



HAL
open science

18th ICPR paper; Distributed Production Planning and Control Agent Based System

Rui Manuel Lima, Rui Manuel Sousa, Paulo Martins

► **To cite this version:**

Rui Manuel Lima, Rui Manuel Sousa, Paulo Martins. 18th ICPR paper; Distributed Production Planning and Control Agent Based System. *International Journal of Production Research*, 2006, 44 (18-19), pp.3693-3709. 10.1080/00207540600788992 . hal-00512920

HAL Id: hal-00512920

<https://hal.science/hal-00512920>

Submitted on 1 Sep 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



18th ICPR paper; Distributed Production Planning and Control Agent Based System

Journal:	<i>International Journal of Production Research</i>
Manuscript ID:	TPRS-2005-IJPR-0527.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	06-Apr-2006
Complete List of Authors:	Lima, Rui; School of Engineering of University of Minho, Department of Production and Systems Sousa, Rui; School of Engineering of University of Minho, Department of Production and Systems Martins, Paulo; School of Engineering of University of Minho, Department of Production and Systems
Keywords:	MULTI-AGENT SYSTEMS, DISTRIBUTED MANUFACTURING CONTROL
Keywords (user):	DISTRIBUTED PRODUCTION PLANNING AND CONTROL



1
2
3
4 Rui M. Lima¹
5
6

7 Department of Production and Systems
8
9 School of Engineering of University of Minho
10
11 Campus de Azurém, Guimarães, Portugal
12
13 Tel: +351253510359; Fax: +351253510343
14
15 E-mail: rml@dps.uminho.pt
16

17
18 Rui M. Sousa
19

20 Department of Production and Systems
21
22 School of Engineering of University of Minho
23
24 Campus de Azurém, Guimarães, Portugal
25
26 Tel: +351253510357; Fax: +351253510343
27
28 E-mail: rms@dps.uminho.pt
29

30
31 Paulo J. Martins
32

33 Department of Production and Systems
34
35 School of Engineering of University of Minho
36
37 Campus de Azurém, Guimarães, Portugal
38
39 Tel: +351253510358; Fax: +351253510343
40
41 E-mail: pmartins@dps.uminho.pt
42
43
44
45

46 47 **Distributed Production Planning and Control Agent** 48 **Based System** 49 50

51
52
53
54
55
56 ¹ Corresponding author.
57
58
59
60

Abstract

A model of an Agent based Production Planning and Control (PPC) system able to be dynamically adaptable to local and distributed utilization of production resources and materials is presented. The PPC system is based on the selection of resources to deal with one order of different quantities of one product each time. In this way it is build one scheduling solution for that particular order. The production resources are selected and scheduled using a multi-agent system supported by an implementation of the Smith Contract Net, using Java Spaces technology. The multi-agent system is based on three main agents: Client, Resource and Manager. These agents negotiate the final product, and the correspondent components, requested by the client. An order for each product (component) triggers a process of dynamic design of a production system to fulfill that particular order. This system exists till the end of the order.

Keywords: Distributed Production Planning and Control, Multi-Agent System

1 Introduction

New paradigms are necessary for enterprise representation and operation, and consequently also for production systems, in order to deal with the progressive reduction on time to market of new customized products and the ever-growing need for new enterprise competition approaches. These requirements make a further stress to the production Planning and Control System, which must be dynamically adaptable to both local and distributed utilization of production resources and materials.

Slightly different approaches to networks of enterprises like Virtual or Extended Enterprises are being referred in the literature. Virtual Enterprises are ephemeral associations of enterprises to give answer to a transitory opportunity, usually highly technological dependent, and Extended Enterprises results from a more steady association of enterprises throughout the manufacturing chain, usually centred on a dominant one.

Some paradigms for building the relations between production resources have been proposed, like Holonic Manufacturing, Fractal Factory and Bionic Manufacturing Systems.

Holonic Manufacturing Systems, according to Tharumarajah, Wells and Nemes (1996), are based on production units, named Holons, inside other units, named holarchy, making sub-systems that will build the final production system. Each one of these Holons have simultaneously the all and the one characteristics. They have characteristics of the one because they have part of the functionality of the system, and on the other perspective they can be viewed as a system (the all) because they have some kind of autonomy (like the all system) and contain other Holons inside them.

1
2
3
4 The Fractal Factory concept was introduced by Warnecke (1993) based on fractal
5 geometry. This factory model is composed of similar units (fractal) that provide production
6 services. The definition of the fractal by Warnecke (1993) is: *a fractal is an independently*
7 *acting corporate entity whose goals and performance can be precisely described.* The
8 fundamental characteristics of the fractal are: self-similar, self-organization, goal
9 orientation and dynamics.
10

11
12 Bionic Manufacturing System is a production system concept conceived by Okino, and
13 inspired on the biological living being systems (Tharumarajah, Wells and Nemes, 1996).
14 Production units of productions systems, viewed as a biological system cell, get and send
15 production objects (materials, products and information) to surround environment.
16 Communication will be established by information and material flow, regulated by
17 coordinating units, which will allow the integration of activities of the autonomous
18 production units. These coordinating units will also have the function of linking cells in
19 different levels of the hierarchy.
20

21
22 Research work addressing the referred concepts is being developed in order to make
23 contributions to several fields related to PPC, namely on organization of distributed
24 production units and selection of production resources or suppliers in supply chain
25 networks.
26

27
28 Carvalho, Moreira and Pires (2005) propose an organization concept, the Autonomous
29 Production System (APS), as the base unit to build autonomous and reconfigurable
30 production systems. The APS unit has, simultaneously, the characteristic of the whole and
31 of the part like Holons and the holarchy in Holonic Manufacturing System. The PPC model
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1
2
3
4 proposed is related with the hierarchical PPC concept (McKay and Wiers (2003);
5
6 Venkateswaran and Son (2005)).
7

8
9 Selection of suppliers in order to satisfy demand in supply chain is one of the problems
10
11 addressed in the literature. Chan, Felix T. S., Chung and Wadhwa (2004) propose a Central
12
13 Coordination System (CCS) to make optimal decisions of demand allocation in a supply
14
15 chain network. The centralized allocation decision is based on the Analytic Hierarchy
16
17 Process (AHP) methodology, and considers qualitative and quantitative criteria. Chan,
18
19 F.T.S. and Chan (2004) propose a multiagent system (MAS) to simulate the distributed
20
21 coordination of a supply chain. This MAS is based on eleven agents per node of the supply
22
23 chain that could act as a company representative. This model is used for a simulation study
24
25 subject to demand and supply uncertainty presented in Chan, F.T.S. and Chan (2006).
26
27
28 Allwood and Lee (2005) present an agent model that could be used to study the supply
29
30 chain network dynamics. The agent model has a strategic and an operational level. The first
31
32 is responsible to determine prices and target inventory levels, rank customers and evaluate
33
34 suppliers. The second is responsible for demand management, production planning and
35
36 control, materials management and accounting.
37
38
39

40
41
42 Ulieru and Cobzaru (2005) propose a Holonic supply chain management system for the
43
44 Telephone Manufacturing Industry. The multi-agent system is based on the following
45
46 agents: Logistics, Order Manager, Customer, Transport, Bank and several Plant agents
47
48 representing the assembly plant and various suppliers.
49

50
51
52 Reconfiguration is explicitly addressed by Bruccoleri, Renna and Perrone (2005) in a
53
54 simulation system representing reconfigurable machine systems (RMS). The simulation
55
56 system is used to test a multiagent model. This model has four negotiation protocols based
57
58
59
60

1
2
3
4 on the Contract Net protocol, presented by Smith (1980), to handle dynamic resource
5 allocation and different types of exceptions, namely: machine breakdowns, machine
6 deterioration and rush orders.
7
8
9

10
11 The three paradigms referred at the beginning of this section share the following
12 characteristics: self similar parts; distribution of production resources; some degree of
13 autonomy of production resources; system reconfigurability. These characteristics are also
14 used as foundation for research work reported on the literature. In the following section a
15 definition of distributed production system is presented, which leads, subsequently, to a
16 proposed production planning and control agent based model.
17
18
19
20
21
22
23
24

25 **2 Distributed production planning and control**

26 **2.1 Distributed production system definition**

27
28
29
30
31
32 In general, production systems are made of different processing elements, working in
33 heterogeneous environments, dealing simultaneously with different processes of different
34 products and communicating in several ways. The term *distributed*, associated to
35 production systems, emerges from the identification of new organising and management
36 needs. These organising and management concepts must solve problems associated with the
37 growing need of adaptability to change. This lead to the definition presented by Lima, Silva
38 and Martins (1999):
39
40
41
42
43
44
45
46
47
48

49 *A Distributed Production System is a production system composed by a network of*
50 *autonomous processing elements, with the capability of rapid dynamic reconfiguration.*
51

52
53 This definition excludes traditional production systems that do not allow instant dynamic
54 reconfiguration. Moreover, new ways of management are necessary to allow this
55
56
57
58
59
60

1
2
3
4 reconfiguration, integrating, in the same model, all parts of the system, including
5
6 processing elements, inputs, outputs and communication systems.
7
8

9 10 **2.2 Distributed production planning and control model**

11
12
13 Production planning and control systems are composed by sub-systems responsible for
14
15 managing several aspects of production activities. Materials managing, capacity planning,
16
17 scheduling production resources and operations monitoring are some of those aspects. A
18
19 production planning and control model should broadly present its components and their
20
21 functionalities in order to allow the definition of:
22
23

- 24
25 • The availability of final product quantities, by period, over a planning horizon.
- 26
27 • Material quantities requirements to deliver those final products.
- 28
29 • Capacity requirements to process production operations.
- 30
31 • Production resources schedule over the planning horizon, in order to satisfy material
32
33 requirements.
34
35

36
37 In this work, the production planning and control model should be in accordance to two
38
39 basic requirements: it must be capable to manage distributed production systems as defined
40
41 previously, and also do it for one order of different quantities of one product each time.
42
43

44
45 Satisfaction of these requirements can be done by selecting distributed resources for one
46
47 order at a time. These resources should be scheduled to process a network of production
48
49 operations that can transform materials in components and then assemble them into final
50
51 products. In this way, each order can be related, depending on some defined criteria, to a
52
53 different set of production resources. This configuration and reconfiguration of production
54
55 systems is related with the definition of DPS presented. Furthermore, each production
56
57
58
59
60

1
2
3
4 resource is selected to completely process a part element, i.e. item, of the product structure.
5
6 So, in this sense, this production resource should have the autonomy to process the
7
8 necessary operations to change the state of the input materials into some part element
9
10 output.
11

12
13 The process of selection and scheduling of distributed resources of a production planning
14
15 and control model could be based on two different alternatives, here named local and global
16
17 processes. The production resources selection, on both the local and global processes,
18
19 should be based on some defined criteria. In the developed model, minimum throughput
20
21 time for each order was used as the main criteria, with the objective of reducing delivery
22
23 time and work in process.
24
25
26

27
28 In the global process, production resources should be selected as a result of one overall
29
30 selection process, based on information about all the resources that can process the change
31
32 of state transformations of all part elements of the product structure. The application of the
33
34 minimum throughput time criteria for resources selection should be made by some entity in
35
36 a centralized way, based on resources ability, free time agenda and operations time for each
37
38 item of the product structure. This way, resources would have some lack of autonomy
39
40 because some external entity will define their delivery time for each item.
41
42
43

44
45 A local process allows, for each item of the product structure, the selection of production
46
47 resources capable to deliver that item. These production resources can release orders for
48
49 their input materials and a new process of resource selection is triggered. The repetition of
50
51 the local process for the orders of all items of the product structure allows the dynamic
52
53 configuration of the distributed production system. Figure 1 illustrates the local process in
54
55 which an order manager can select some resources to deliver an ordered quantity of an
56
57
58
59
60

1
2
3
4 item, based on its own criteria and decision factors. These decision factors could be
5
6 resources ability, capacity, delivery time, quality indicators or costs.
7
8

9
10
11 Figure 1: Local process for production resources selection.
12

13
14
15
16 The agent based system presented in the following sections is based on the application of
17
18 local process principles, with throughput time as main criteria for decision making. This
19
20 allows resources to maintain a high degree of autonomy because they can decide when they
21
22 would answer to an order request and define their own delivery time.
23
24
25

26 27 28 **3 Agent systems** 29

30 31 32 **3.1 Software agents** 33

34
35 Software agents are components of software that represent user intentions. Table 1 presents
36
37 definitions of three different authors.
38
39

40
41
42 Table 1: Software agent definitions
43
44

45
46
47 These definitions have some common characteristics like agent autonomy and project-
48
49 defined objectives. In this context, autonomy is the software component ability to keep on
50
51 executing their processes independently of interacting software or users. If the agent is not
52
53 subject to direct interferences it can refuse task execution requests. Technically, agents
54
55 respond to task requests in the opposite of objects that react by task invocation.
56
57
58
59
60

1
2
3
4 The agent autonomy must be integrated with some kind of environment monitoring ability
5 in such a way that the agent actuation is compliant to the project objectives. These
6 objectives are directly connected with the user represented by the agent. The action and
7 reaction executed by agents depends on objectives introduced during the project.
8 Depending on requests or environment changes, the agent must evaluate its internal state
9 and objectives to be able to deliver an answer.
10
11
12
13
14
15
16
17
18
19

20 **3.2 Multi-agent systems**

21
22 Multi-agent systems are characterized by communities of agents whose interaction allows
23 the achievement of system objectives. Nwana and Ndumu (1999) argues that multi-agent
24 systems are created with the intention to establish connections between agents developed
25 separately, allowing the overall capacity to go beyond the sum of the individual agents
26 capacities.
27
28
29
30
31
32
33

34 In multi-agent systems a common notion of the involved concepts must exist, in order to
35 reach their objectives. These concepts can be explicitly defined in the ontology of the
36 system, or can be implicitly defined in the knowledge database of each agent. Despite
37 having a common knowledge of concepts of the system, agents can also have knowledge on
38 the behaviour and reaction of the system due to other agents and environment changes. It
39 can be said that software agents can have models of other agents and systems in which they
40 are integrated, based on their project functionalities and behaviour.
41
42
43
44
45
46
47
48
49
50

51 Communication between agents depends on system architecture and can be done by
52 message exchange. This exchange of messages can rely on known standard languages or on
53 a specifically defined message protocol. System (environment) monitoring depends on its
54
55
56
57
58
59
60

1
2
3
4 objectives, architecture and agents, and can be done by interception of requests that
5 circulate in the system, by sensors reading, or by direct inquiries to the agent user or to
6
7 other agents.
8
9

10
11 According to Shen (2001) and Shen, Norrie and Barthès (2001), the connection structure of
12 agents can be based on the following three architectures:
13
14

- 15 • Hierarchy of agents.
 - 16 • Federation of agents.
 - 17 • Autonomous Agents.
- 18
19
20
21
22

23 The hierarchy is characterized by control relations from some agents over others. In that
24 case, the autonomy of the agents is restricted by some degree of control imposed by
25 dominant agents.
26
27

28 In a federation, individual agents or groups of agents communicate through mediator agents
29 who supply communication services. These agents of communication can, basically, be of
30 three types: facilitator; broker; match maker.
31
32

33 The facilitator is connected to a set of agents and communicates with other facilitators to
34 supply communication services that allow system operation (examples in Shen, Maturana
35 and Norrie (2000), Sun, Zhang and Nee (2001) and Ulieru and Cobzaru (2005)).
36
37

38 The broker is an agent who actuates in one specified market, between supplier agents and
39 client agents, supplying communication services that makes transparent the connection
40 between them (examples in Parunak *et al.* (1999), Baker, Parunak and Erol (1999), Kim,
41 Choi and Yoo (2001) and Carvalho, Putnik and Cunha (2003)).
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1
2
3
4 The match maker supplies services similar to those supplied by the broker, being able, after
5 that, to abandon the system, because the agents start to communicate directly between
6 themselves (examples in Parunak *et al.* (1999) and Shen, Maturana and Norrie (2000)).
7
8

9
10
11 The architecture based on autonomous agents is an architecture where the agents can
12 communicate between themselves without appealing to mediator agents. This type of
13 architecture leads to systems where agents know from each other, or to systems where a
14 common platform exists to transmit information available to all agents (examples in Shen,
15 Norrie and Barthès (2001), Wiendahl and Ahrens (1997), Parunak, Baker and Clark (1997),
16 Leitão, Restivo and Putnik (2001) and Chan, F.T.S. and Chan (2004)).
17
18
19
20
21
22
23
24
25
26
27

28 **4 Agent production system model**

29
30 In this model, software agents represent every production system element. These elements
31 will be able to communicate with all other elements in an autonomous and interactive way.
32
33
34
35

36 **4.1 System requirements**

37
38
39 The project of the proposed agent production system model is based on the following
40 requirements:
41
42
43

- 44 1. Product information input.
 - 45 i. Product specification.
 - 46 ii. Product structure definition.
 - 47 iii. Production processes specification.
- 48
49
50
51
52
53
54
55
56
57
58
59
60

- 1
- 2
- 3
- 4
- 5 2. Client information input.
- 6
- 7 i. Identification.
- 8
- 9 ii. Product orders generation.
- 10
- 11 3. Product orders input.
- 12
- 13 i. Product Definition.
- 14
- 15 ii. Quantity definition.
- 16
- 17
- 18 4. Order management agents should be able to:
- 19
- 20 i. Select production resources.
- 21
- 22 ii. Handle resource failure.
- 23
- 24 iii. Do order monitoring.
- 25
- 26
- 27
- 28 5. Resource agents should be able to:
- 29
- 30 i. Represent a production resource.
- 31
- 32 ii. Register types of production processes that can be executed by the resource.
- 33
- 34 iii. Publish lead time and costs for order requests.
- 35
- 36 iv. Answer to agenda request.
- 37
- 38 v. Do task scheduling.
- 39
- 40 vi. Register confirmed orders.
- 41
- 42 vii. Do monitoring tasks.
- 43
- 44 viii. Process orders.
- 45
- 46
- 47
- 48
- 49 ix. Act in conformity with defined objectives.
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

- 1
- 2
- 3
- 4
- 5 6. Evaluation criteria
- 6
- 7 i. Order request answer time.
- 8
- 9 ii. Lead time.
- 10
- 11 iii. Negotiated due date fulfilment.
- 12
- 13 iv. Cost.
- 14
- 15
- 16 7. Some of these requirements demand:
- 17
- 18 i. Agent communication.
- 19
- 20 ii. Actuating on environment.
- 21
- 22 iii. Reacting to environment changes.
- 23
- 24
- 25
- 26

27 **4.2 System model**

28
29
30 The project of a distributed production planning and control system presented in this work,
31 that fulfils the system definition and, partially, the requirements, is based on a multi-agent
32 system. A multi-agent system is adequate for modelling and implementation purposes due
33 to the following reasons:
34
35
36

- 37
- 38
- 39
- 40 ▪ Distribution – Multi-agent systems are adequate for the implementation of distributed
41 and complex systems with resource allocation tasks.
- 42
- 43
- 44 ▪ Autonomy – This is simultaneously an attribute of the distributed production system
45 and of the agent definition, so agents can be used to represent autonomous production
46 processing elements.
- 47
- 48
- 49
- 50
- 51 ▪ Reconfigurability – The reconfigurability of distributed production systems can be
52 implemented by proper coordination mechanisms of the multi-agent system.
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

1
2
3
4 In this model, all elements of the distributed production system are represented by agents,
5 which communicate with, and in name of, each element. The production resources, i.e.
6 production system processors, delegate their representation on agents. The clients, direct or
7 indirect users of resources, are also represented by agents. An order management agent is
8 responsible for coordinating the resource allocation task based on the Smith Contract Net
9 protocol.
10

11 According to Rich and Knight (1991), in this type of coordination an agent decomposes the
12 problem and negotiates the attribution of subtasks with other agents. In this contract net,
13 agents may have two roles: Manager, who decomposes the problem, looks for contractors
14 to execute parts of its problem, and supervises the execution; Contractor, who executes
15 subtasks or starts looking for contractors to execute part of the work, becoming thus a
16 manager.
17

18 In this model it is assumed that resource agents know the resource processing ability and
19 capacity for executing tasks of known products. So, based on the knowledge about the
20 transformations that the represented resource can execute, the agent can try to obtain orders
21 for that resource.
22

23
24 **4.2.1 Implementation.** The system is implemented with agents distributed by different
25 places, communicating through a shared repository. This form of communication is similar
26 to a black board. All system agents access to this repository, monitoring thus the system
27 activity.
28

29 JavaSpaces Technology (<http://java.sun.com/javaspaces>) included in Jini Network
30 Technology was used for implementation purpose. This technology is described by Halter
31
32

1
2
3
4 (2002) and Bishop and Warren (2003) as a system that delivers a set of services to manage
5 distributed software objects. The objects can be placed on a 'space', named JavaSpace,
6 where they can stay in a persistent way. The JavaSpace works like a Jini service, which
7 allows the space clients to store and share software objects.
8
9

10
11
12
13
14 The software agents run on own threads, possibly in different machines, and send messages
15 to other agents through a JavaSpace (figure 2). These messages are in conformity with a
16 message protocol, which is defined based on a common information model of the
17 Distributed Production System. In some cases, messages don't have to be sent to a
18 particular client, and can be addressed to a given type of agents. For example a request for
19 bids on some task for some order can be placed in the JavaSpace and several resources can
20 answer to that bid. Figure 2 represents two resource agents, two client agents, an order
21 management (OrderMgm) agent, three different messages flowing between several agents
22 and the JavaSpace.
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Figure 2: General structure of the agent model.

4.3 Agents specification

Agents Client represents each client and can, after registration in the system, send an order request for a particular product, through messages. This message is read by the order management (OrderMgm) agent, which will divide this in suborders for each element of the product structure. Each of these suborders will be recursively published in the JavaSpace. All interested Resource agents can make a bid for each of those suborders. Figure 3

1
2
3
4 illustrates the process of suborder message publication and the bid messages answers from
5
6 interested resources.
7
8
9

10
11
12 Figure 3: Elementary order task negotiation.
13

14
15
16 The Agent OrderMgm negotiates (requests) the final product requested by the client and
17
18 respective components with Agents Resource. Each component has a detailed structure,
19
20 associated to the production type of processes in all structure levels. The type of process is
21
22 the transformation needed to get the component. The Agent OrderMgm is responsible for
23
24 the selection of the resources needed for the production (candidate selection followed by
25
26 particular resource(s) selection). An order for each product (component) triggers a process
27
28 of dynamic design of a production system to fulfil that particular order. This system exists
29
30 till the end of the order and is related with a particular set of resources selected. These
31
32 resources are allocated to execute all tasks needed to make the production of all elements of
33
34 the product structure. Figure 4 illustrates a configuration for a product P_1 , which structure
35
36 refers two components and one of them also has two components.
37
38
39
40
41
42
43
44

45 Figure 4: Production system example.
46
47
48

49 Each Agent Client is able to store information about the required products; the production
50
51 means to execute them, the required processes and the orders already done. Each order a
52
53 Client Agent makes is related to a final product that can be produced in different resources
54
55 related to different types of processes.
56
57
58
59
60

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

4.3.1 Agent Client. Each client has an Agent Client that represents him and makes possible the orders input into the system. The introduction of an order, sending a message to the JavaSpace, will be the initiating event for the formation of each production system. Thus being, the Agent Client creates orders of a product in some required quantity.

Each Agent Client must be capable to identify the client it represents, to store information about products, orders, candidate and selected resources.

The ordered product results from the execution of some transformations that can be executed by some resources, if these have the required abilities. The Agent OrderMgm is responsible for the selection of offers, that is, for the attribution of some amount of work for candidate resources. This agent is also responsible for monitoring the execution of this order. An Agent Client creates an order for an intended product and sends a message for an Agent OrderMgm for the management of this order.

An order success should be reflected on the information management carried by the client. The client should, at least, store information for future reference about lead time, quality, cost and resources involved.

An order failure generates an internal conflict on the client, which the Agent should resolve, abdicating on the order or generating a new order. The first one of these solutions has implications on the user of this agent, who should be responsible for this decision, by inquiry or delegation. The decision to generate a new order leads again to the activity execution of launching an order.

There are more business activities related to clients that are not object of analysis in this study, because the objectives defined for this work were mainly related with production planning and control activities for distributed production systems.

1
2
3
4 **4.3.2 Agent OrderMgm.** The Agent OrderMgm will negotiate / request the product
5 intended by the customer. The Agent OrderMgm will search for resources able to execute
6 the necessary processes of production. This search process could be done by Agents with
7 the ability to search information on the net, that is, to search information on operations
8 and/or processes published by different resources. In a federate architecture, broker agents
9 would have the responsibility to find the appropriate resources. In this work it is used a
10 shared repository of information, where interest in the attainment of one determined
11 product is published, and in which can be collected information on production means that
12 can execute the required order.
13
14
15
16
17
18
19
20
21
22
23
24

25
26 Resource selection depends on the offers made by all resources for a particular order of a
27 product. The interested resources have to publish their offers for execution of the work,
28 indicating the execution time.
29
30
31

32
33 An activity of resource selection is composed, in general, by several sub-activities, that can
34 be resumed by: resource selection, monitoring the order execution and communication with
35 the customer. In this system, the production resources selection is made by the Agent
36 OrderMgm, executing the actions presented in the UML (OMG (2005)) activity diagram
37 represented in figure 5. This activity is initiated by publishing the order requirements,
38 followed by offers gathering. If exist offers from resources, then the control will be
39 transferred to the 'resource selection' action, else it will have to communicate that fact to
40 the client.
41
42
43
44
45
46
47
48
49
50
51
52
53

54 Figure 5: 'Resource Selection' activity diagram.
55
56
57
58
59
60

1
2
3
4 If the resources selection ('Select Resources' action state) is successful, then the Agent
5 OrderMgm executes the activity of effective solicitation of resources ('Request Resources'
6 action state). The impossibility to make the selection of resources, due to the incapacity of
7 the resources or to the application of the selection strategy, will be communicated to the
8 order manager agent. This Agent OrderMgm will inform the Agent Client that should act in
9 compliance with its objectives.
10
11
12
13
14
15
16
17
18

19
20 **4.3.3 Agent Resource.** The Agents Resource represents the available production
21 resources, with the objective of obtaining work to be executed by the resources that they
22 represent. When the Agent OrderMgm places information about an order in the shared
23 JavaSpace, the Agent Resource can answer placing offers for this order in the space. Each
24 of these offers depends on the request analysis and a decision is made, based on the agenda,
25 capacity and abilities of the resource that is represented by the agent. This offer should have
26 all information necessary for proper evaluation. The necessary information depends on the
27 strategy used in the system for selection of resources. One of the functions of the agent is to
28 keep its agenda updated, considering the selected orders and synchronized with the 'real'
29 resource.
30
31
32
33
34
35
36
37
38
39
40
41
42

43 The Agent Resource stores information about the resource it represents, the production
44 processes that can execute, the orders for which it made offers and on those where were
45 selected.
46
47
48

49 Figure 6 represents the activity diagram illustrating the activity of creation of offers for an
50 order by the Agent Resource. This activity is initiated with the reception of a notice about
51 publication of an order. This is followed by the execution of two parallel actions: one to
52
53
54
55
56
57
58
59
60

1
2
3
4 update the resource agenda, in accordance with its internal information and the information
5 of the resource it represents; other to search the order information in the shared JavaSpace.
6
7
8
9 If any one of these actions fails, the activity finishes without creation or publication of
10 offers.
11

12
13
14 The success of both of these actions allows to analyze the possibility of order fulfilment by
15 the resource, that is, to analyze its aptitude, the availability of material and capacity and the
16 fulfilment of the defined objectives. If the analysis is negative, then it refuses the order and
17 finishes the activity without publishing any offer. The internal acceptance of the order leads
18 to the creation and publication of an offer. The offer must include the lead time and
19 associated cost.
20
21
22
23
24
25
26
27
28
29

30 Figure 6: Agent Resource 'Offer' activity diagram.
31
32
33
34

35 If offers published by the Agent Resource are selected by an Agent OrderMgm, then there
36 will be a request of a particular amount of work to be processed by the resource. In this
37 case, the Agent Resource 'work' activity (figure 7) is similar to the 'offer' activity. With
38 this activity the Agent Resource has the objective of verifying the possibility of acceptance
39 of the work request made by the Agent OrderMgm.
40
41
42
43
44
45
46
47
48

49 Figure 7: 'Work Acceptance' activity diagram.
50
51
52
53

54 This 'Work Acceptance' activity initiates with the reception of a work request, followed by
55 the execution, in parallel, of two actions: one action to search the work request in the
56
57
58
59
60

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

JavaSpace; other action to update the resource agenda. Having information about the work request and the agenda updated makes possible to take a decision about the acceptance of the request. This decision is based on two parallel actions: one to analyse the work request in relation with a previously made offer; other to verify the updated agenda in relation with a previously made offer. The success of these two activities allows the agent to publish the acceptance. This acceptance implies the commitment of the resource with the execution of the work.

4.3.4 Coordination of agents. The definition of the distributed production system, the agent production system requirements and model place several orienting boundaries for the coordination of the agents.

The production system results from the selection of resources that are able to execute product transformations for the order. Making the selection of resources for each product transformation and grouping the *partial* solutions for each part of the product structure, makes possible to complete the order. This selection results, like referred on section 4.2, from a recursive application of a Contract Net negotiation protocol supported by message exchange through a JavaSpace. The implementation of this coordination mechanism is based on the exchange of object messages represented in figure 8. This simple object allows the implementation of all communication protocol. Attributes ‘origin’ and ‘destination’ can identify specific agents. Attributes ‘type’ and ‘param’ are related with the message content. The first can be, for instance, an order or an offer and the ‘param’ attribute is a vector data type containing information about the offer or order. In this object, only the first and the third of these attributes are mandatory.

Figure 8 : Message object

In the resulting production system, the order can be executed by one Agent Resource or several agents. This depends on the existence of the required ability and capacity, and the application of the selection strategy. In the developed system the selection strategy depends only on the lead time offer of each resource for each order.

A model for processing orders, as presented by Scheer (1999), can be extremely detailed, with specification of functions, objects flow, control flow, materials flow, use and responsibility of functions, objectives, and organizational elements. In this model it is specified the order creation and publication activities and the resources selection, being encapsulated, in the Agent Resource, the order processing.

5 Conclusion

A definition for distributed production systems and a related planning and control system model are presented in this work. This model is based on three main characteristics: distribution, autonomy and reconfigurability. These are fundamental characteristics for new paradigms of production systems that propose to deal with highly dynamic environments.

The distributed production planning and control agent based system is conceptually modelled and implemented as a multi agent system based on JavaSpaces technology. This implementation made possible the verification of the model validity in respect with production planning and control concepts. This system is able to make a production

1
2
3
4 resource selection to fulfil a particular order considering the resources agenda. Moreover, it
5
6 does this with distributed resources, which can accept or refuse orders depending on their
7
8 own strategy. So, in this way, the production resources autonomy is respected and it is
9
10 possible to deal with their distribution. Reconfigurability exists for each order, that is, there
11
12 is one configuration of resources for each order. If one resource fails then it is also possible
13
14 to build a new fraction of the system to fulfil the required part of the product structure.
15
16

17
18 There are three basic agents on the system: Agent Client, Agent OrderMgm and Agent
19
20 Resource. These three agents have a common knowledge about the system and the
21
22 environment, allowing communicating with each other. The presented UML specification
23
24 of these agents is based on the description of the main activities they execute and that
25
26 support their interaction.
27
28

29
30 In this work it is also presented the description of the communication protocol for sending
31
32 message objects to a shared space and the coordination mechanism based on the application
33
34 of the Smith Contract Net protocol.
35
36

37
38 Future work is planned in order to extend this system in functionality and also in different
39
40 directions. The model functionality should be incremented in order to apply and compare
41
42 different criteria for selection of resources, like cost or resource utilization. Moreover, it
43
44 should allow implementing different architectures: a hierarchical architecture with
45
46 optimized resource selection methods; creation of orders for the items by the resource
47
48 agents, which could, in some cases, ask for different types of operation that could deliver
49
50 the same result; implementation of the model in industrial environments.
51
52
53
54
55
56
57
58
59
60

References

- Allwood, J. M. and Lee, J.-H., "The design of an agent for modelling supply chain network dynamics." *International Journal of Production Research*, 2005, **43**(22): 4875 - 4898.
- Baker, A. D., Parunak, H. V. D. and Erol, K., "Agents and the Internet: Infrastructure for Mass Customization." *Ieee Internet Computing*, 1999, 62-69.
- Bishop, P. and Warren, N., *JavaSpaces in Practice*, Addison-Wesley, 2003.
- Bruccoleri, M., Renna, P. and Perrone, G., "Reconfiguration: a key to handle exceptions and performance deteriorations in manufacturing operations." *International Journal of Production Research*, 2005, **43**(19): 4125 - 4145.
- Carvalho, J. D., Moreira, N. and Pires, L. C. M., "Autonomous Production Systems in virtual enterprises." *International Journal of Computer Integrated Manufacturing*, 2005, **18**(5): 357-366.
- Carvalho, J. D., Putnik, G. D. and Cunha, M., "Infrastructures for Virtual Enterprises." *Cadernos do Departamento de Produção e Sistemas, Escola de Engenharia da Universidade do Minho*, 2003, **DPS-03**(4). (2005/01/12: <http://www.dps.uminho.pt/cad-dps/cad-dps-detail.htm?rid=4&lang=pt>).
- Chan, F. T. S. and Chan, H. K., "A new model for manufacturing supply chain networks: a multiagent approach." *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 2004, **218**(4): 443-454.
- Chan, F. T. S. and Chan, H. K., "A simulation study with quantity flexibility in a supply chain subjected to uncertainties." *International Journal of Computer Integrated Manufacturing*, 2006, **19**(2): 148-160.

- 1
2
3
4 Chan, F. T. S., Chung, S. H. and Wadhwa, S., "A heuristic methodology for order
5
6 distribution in a demand driven collaborative supply chain." *International Journal of*
7
8 *Production Research*, 2004, **42**(1): 1-19.
- 9
10
11 Halter, S. L., *JavaSpaces: Example by Example*, Prentice Hall, 2002.
- 12
13
14 Jennings, N. R. and Wooldridge, M., Applications of Intelligent Agents, in *Agent*
15
16 *Technology Foundations, Applications and Markets* (N. R. Jennings and M.
17
18 *Wooldridge*), Springer Verlag, 1998.
- 19
20
21 Kim, Y., Choi, Y. and Yoo, S. B., "Brokering and 3D collaborative viewing of mechanical
22
23 part models on the Web." *International Journal of Computer Integrated Manufacturing*,
24
25 2001, **14**(1): 28-40.
- 26
27
28 Leitão, P., Restivo, F. and Putnik, G. D., A Multi-Agent Based Cell Controller, in
29
30 *Proceedings of the 8th IEEE International Conference on Emerging Technologies and*
31
32 *Factory Automation - ETFA 2001*, Antibes - Juan les Pins, France, 2001, 463-470.
- 33
34
35 Lima, R. M., Silva, S. C. and Martins, P. M., Sistemas Distribuídos de Produção, in *1º*
36
37 *Congresso Luso-Moçambicano de Engenharia* (J. F. S. Gomes, A. Matos and C.
38
39 *Afonso*), Maputo, Moçambique, 1999, B13-B20 (in portuguese).
- 40
41
42 McKay, K. N. and Wiers, V. C. S., "Integrated decision support for planning, scheduling,
43
44 and dispatching tasks in a focused factory." *Computers in Industry*, 2003, **50**(1): 5-14.
- 45
46
47 Nwana, H. S., "Software Agents: A Overview." *The Knowledge Engineering Review*, 1996,
48
49 **11**(3): 205-244.
- 50
51
52 Nwana, H. S. and Ndumu, D. T., "A Perspective on Software Agents Research." *The*
53
54 *Knowledge Engineering Review*, 1999, **14**(2): 1-18.
- 55
56
57
58
59
60

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

OMG, "Introduction to OMG's Unified Modeling Language™ (UML)." 2005. (2005/01/10:

http://www.omg.org/gettingstarted/what_is_uml.htm).

Parunak, H. V. D., Baker, A. D. and Clark, S. J., The AARIA Agent Architecture: An example of requirements-driven agent-based system design, in *Proceedings of the First International Conference on Autonomous Agents*, Maryna del Rey, CA, ACM Press, 1997, 482-483.

Parunak, H. V. D., Sauter, J., Fleischer, M. and Ward, A., "The RAPPID Project: Symbiosis between Industrial Requirements and MAS Research." *Autonomous Agents and Multi-Agent Systems*, 1999, **2** 111-140.

Rich, E. and Knight, K., *Artificial Intelligence*, 2nd. McGraw-Hill, 1991.

Scheer, A.-W., *ARIS - Business Process Frameworks*, 3rd. Springer-Verlag, 1999.

Shen, W., "Agent-Based Cooperative Manufacturing Scheduling: an Overview." *COVE Newsletter*, 2001 (2). (2005/01/10: <http://www.uninova.pt/~cove/newsletter.htm/2/Shen.pdf>).

Shen, W., Maturana, F. and Norrie, D. H., "MetaMorph II: an agent-based architecture for distributed intelligent design and manufacturing." *Journal of Intelligent Manufacturing*, 2000, **11**(3): 237-251.

Shen, W., Norrie, D. H. and Barthès, J. P., *Multi-agent Systems for Concurrent Intelligent Design and Manufacturing*, Taylor & Francis, 2001.

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

- 1
2
3
4 Sun, J., Zhang, Y. F. and Nee, A. Y. C., "A distributed multi-agent environment for product
5 design and manufacturing planning." *International Journal of Production Research*,
6
7 2001, **39**(4): 625-645.
8
9
10
11 Tharumarajah, A., Wells, A. J. and Nemes, L., "Comparison of the Bionic, Fractal and
12
13 Holonic Manufacturing System Concepts." *International Journal of Computer*
14
15 *Integrated Manufacturing*, 1996, **9**(3): 217-226.
16
17
18
19 Ulieru, M. and Cobzaru, M., "Building Holonic Supply Chain Management Systems: An e-
20
21 Logistics Application for the Telephone Manufacturing Industry." *IEEE Transactions on*
22
23 *Industrial Informatics*, 2005, **1**(1): 18-30.
24
25
26 Venkateswaran, J. and Son, Y. J., "Hybrid system dynamic-discrete event simulation-based
27
28 architecture for hierarchical production planning." *International Journal of Production*
29
30 *Research*, 2005, **43**(20): 4397-4429.
31
32
33 Warnecke, H. J., *The Fractal Company*, Springer-Verlag, 1993.
34
35
36 Wiendahl, H. P. and Ahrens, V., "Agent-based control of self-organised production
37
38 systems." *Annals CIRP*, 1997, **46**(1): 365-368.
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Table 1: Software agent definitions

Author	Definition
Nwana (1996)	<i>“When we really have to, we define an agent as referring to a component of software and/or hardware which is capable of acting exactly in order to accomplish tasks on behalf of its user.”</i>
Jennings and Wooldridge (1998)	<i>“First, an agent is a computer system situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives.”</i>
Parunak et al. (1999)	<i>“Agents add two things to (passive) objects: a local thread of control, and local initiative (usually expressed as local goals). Together, these enable the agent to monitor and respond to its environment autonomously (that is, without being externally invoked).”</i>

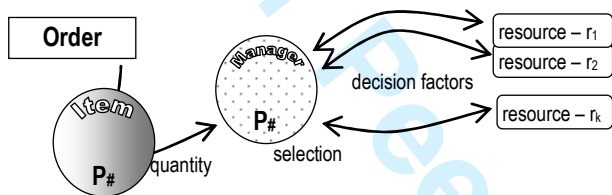


Figure 1: Local process for production resources selection.

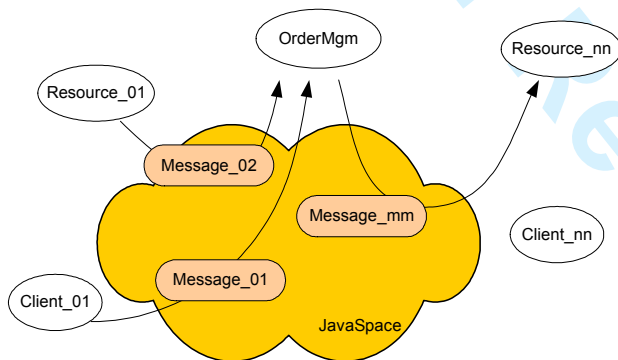


Figure 2: General structure of the agent model.

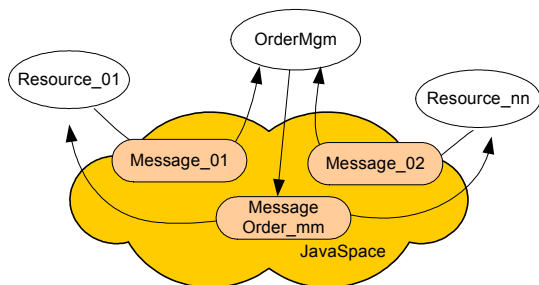


Figure 3: Elementary order task negotiation.

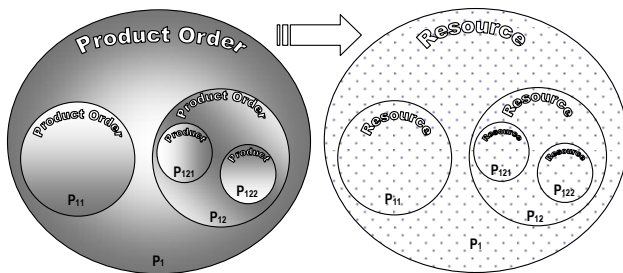


Figure 4: Production system example.

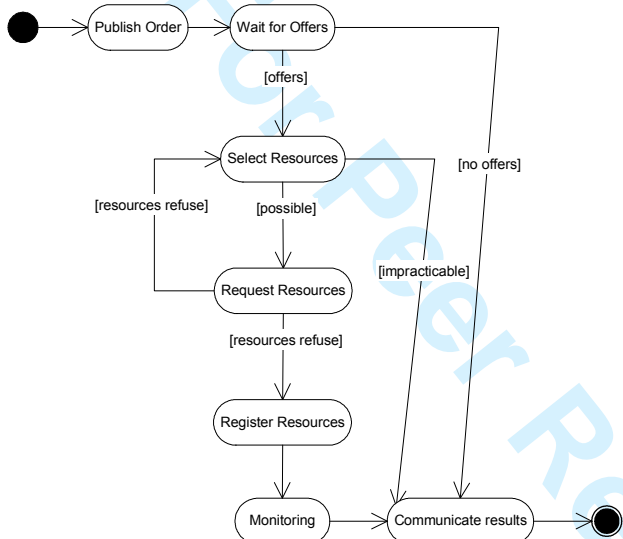


Figure 5: "Resource Selection" activity diagram.

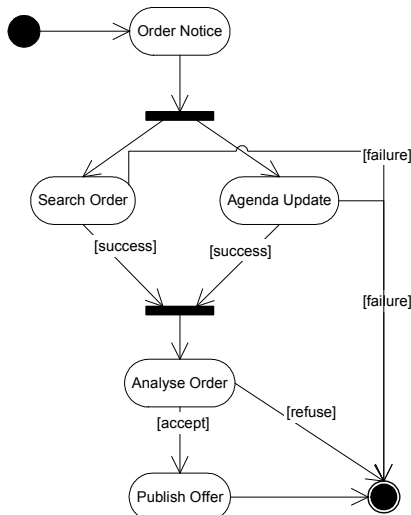


Figure 6: Agent Resource "Offer" activity diagram.

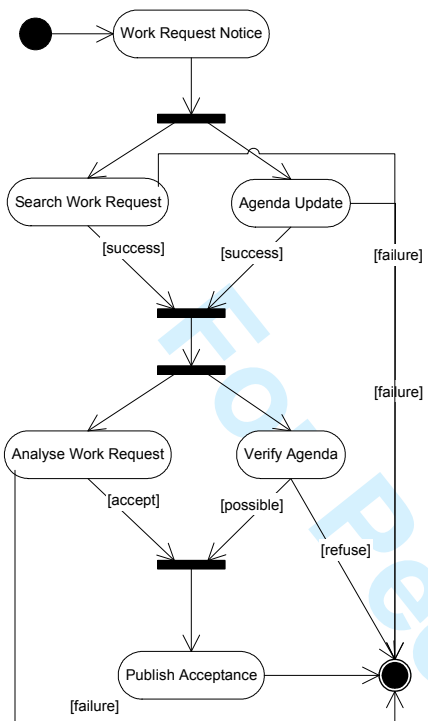


Figure 7: "Work Acceptance" activity diagram.

Message
-origin
-destination
-type
-param

Figure 8 : Message object.