developed several knowledge maintenance techniques to significantly reduce communication during urgent decision making. The other natural requirement is to keep messages as short as possible while keeping the information content rich. For this purpose, knowledge ontologies help agents understand and correctly interpret the semantic meaning of the messages received. Finally, standards play an important role in agent-based deployments, of the several standardization initiatives in the multi-agent domain, the Foundation for Intelligent Physical Agents (www.fipa.org) is the most efficient and influential. FIPA provides numerous standards for organizing communication and negotiation, for creating and maintaining ontologies, and so on, [4].

The abstraction level provided by the middleware should enable knowledge acquisition in real time. Ontology management and description logics can provide necessary reasoning mechanisms. The Semantic Web Services technology is a candidate solution to enable plug-and-play capabilities for agent-based systems, [7].

The use of ontologies and explicit semantics enable performing logical reasoning to infer sufficient knowledge on the classification of processes that machines offer, and on how to execute and compose those processes to carry out manufacturing orchestration autonomously, [8].

2.2. A Case Using Implementation of a Real Time Methodology

In the research [9] and [10], an application case was developed using the methodology of Vicente J. Inglada RTM and they concluded, that methodology was chosen assuming others because it is the best for the application. The analysis and design developed on [9] was implemented on [10]. In that investigation they argue “the methodology is helpful to divide the problem but not give a complete solution especially because itself constraint programming of software application. In these sense there are not a clear way to develop it”. To solve that, they propose to unify a function of FSM like inference mechanism and programming reactive JADE agents achieving a final application.

Then on [11] say, there are not a clear way on the methodology to go from design to implementation following the steps of methodology. To solve it propose a new design using an inference mechanism based on graphs, these yields a solution using RT-MESSAGE and JADE.

Summary the propose of research as [9,12,11,13,10], propose a solution for a Real Time problem in the project SCAIZA.

3. Intelligent Mechatronic System (IMS)

As is depicted in the Figure 1, a mechatronic system is integrated by the join of mechanic, electronic, and some kind of information system.

The information processing within mechatronic systems may range between simple control functions and intelligent control. An intelligent control system is organized as an on-line expert system and comprises:

1. Multicontrol functions (executive functions);
2. Knowledge base;
3. Inference mechanisms;
4. Communication interfaces.

The on-line control functions are usually organized in multilevels, as already described.

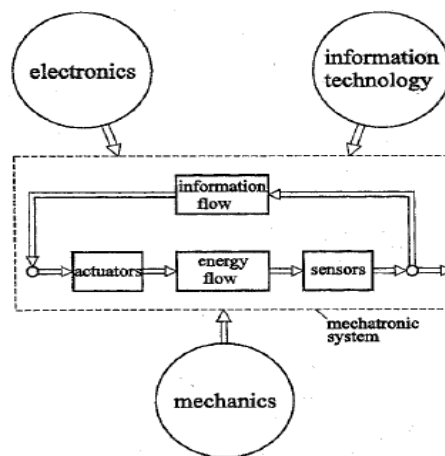


Figure 1. Integration of mechanic, electronics, and information technology leads a Mechatronic System.

The knowledge base contains quantitative and qualitative knowledge. The quantitative part operates with analytic (mathematical) process models, parameter and state estimation methods, analytic design methods (e.g., for control and fault detection), and quantitative optimization methods. Similar modules hold for the qualitative knowledge, e.g., in form of rules (fuzzy and soft computing). Further knowledge is the past history in the memory and the possibility to predict the behavior. Finally, tasks or schedules must be known. The inference mechanism draws conclusions either by quantitative reasoning (e.g., Boolean methods) or by qualitative reasoning (e.g., possibilistic methods) and takes decisions for the executive functions. Finally communication between the different modules, an information management data base and the man-machine interaction has to be organized. Based on these functions of an on-line expert system an intelligent system can be built up, with the ability “to model, reason, and learn the process and its automatic functions within a given frame and to govern it toward a certain goal”. Hence, intelligent mechatronic systems can be developed, ranging from “low-degree intelligent”, as intelligent actuators, to “fairly intelligent systems” as e.g., self-navigating automatic guided

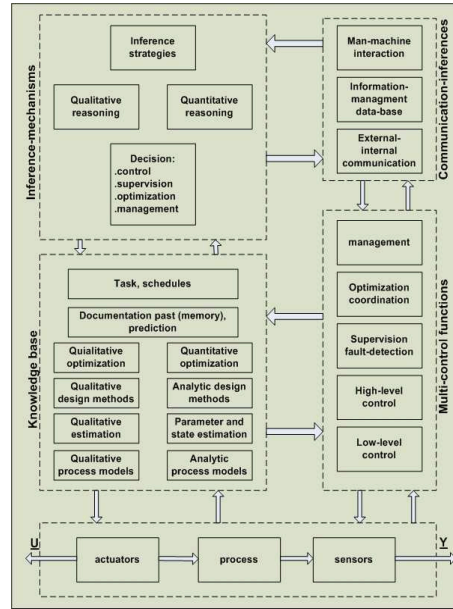


Figure 2. Advanced intelligent automatic system with multicontrol levels, knowledge base, inference mechanism, and interfaces.

vehicles. An intelligent mechatronic system e.g., adapts the controller to the mostly non-linear behavior (adaptation) and stores its controller parameters in dependence on the position and load (learning), supervises all relevant elements, and performs a fault diagnosis (supervision) to request for maintenance or if a failure occurs to fail safe (decisions on actions). In the case of multiple components, supervision may help to switch off the faulty component and to perform a reconfiguration of the controlled process, [14].

Rolf Isermann on his paper [14], illustrated a diagram (Figure 2) for an Advance intelligent automatic system with multicontrol levels, knowledge base, inference mechanism, and interfaces. Describing the different levels as:

- Level 1: low level control (feedforward, feedback for damping, stabilization, linearization);
- Level 2: high level control (advanced feedback control strategies);
- Level 3: supervision including fault diagnosis;
- Level 4: optimization, coordination (of processes);
- Level 5: general process management.

Isermann conclude "Process model-based methods allow the generation of analytical symptoms and even a fault diagnosis by reasoning methods" and "Mechatronic systems will become more and more intelligent, making use of quantitative and qualitative process knowledge bases and inference mechanisms in the higher automation levels".

4. Holonic Manufacturing System (HMS)

This new generation of manufacturing systems is referenced as intelligent manufacturing systems (IMS), whose constituent resources have to be capable of simultaneously addressing both knowledge processing about manufacturing process and equipment, and material processing requirements, [15].

Resistance the manufacturing world has not unreservedly embraced holonics. “Holonics systems require a bit of a paradigm shift of the way manufacturing systems were being done for the last 100, 150 years”, Gruver says. Most systems today have machines in one place and one smart computer somewhere else organizing everything. Holonic systems bring the intelligence closer to the machines and have the machines work out what needs to be done. “Companies have invested a lot of time and money in doing a centralized way of manufacturing they just do not want to let go”, he explains. Sabaz points out holonic is incompatibility with hierarchical systems: “One of the concerns that people have about holonic systems is that they literally attack middle management, because they take away the decision-making” Chand is aware of some reluctance toward holonics in manufacturing. “What we are looking for right now are high-leverage applications, small redundant cells where flexibility and fault tolerance are absolutely critical, like the Navy application” he says “In the water-routing example for the Navy, there is no operator involved it is a completely autonomous system”, [16].

Int the research report of ICT for Manufacturing were reported: theoreticians and practitioners need a new form of thinking and working in a truly multidisciplinary environment. To achieve this, new versions of production systems incorporating ICT in the products and all manufacturing facilities must be enhanced with new ontology. Collaborative production environments, characterized by production components with embedded intelligence, will be the result of the integration of agent technology, mechatronics and advanced control for doing network control systems. At this point, it is important to distinguish between devices (sensors/actuators) that may be wire-connected, and products that are necessarily wireless, [7].

Duncan McFarlane make some specific points about holonic manufacturing:

1. Holonic manufacturing is primarily a systems engineering methodology rather than a solution to a specific control problem.
2. Holonic manufacturing is referred to as a bottom-up approach because the overall plant control can be developed through the piecewise integration of flexible, interchangeable manufacturing modules called holons.
3. Each holon encapsulates the operational, sensing, decision-making and execution activities of a physical resource, order or product in the manufacturing environment, and behaviour emerges via the interaction of several of these holons.
4. In a manufacturing context orders are assigned and fulfilled via direct negotiation between orders and resources and in customising environments an order may be as small as a single product.
5. The holonic manufacturing approach is in direct contrast with conventional top-down methodologies for designing and specifying manufacturing control systems (e.g. computer-integrated manufacturing (CIM)) in which a computer control systems hierarchy is centrally devised to support the planning, scheduling and shop floor control processes.

6. The discriminating value of holonic manufacturing is that it represents the only methodology for control system design which manages short and long-term changes in the manufacturing environment as "business as usual".

Summary some simple descriptions of holons and holonic manufacturing systems:

- Stefan Bussmann: Holonic Manufacturing was first proposed as a new manufacturing paradigm in the beginning of the 1990s and has since then received a lot of attention in academic and industrial research. The holonic concept was developed by the philosopher Arthur Koestler in order to explain the evolution of biological and social systems. On the one hand, these systems develop stable intermediate forms during evolution that are self-reliant. On the other hand, it is difficult in living and organizational systems to distinguish between "wholes" and "parts": almost everything is part and whole at once. These observations led Koestler to propose the word "holon" which is a combination of the Greek word "holos" meaning whole and the Greek suffix "on" meaning particle or part as in proton or neutron, [17].
- Paulo Jorge Pinto Leitão: One paradigm for the factory of the future that translates to the manufacturing world the concepts developed by Koestler to living organisms and social organisations. Holonic manufacturing is characterised by hierarchies of autonomous and cooperative entities, called by holons, that integrates the entire range of manufacturing entities. A holon, as Koestler devised the term, is an identifiable part of a (manufacturing) system that has a unique identity, yet is made up of sub-ordinate parts and in turn is part of a larger whole. The essence of the holonic approach is the capability to decompose a complex problem into stable intermediate sub-problems, using hierarchy structures, [18].
- Duncan McFarlane define a holon as "an autonomous and co-operative building block of a manufacturing system for transforming, transporting, storing physical and information objects", [5].

A general architecture of a holon is depicted in Figure 3. The physical processing layer is the actual hardware performing the manufacturing operation, like for example milling or assembly. It is controlled by the physical control layer. The decision making represents the kernel of the holon and provides two interfaces: the first for interaction with other holons, and the second for interaction with humans, [17].

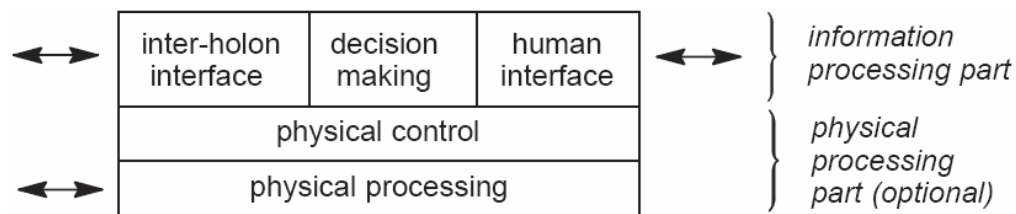


Figure 3. General architecture of a holon.

The integration of both the "holonic manufacturing system" (HMS) and the "agent-oriented manufacturing system" paradigms is currently presented as the basis for an IMS,

[15] and a "system of holons which can co-operate to achieve a goal or objective" is then called a holarchy, [5].

The novel elements of a holonic approach to execution are: a) execution proceeds via a negotiated set of steps rather than a pre-determined sequence and b) the resources (machines) executing the manufacturing operation are also responsible for the decisions made about the timing and nature of the execution, [5].

Establishing suitable implementation approaches with existing and future commercial computing systems, from an implementation perspective, there has been little or no work done in determining the compatibility of the holonic vision with the current or the next generation of industrial control and computing systems.

Determining how to construct and interface systems capable of fully supporting holonic operations with existing legacy systems will also be a major issue as holonic systems capabilities reach industrial strength. In the shorter term, suitable migration approaches for the implementation of intermediate holonic control capabilities are required, one such approach involving the combination of PC, Programmable Logic Controller and Machine (robot) controller to provide a holonic control infrastructure for an assembly cell Duncan McFarlane and Stefan Bussmann et. al 2000, [5].

Vladimír Marík on [4] tell, agent solutions for real-time control (such as those embedded in holonic control systems) work for only a limited scope of tasks for example, for very fast reconfiguration of the control equipment in a predetermined, "hard-wired" way. To add more intelligence to holonic decision making, you need to apply methods and techniques that are more knowledge intensive, [4].

So, researchers have proposed a general architecture where a single structure encapsulates a low-level real-time-control subsystem (a function-block or ladder-logic application) with a more deliberative intelligent agent.

4.1. Basic Holons

Basic holons for the HMS architecture can be of four types:

- **Product Holon (PH):** A Product Holon holds information about the process status of product components during manufacturing, time constraint variables, quality status, and decision knowledge relating to the order request. A Product Holon consists of a physical "component" and an information "component". The physical component of the Product Holon develops from its initial status (raw materials or unfinished product) to an intermediate product, and then to the finished one, i.e. the end product.
- **Product Model Holon (PMH):** A Product Model Holon holds up-to-date engineering information relating to the product life cycle (configuration, design, process plans, bills of materials, quality assurance procedures, etc.).
- **Resource Holon (RH):** A Resource Holon also contains physical and information components. The physical part contains a production resource of the manufacturing system (machine, conveyor, pallet, tool, raw material, and end product, or accessories for assembling etc.), together with controller components and planning scheduling components.
- **Mediator Holon (MH):** A Mediator Holon serves as an intelligent logical interconnection to link and manage orders, product data, and specific manufacturing resources dynamically. The Mediator Holon can collaborate with other holons to

search for and coordinate resources, product data, and related production tasks. A Mediator Holon can be itself a holarchy. A Mediator Holon can create a Dynamic Mediator Holon (DMH) for a new task such as a new order request or sub-order task request. The Dynamic Mediator Holon then has the responsibility for the assigned task. When the task is completed, the DMH is destroyed or terminates for reuse. DMHs identify order-related resource clusters (i.e. machine group) and manage task decomposition associated with their clusters, [19].

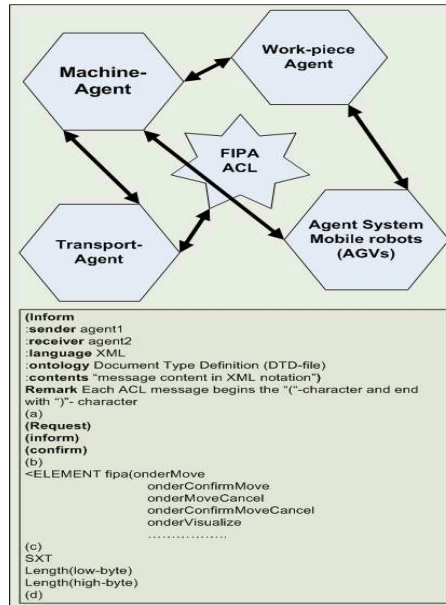


Figure 4. Implemented architecture of an intraenterprise holonic system. (a) Principle structure of the implemented FIPA ACL messages. (b) Types of communicative acts. (c) Characteristics of the developed ontology. (d) Length descriptor specifications.

4.2. Communication

Such communication is usually both application and company-specific and connected with the solution of the "migration problem" (adapting classical real-time control hardware to holonic control). Duncan McFarlane and his colleagues introduced a blackboard system for accomplishing intra-holon communication, while Martyn Fletcher and Robert Brennan proposed using a special management-service-interface function block. Greater interlinking with the operations management community would ensure equal attention to inter-holon communication, [4].

Of critical importance in holonic manufacturing is the control network which connects the physical machining operation to the software agent that coordinates it, because the algorithms used for developing real-time control actions rely on up-to-date information about the state of the machine. However, in the case of a product holon, it is far more difficult to achieve and maintain a connection between a physical product and the soft-

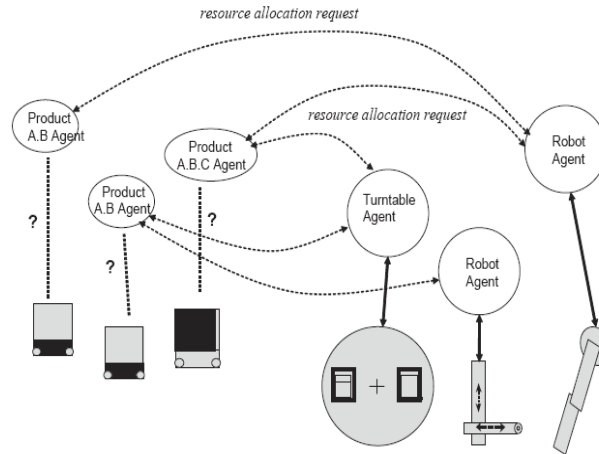


Figure 5. Product and resource holon interactions.

ware agent related to it because the product is continually moving and there is no permanent network connection available. We depict this situation in Figure 5 from [6] for the example of the manufacture of an electrical meter box. Here we see a number of meter box orders requiring appropriate resource allocations for assembly. Product and resource software agents (i.e. software sections of the corresponding holons) interact but because of limited real-time information about the movement and identity of products being made the flexibility of the system is severely limited. In order to address the decoupling of the physical parts of product holons from their corresponding software environments, [6].

The intelligent distributed control system platform, constituted by different types of agents running in PLC and CNC components, was developed and integrated with a Microsoft/DCOM and Ethernet-TCP/IP-based communication platform.

Encapsulation of the agent type-specific algorithm (COM DLL). The interface of each agent is basically message oriented. Based on the desired DCOM interface, the messages have been mapped to method calls. Therefore, the Factory Broker agent interface is described by the methods. The subset of interactions was described as a set of methods offered by various DCOM-based classes, which are providing a remote agent method interface (AMI) for these interactions for PC-based agents, [15].

Remark: As shown in [15] the Figure 4(a)-(d), in the socket layer, each FIPA ACL message is enhanced with a message length descriptor. As specification of the communication interface, a document type definition "DTD-file" is used to describe the grammar of the XML content included in each FIPA ACL message.

However, while multi-agent based control systems are purely software environments, a holonic manufacturing system encapsulates both the physical and software elements in the manufacturing environment. Hence a resource holon for materials handling, for example, might contain a robot, sensing and actuation, a local controller, a network connection and one or more coordinating software agents located either locally or on the network. In some sense, holonic control in the evolution of distributed manufacturing represents one further step beyond that of multi-agent control, in that the architecture as well as the underlying algorithms are increasingly product and resource oriented, [6].

We emphasise here that holonic manufacturing is neither an alternative nor identical to multi-agent control but rather it is complementary in that it represents a systems engineering approach to the development of manufacturing control systems infrastructure, rather than a solution mechanism for solving individual manufacturing control problems. Most developments of holonic systems to date have deployed agent-like solvers as a means of resolving planning, scheduling and shop floor control issues, [6].

5. Embodied BDI approaches

There are many researches striving to get embodied Believe-Desire-Intention(BDI), proposing different technics with the objective of get a novelty solution. Some investigation are leading and making inroad like [20,21,22], this work make emphasis specially on the next two, considering the newest and relevant for the objective.

Vasu S. in the paper [20], reference to exploited the essential BDI properties to embed it in software agents that model them. The paper introduces a formal software architectural design of a MAS in which the BDI architecture is embedded. They embed the BDI properties of agents in an extended state machine(ESM) modeling and suggesting that an implementation of the BDI architecture in a high-level programming language can be tested for conformance by generating test cases from the ESMs getting a formal software architecture which embeds the BDI properties of a MAS. On this research a behavior of the component (agent) $A_i \in A$ is represented as an ESM, which is an 8-tuple $(\varepsilon, \theta, \chi, \xi, \phi_{at}, \Lambda, \gamma, \rho)$ and propose an abstract interpretation of agent behavior depicted in Figure 6 named Agent Modes and Agent Behavior. Mao Zheng and colleges propose

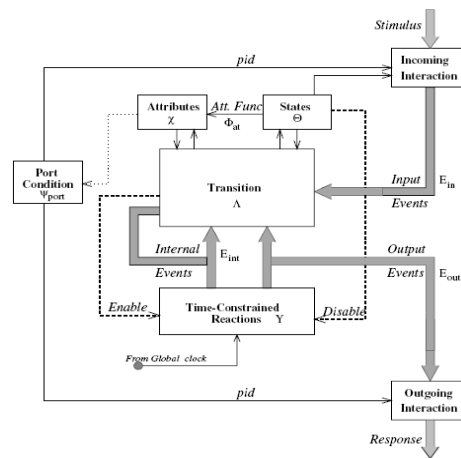


Figure 6. Agent Modes and Agent Behavior.

another paper [23] where develop a kind of proves for their research telling, it is necessary to choose an agent software abstraction that has a minimal deviation from the BDI agent concept, and use it as a formal basis to test an implementation. This paper propose one such method for conformance testing of BDI properties in an agent-based system, they model a software agent as faithfully as possible to a given BDI agent, and use the

model to generate test cases for a black-box testing of an implemented system, and have embedded the virtues of BDI concepts in ESMs, which are formal behavior models of components that encapsulate agents.

In Deepak Kumar paper [22], were working on a hybrid connectionist and BDI based. The architecture makes specific commitments in order to achieve a harmony among the tasks of reasoning and acting. The architecture uses a bottom-up learning strategy to acquire rules for reactive behavior. They architecture tries to accommodate the benefits derived from both types of models in the context of physical mobile robots. This research is explicitly concerned with implemented theories and systems then the objective of the work described is to contribute to the evolution of BDI architectures that can be used to model embedded rational agents capable of reasoning, acting, learning and interacting based on unifying underlying principles. Kumar job's its driven by some questions like "Is it possible to apply a BDI approach to model agents that are physically embedded in the world?".

6. JADEX

Jadex is an agent-oriented reasoning engine for writing rational agents with XML and the Java programming language. Thereby, Jadex represents a conservative approach towards agent-orientation for several reasons. One main aspect is that no new programming language is introduced. Instead, Jadex agents can be programmed in the state-of-the art object-oriented integrated development environments (IDEs) such as Eclipse and IntelliJ IDEA. The other important aspect concerns the middleware independence of Jadex. As Jadex is loosely coupled with its underlying middleware, Jadex can be used in very different scenarios on top of agent platforms as well as enterprise systems such as J2EE. Similar to the paradigm shift towards object-orientation agents represent a new conceptual level of abstraction extending well-known and accepted object-oriented practices. Agent-oriented programs add the explicit concept of autonomous actors to the the world of passive objects. In this respect agents represent active components with individual reasoning capabilities. This means that agents can exhibit reactive as well as pro-active behaviour.

Jadex is a Belief-Desire-Intention (BDI) reasoning engine for intelligent agents. The term reasoning engine means that it can be used together with different kinds of (agent) middleware providing basic agent services such as a communication infrastructure and management facilities. Currently, two mature adapters are available. The first adapter is available for the well-known open-source JADE multi-agent platform and the second one is the Jadex Standalone adapter which is a small but fast environment with a minimal memory footprint. In this tutorial the Jadex Standalone adapter is used, but in principle the used adapter is not of great importance as it does not change the way Jadex agents are programmed. The concepts of the BDI-model initially proposed by Bratman were adapted by Rao and Georgeff to a more formal model that is better suitable for multi-agent systems in the software architectural sense. Systems that are built on these foundations are called Procedural Reasoning Systems (PRS) with respect to their first representative.

7. First Approach of an Embodied BDI Architecture

Kumar job propose some relevant questions like "is it possible to apply a BDI approach to model agents that are physically embedded in the world". Using this like a hypothesis, we address the aim of our research to propose a different approach.

All the related issues presented summarize the importance of developing new methods and techniques that should be more knowledge intensive, applied at the level of embedded devices. To solve this requirement, the propose is to unify the mentioned topics into an Embedded Architecture for Micro-Controllers (μ C). A Micro-Controller (μ C) will be able to make decisions based on decentralized embodied BDI architecture, striving a solution to the problem presented specially on the need presented in the topic of more embedded intelligent design.

The propose of a new solution for the need of intensive knowledge embodied methods presented in several works of research, is oriented to join the related topics listed previously on this paper. To do that, we propose to focus on embedded intelligence, integrating research and technology from different areas such as:

- MAS: For distributed intelligent functions using ontologies.
- RT-MESSAGE (RTM): For analysis and design [24] and considering the successful case cited before, RTM could help for our propose specially in the sense of getting the different parts of the problem being tackled, that using the methodology of [24].
- JADE: For special task like getting the inputs and produce outputs.
- JADEX: The reasoning engine to be embedded.
- EMS: For specific particular tasks based on Finite State Machines (FSM).
- IMS: The base for the intelligent diagram proposing a new one, based on μ C-BDI Embedded Architecture.
- HMS: Using the capacity of decompose the problem combining with MAS-RTMESSAGE, used to define the types of holons PH-PMH-RH-MH, and considering the special characteristic mentioned on section 2.3.

7.1. Introduce for the first approach

The architecture should have different levels where: one level indicates the software engineering (SW) part, the principal part to be tackled in this level is to join research on SW methodologies such as IMS-RTM-HMS. The use of the methodology HMS help to identified the principal parts of the real time problem, once done that, the next step is to apply an analysis and design with RTM. One of the resulting of apply RTM is an ontology for the general system. Having done the combination of HMS-RTM the next is to join it with IMS, how is showed in the literature of IMS and RMT those envelop similar concepts, using this characteristic could be made a fusion of methodologies, this fusion ends the first part getting as result an analysis and design of the real time problem.

Another level could be interpreted as: Once having the design resulting of join RTM-HMS-IMS and using it, is possible to interpret the requirements such as a BDI-XML file and programming an intelligent procedural reasoning agent.

As third work is to develop a parser between BDI to High level code to be embedded, the code to be embedded is preferable to be Java, but it's must be research taking into account specification of the microcontroller to be used.

The last level of the architecture is the high level control and could be described in different levels as is depicted on Isermann diagram. The role of knowledge in Isermann could be described as: one section dedicated to receive -as inputs- and interpret information of the data acquisition system (hardware), and then the translated information must be send to inference mechanism and communication section. Inference section is based on BDI file download previously, inference section makes use of one knowledge data base to support deliberative process and then to be in communicating with communication and inference sections. Translate sections has a direct communication with output section dedicated to inform task to be ejec by the act system.

The dynamic of high level control has two different communication sections where ontology is needed, one of is to be in communication with act and scada systems and the other is to share relevant information with others nodes of the entire system.

Hereafter the research will be driven in different lines:

1. Constraint the investigation on the related topic
2. Made a fusion of an IMS and RTM
3. Create all communication protocols based on ontologies
4. Get a BDI design based on the previous fusion
5. Design the method to embody the BDI design in a Micro-controller.

8. Future Work

This research is focused on the development of the first draft of an architecture which provides reasoning like an answer to complicated task executables by an embodied system, including the intelligent part based on the procedural reasoning (BDI). The idea of μ C-BDI Embedded Architecture is to be applicable for several domains like intelligent mechatronic devices, intelligent machines, multi-agent distributed control, monitoring, supervisory control. The goal considered, hereafter, is to create an architecture for developing intelligent embedded systems using elements like RT-MESSAGE (RTM), JADE, JADEX, EMS, IMS, HMS into procedural embedded architecture. The next of this project is to formulate scenery of how the mention topics could be fusion to generate the desired architecture.

9. Conclusion

Holonic systems will require a high level of reasoning and computational capability at the shop floor levels, coupled with more flexible communications and more dynamic interfaces to human operators and users, [5].

The goal of modern assembly planning systems is to create activity sequences that are not only feasible but also optimized according to one or more parameters, such as makespan, machine or tool utilization, cost. The practice of optimization methods requires knowledge of the actual manufacturing resources that are available to execute the plan. Current practices for resource-based planning require manually feeding information about the available equipment, which is unsuitable within dynamic environments.

The increasing computational capacity of embedded systems is enabling the implementation increasing levels of intelligence in devices and machines. The possibility to

embed agent behavior in devices has led to the development of architectures based on holons and more recently on physical agents or actors.

Suitable technologies for knowledge processing and reasoning for embedded systems need to be further researched.

Suitable knowledge management and trust management within the domain of SWS needs to be further developed, [8].

Some researches in 2004 like Danna Voth [16] conclude, “Rockwell hopes to roll out an agent-embedded product in five years”.

The report makes mention to different topics such an idea to be used as backbone for the propose. And is presented too, a first approach of an architecture that embodied intelligent into a μ C, for the aim of solving control problems.

References

- [1] A.W. Colombo, F. Jammes, and H. Smit. Service-oriented architectures for collaborative automation. *IEEE*, 7803-9252, 2005.
- [2] Harm Smit, Jose L. Martinez Lastra, and Franvois Jammes. Orchestration of service-oriented manufacturing processes. *IEEE*, 1(05):7803–9402, 2005.
- [3] Paulo Leitão, Armando W. Colombo, and Francisco J. Restivo. Adacor: A collaborative production automation and control architecture. *IEEE INTELLIGENT SYSTEMS*, 1541-1672, 2005.
- [4] Vladimír Marík and Duncan McFarlane. Industrial adoption of agent-based technologies. *Published by the IEEE Computer Society*, 14, 27-35, 2005.
- [5] Duncan McFarlane and Stefan Bussmann. Developments in holonic production planning and control. *Int. Journal of Production Planning and Control*, 11(6):522–536, 2000.
- [6] Duncan McFarlane, Sanjay Sarma, Jin Lung Chirn, and Kevin Ashton C.Y. Wong. Auto id systems and intelligent manufacturing control. *Engineering Applications of Artificial Intelligence*, 2003, 16.
- [7] ICT for Manufacturing. The agile, wireless manufacturing plant. Technical report, February 2005.
- [8] Jose L. Martinez Lastra Member IEEE and Ivan M. Delamer Student Member IEEE. Semantic web services in factory automation: Fundamental insights and research roadmap. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, 2(1), FEBRUARY 2006.
- [9] Valeria Javalera Rincon. Aplicación de la metodología rt-message para el diseño de un sistema multiagente en tiempo real para el control de clima, riego y nutrición de cultivos en invernaderos, tesis de maestria, 2005.
- [10] Fernando Orduña Cabrera. Implementación de un sistema multiagentes para el control de clima en invernaderos, tesis de maestria, 2005.
- [11] Fernando Orduña Cabrera and Valeria Javalera Rincón. Sistema multiagente para el control de clima. *ITCH-ELECTRO*.
- [12] Valeria Javalera Rincón and Fernando Orduña Cabrera. Aplicación de la ingeniería de software para el diseño de un sistema multiagente en tiempo real. *ITCH-ELECTRO*.
- [13] Fernando Orduña Cabrera and Valeria Javalera Rincón. Agente para el control de clima en invernaderos. *AMCA*.
- [14] Rolf Isermann. Modeling and desing methodology for mechatronic system. *IEEE/ASME TRANSACTIONS ON MECHATRONIC*, 1(1):16–28, 1996.
- [15] Armando Walter Colombo Senior Member IEEE, Ronald Schoop Senior Member IEEE, and Ralf Neubert. An agent-based intelligent control platform for industrial holonic manufacturing systems. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*, 53(1):1541–1672, FEBRUARY 2006.

- [16] Danna Voth. Holonics in manufacturing: Bringing intelligence closer to the machine. *Published by the IEEE Computer Society*, 1541-1672, 4-7, 2004(1541).
- [17] Stefan Bussmann and Daimler-Benz AG. An agent-oriented architecture for holonic manufacturing control. *First Open Workshop IMS Europe Lausanne, Switzerland, organized by ESPRIT Working Group on IMS*, 1.
- [18] Paulo Jorge Pinto Leitão. *An Agile and Adaptive Holonic Architecture for Manufacturing Control*. PhD thesis, Department of Electrotechnical Engineering Polytechnic Institute of Bragança, 2004.
- [19] Xiaokun ZHANG and The University of Calgary 2500 University Drive Calgary Alberta T2N-1N4 Canada Douglas H. NORRIE, Division of Manufacturing Engineering. Holonic control at the production and controller levels. submitted to IMS'99, Leuven, Belgium, 1999.
- [20] Vasu S. Alagar and Mao Zheng. A software architecture for multi-agent systems. *CIS 2005, Part I, LNAI 3801, Springer-Verlag Berlin Heidelberg*.
- [21] Pierangelo Dell'Acqua, Mattias Engberg, and Luis Moniz Pereiray. An architecture for a rational reactive agent. *Department of Science and Technology - ITN, Linköping University, 601 74 Norrköping, Sweden, EPIA 2003, LNAI 2902*, pp. 379-393, 2003.
- [22] Deepak Kumar and Lisa Meeden. A hybrid connectionist and bdi architecture for modeling embedded rational agents. *Department of Math and Computer Science Bryn Mawr College, 1998 AAAI Fall Symposium, Technical Report FS-98-02*.
- [23] Mao Zheng and Vasu S. Alagar. Conformance testing of bdi properties in agent-based software systems. *Proceedings of the 12th Asia-Pacific Software Engineering Conference (APSEC'05)*.
- [24] Julián Inglada Vicente J. *RT-MESSAGE: Desarrollo de Sistemas Multiagente de Tiempo Real*. PhD thesis, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, 2002.