A Framework for Improving Planning-Scheduling Collaboration in Industrial Production Environment

Pedro Gómez-Gasquet, Francisco-Cruz Lario, Rubén-Darío Franco, Víctor Anaya-Fons

Centro de Investigación de Gestión e Ingeniería de la Producción,
Universidat Politècnica de València, Cno. de Vera s/n,
Valencia, 46022, Spain,
{pgomez, fclario, dfranco, vanaya}@cigip.upv.es

Abstract: The work presented in this paper highlights how a real scheduling problem can be considered as a process avoiding the traditional operation research-based approach. Understanding the scheduling problem as a dynamic, uncertain and continuous process, in order to reduce the gap between reality and academic models, new conceptual elements emerge. The framework includes methodologies, concepts, architecture, and algorithms, which are based on the idea of a multiagent system in which planning-scheduling relationship is the core proposal. From the framework a software platform has been developed and tested in real industry.

Keywords: Planning-Scheduling collaboration, Framework, Multiagent.

1. Introduction

The objective of this paper is to present a modelling approach integrating conceptual models, architectures, methodologies and algorithms in the area of production scheduling. The scheduling problem exists not only in manufacturing enterprises, but also in public and service oriented organizations. It is typically NP-hard.

Problem representation and problem solution are strongly interconnected but each one has its own ways. In the paper we present a framework for collaborative production scheduling and we show how problem description and problem solution can be integrated. To achieve this we follow the Agent-based modelling paradigm.

Agent-based modelling has its roots in Artificial Intelligence (AI) and is further development in the area of Software Engineering. Essential for this kind of frameworks are efficient algorithms to find the problem solution. For the domain of production scheduling algorithms are mainly development at the border of two areas, Operations Research (OR) and Computer Science.

The remainder work is divided according to the following form: the second section review the background, the third section analyses the problem of collaborative scheduling, the fourth section deployed the framework proposed, and fifth section is dedicated to comment implementation and testing of proposed framework. The paper ends with a conclusion about the aforementioned information.

2. Background

Production scheduling is a large working area that usually is considered in OR discipline, as it has been previously highlighted. It is due to the way problems are focused and solutions are used [1]. In OR area problems are generally formally delimited creating a gap respect to real world. Users can hardly get the edge of some king of academic OR-based proposal.

It must be highlighted that OR-based approach has the origin when tools were lack of power in order to reach hard problems. Nowadays, actually industrial problems could be reach suitable solutions using lesson learnt in OR and IT areas. New challenges are waiting for [2], [25].

There are two big groups of researchers; those who try solve problems that have not yet been addressed, that trying to improve results (optimize if possible) of problems already discussed. The aim of this paper falls within the first one. We tackle a complex case demanded by practitioners. This one is characterized by dynamism and uncertainty, and usually need a convoluted decisional process, which introduces a multi-perspective and multi-objective system.

A changing problem where the relationships are the key piece requires a framework where quantitative and qualitative views are supported [8].

OR area and Software Engineering must work together to address complex problems. There are some scheduling systems available but differences in production shops and management approaches make it difficult to come up with standard scheduling solutions [3], and usually specific framework are needed. Quite often customization of a standard product turns the main contribution of final version [18].

There are some papers dealing with production scheduling frameworks. These contributions range from high-level description of main components of a general-purpose framework (some example in [8], [16], [20], [23], [26]). Some papers dedicate to deliver partial elements. Sometimes papers describe some conceptual models or architectures [6], [22], others papers introduces methodologies or algorithms [7], [31], [26].

Within the past decade, a large number of researchers have applied a new paradigm named agent technology in attempts to resolve scheduling problems [4], [14]. An extensive bibliography on agent/multiagent scheduling in manufacturing systems is compiled in [21]. Agent-based approaches have several potential advantages for a collaborative scheduling framework [24].

In addition, we have also considered works discussing the scheduling gap, the new challenges, and contributions presenting the requirements of scheduling systems. Even if most of these contributions do not explicit mention the components that should constitute a scheduling system, they serve to establish the boundaries of such a system, and provide information about some services and approaches that could be useful in this framework.

3. Collaborative Production Scheduling Problem

Scheduling problems can be formulated in a deterministic and a stochastic sense. In this paper we will only refer to stochastic problems. Production scheduling problems considered in this paper are characterized by three basic sets: set J of n Jobs J = {J1,J2,...,Jn}, set R of m resources R= {R1,R2,...,Rm} and set O of s operations O = {O11,O12,...,O1s,O21,...,O2s,…On1…Ons}. Scheduling usually means to assign resources from R to jobs from J in order to complete all tasks under the imposed constraints considered in O. In general each job has a specific route and the number of operations, i.e. s-value, and is different. In this sense, in order to model the problem selected, it has been included a set O-R of where resources and operations are linked. O-R set will determine if there are some resources available to carry out an operation (multi-processor or hybrid shop, technical constraints, etc.). It let us consider the most general scheduling problem, job-shop. J, R, O, and O-R are sets informing about static characteristics.

The problem considered introduces uncertainty by 3-ways:

- Imprecise processing times.
- Sequence dependent set-up times.
- New jobs arrivals.

New sets, generally matrix, should be used to contain processing and set-up time information. In this context production scheduling keeps as complex problem and yet does not provide stable results over time due to changing environment that requires continuous model and the solution refining.

Traditionally, are considered in an isolated decisional level where constrains were including not only technical aspect but requirements from other decisional levels. When production scheduling problem is considered as “a live problem” a new approach should be provided to avoid the requirements from other decisional levels remain unchanged when circumstances have changed.

The framework proposed in the following paragraphs is based on the idea that the key is to "manage the scheduling process". It supposes:

- The main actions are dedicated to update information, constraints and requirements among all actors involved.
- Decisional levels collaborate in an orderly way to solve the problem.
- Algorithms as resolution methods are in the background.

In other words we propose migrate from the quantitative view to the decisional view, and model the process not as specific action but set of decisions, actions, calculations coordinated. The goal more difficult to achieve is to build a collaborative system.

Scheduling level interacts with planning level, control level, product design level and engineering level. But in operative environment planning and control levels are the more representatives. In this work planning-scheduling relationship is considered. Several paper address planning-scheduling integration [11], [15] but no more address planning-scheduling collaboration in a distributed and dynamic context from the scheduling point of view [12], [28].

4. Agent-Oriented Approach

In Figure 1 is shown a picture where problem framework and proposal framework is related. Problem framework is presented as a double level structure (manage and operative). And the proposal framework is introduced as container (platform) including some relevant utilities (algorithms). It is useful to understand the methodology, the conceptual items and the architecture used.

We related a software platform based on multiagent systems (MAS) to manage level facilities (coordination, communication, etc.).

Two major software blocks are linked with predictive and reactive scheduling business processes; both are used considering each time global and scheduling level key performance indicators.

When two decisional levels are interacting asynchronously each one with its own objectives and each one with different know-how and tools it is recommend the use of agent paradigm [29].

According to Software engineering foundations the following iterative steps are the core of the methodology used to develop the platform:

- Requirement analysis.
- Solution Design.
- Tool Implementation.
- Validation and Test.

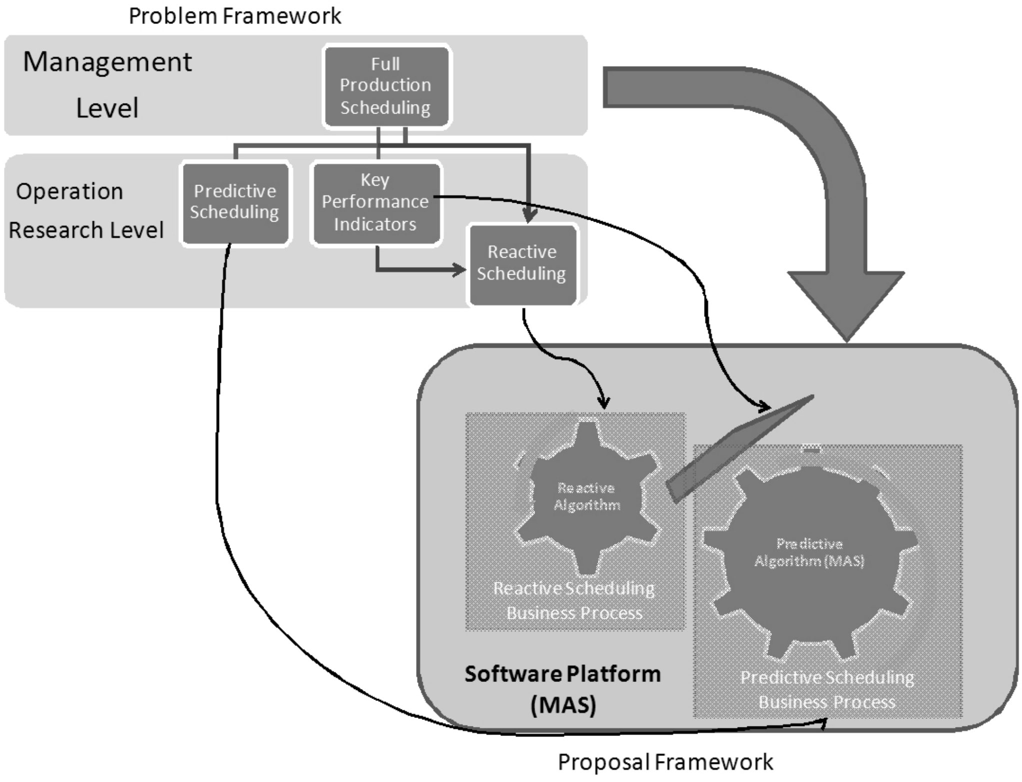

Figure 1. Problem and proposal framework.

Each step has been refined by selection and adaption of the most suitable existing proposals. We would like to highlight the requirement and analysis steps.

The analysis of the requirements has been obtained by applying CIMOSA [5], [13], [30]. CIMOSA was proposed during 90 decade and it covers the complete life cycle in software development. Only requirement analysis has been used in the framework from CIMOSA.

The reader should consider the proposed system to be the structural element on which the algorithms, methods, and other resources that are oriented to the scheduling in changing environments will subsequently be placed or where it is necessary to apply rescheduling of the previously established schedules. The functional coordination between the environment of the planner and the scheduler, in this type of environment, is based on cooperation. Communication is the fundamental facility of this software element.

Initially identified solution is based on the agent-based. Therefore, its design itself includes the employee methodological proposal INGENIA [17], which facilitates the development of systems that are based on agents and multiagents. Although the design is closely linked to the Software Engineering, a vision of the employed methodology is provided in order to connect it with the most conceptual levels of the process of production scheduling.

The following sub-paragraphs show how the methodology is applied. This process helps to identify conceptual and software elements that will contribute to build concrete architecture. Methodology, conceptual elements and architecture are the core of the framework will be presented and the foundations for implementing the software platform.

4.1 Analysis of collaborative scheduling requirements

To proceed in the phase of requirement analysis, CIMOSA methodology has been selected. It provides the advantage of offering a vision of the problem to analyse that is structured according to various views (i.e., functional, organizational, resource-based, and informational) that enrich the final analysis. From the functional point of view production scheduling is defined like a domain; that is, directly related to the domains of the planning (i.e., tactical), the design of processes and products (i.e., strategic) and the operations control (i.e., operating) as can be seen in Figure 2.

The objective that is defined inside the production scheduling domain is to obtain production schedules in which the average of productivity that is reached in the different periods is the highest that is possible. This aspect is fundamental and, from our point of view, implies that the quantitative methods are suitable to give solutions to the corresponding combinatorial problems as well as fundamentally

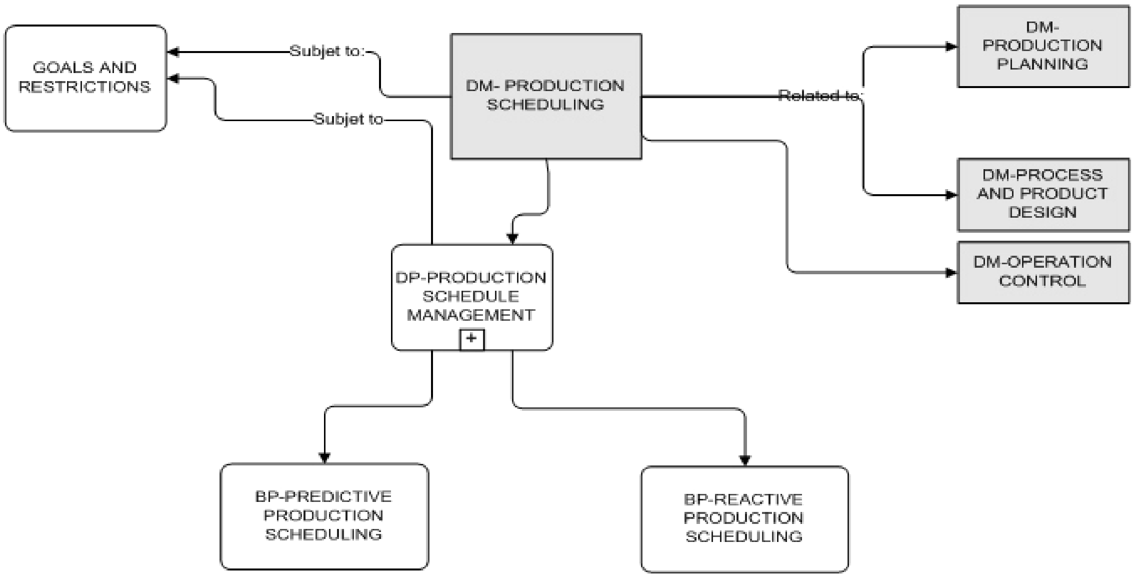
GOALS AND RESTRICTIONS
Subjet to:
Subjet to
DM- PRODUCTION SCHEDULING
Related to:
DM- PRODUCTION PLANNING
DM-PROCESS AND PRODUCT DESIGN
DM-OPERATION CONTROL
DP-PRODUCTION SCHEDULE MANAGEMENT
+
BP-PREDICTIVE PRODUCTION SCHEDULING
BP-REACTIVE PRODUCTION SCHEDULING

Figure 2. Main Elements in the Functional View of Production Scheduling Domain.

promoting the contribution among the planner. The planner decides the lots that one must carry out in each period. The scheduler manages the capable manufacturing system that offers the service in question. In this sense, one must emphasize that the suitable adjustment of the capacity to the needs of each moment will be the one that facilitates the attainment of the greatest productivity average.

Inside the production scheduling domain, only the domain process has been defined (DP-production schedule management) as being responsible for managing the allocation, sequencing, and timing of the lots that the planner has requested to be manufactured in a determined period. Such management implies that the initial schedule is obtained by means of a business process (BP-predictive PS) that employs some algorithm that is adjusted to the characteristics of the problem in function of the known data to priori (e.g., static or predictive schedule). At the beginning of the period, the initial schedule will become the schedule in process. A new business process will run (BP-REVIVE PS) to detect events and, in that case, to try those that can affect the schedule in process (EV-upset, EV-new period, EV-restart) and imply the execution of some method to readjust the schedule in process to the current reality.

The proposed system is focused on a make-to-stock management, although other type of management will be tackled easily. Algorithms and methods should be adapted to each problem. What is provided is an architecture that enables the coordination between the planner and the scheduler to adjust the lot number and its volume of each period by means of an iterative process to carry out the major number of possible units. For example, consider the following in the case of:

– Predictive scheduling (i.e., initial schedule). Normally an upset among the employed times for the calculation in the environment of planning exists and the available time in the environment of scheduling once generated the schedule (e.g., the variable times of process and setups, dependence of the sequences, etc.).

– Reactive scheduling (i.e., schedule in process) consists of detecting the possibility of incorporating some new lot if the schedule in process advances faster than predicted. In other cases, it consists of cancelling some lot that is included on the schedule in process if some delay that impedes the ability to carry out the predicted schedule occurs inside the current period.

One must remember that is possible that the initial schedule include some idle time that provides a solution to some small incident that is not predicted without modifying the initial selection of lots. This situation is normal, in order to avoid a high number of cancellations or incorporations of lots in each period.

The views of information and resources indicate that they have established the type of data and capacities that are necessary to manage the productive schedule.

4.2 Design of scheduling system

Complex manufacturing systems consist of a number of related subsystems that are usually organized in a hierarchical fashion. At any given level, subsystems work together to achieve the functionality of their parent system. Each component can be thought as achieving one or more objectives. Thus, entities should have their own thread of control (i.e., they should be active), and they should have control over their own actions (i.e., they should be autonomous). Given this fact, it is apparent that the natural way to modularize a complex system is in terms of multiple autonomous components that act and interact in flexible ways to achieve their objectives. Therefore, the agent-oriented approach is simply the best fit.

Description of modelling process

INGENIAS methodology employs several meta-models and a meta-model language for constructing models. All meta-models are based on objects, attributes, and relationships. INGENIAS methodology also integrates its meta-models into the Rational Unified Process (RUP) for developing software systems and offers a graphical development tool called the INGENIAS Development Kit (IDK).

During analysis and the design phases, five different meta-models are used: (i) an organization meta-model, which defines how agents are grouped and identifies the system functionality and the existing constraints in the agents' behaviour; (ii) an agent meta-model, which describes the particular agents to be used and their internal mental states; (iii) an interaction meta-model, which details how agents are coordinated and interact; (iv) an environment meta-model, which defines the types of resources and applications that are used by the system; and (v) a tasks and objectives meta-model, which relates the mental state of each agent with its tasks.

Use case diagrams

A use case diagram provides a snapshot model of a set of system behaviours that meets a user goal. Thus, this description represents a functional requirement that shows what happens, but not how it is achieved by the system. As previously mentioned, our study is focused on the scheduling system, in which four main use cases can be identified. In the "calculate predictive scheduling" use case, a feasible initial schedule to be carried out in the following days is created. This schedule is developed according to the manufacture lots that are defined in the master plan which is included in the planner set of use cases (not shown) but that is available with the "get periodic plan" use case. In the "reactive production adjustment" use case, previous schedules in which problems developed during execution are modified. Therefore, those schedules are reconfigured in order to adjust to factory changes. The schedule execution monitoring data is included in the last use case.

Organization model

The organization model is defined by the organizational goals and tasks, the workflows that determine associations among tasks and general information about their execution; groups, which may contain agents, roles, resources or applications; and social relationships.

Several roles are distinguished in the organization model for the scheduling process (Figure 3).

(1) Planner, which is responsible for selecting the lots in a period; (2) coordinator, which maintains information about all plants' configuration and knows all restrictions and features of each machine and plant element; (3) scheduler, which has the ability to schedule tasks and resources and supervises actual execution of a schedule in a specific plant; and (4) worker, which oversees the schedule execution at the factory.

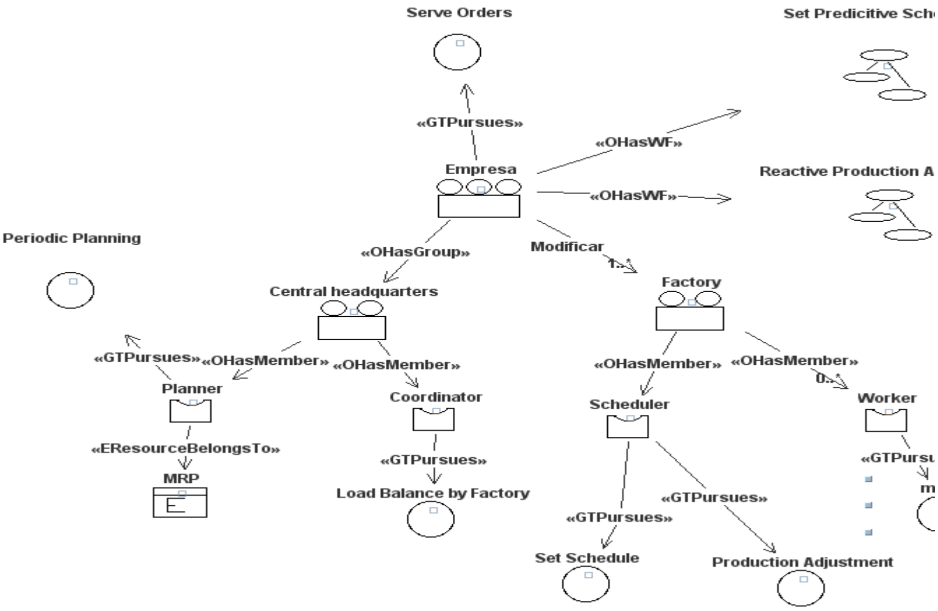

Figure 3. Scheduling Process Organization Model.

Interaction model

The interaction model is composed of the set of diagrams that capture the interaction between agents of the scheduling system. Each interaction diagram shows an interaction protocol, that is, the interactions, their order and the messages exchanged between a pair of agents. Interactions are one of the key points of agents as far as is within the interaction where agents take choices to achieve their specific goals. Without interacting, agents could never reach their goals. Interactions consider agents' goals but also the specific mind states of the involved agents in a specific moment. Interactions consider a common goal, but also specific agents' goals that in some way must be balanced to achieve an optimum. At the same time, interactions are not a well define set of steps predefined that are all the time executed in the same way, instead, they sketch phases and behavioural patterns that will be different in each specific execution.

For example, we have the planner and scheduler agents both pursuing the same goal, to produce as many pieces as possible with the available resources. At the same time, planner have a finer goal, assign plan according to parameters like capacity, proximity, etc. Scheduler agent's sub-goal is to keep a balance between productivity, resources and times, keeping a safety span time. Both goals are in some point contradictory, and a proper negation between one planner agent and more than one scheduler agents. A planner proposes production plans to schedulers of each plant according to its specific goals. Scheduler agents compute that plans and accept or reject. This interaction is not only complex but also dynamic. If any production plan has any technical problem and the work schedule is unviable, scheduler must interact with the planner to readjust the planning and scheduling of the whole network of agents. All this behaviour is expressed in INGENIAS using two types of diagrams, the interaction diagram and the specification of an interaction.

Agent model

In an agent model, a specific agent has been assigned to each role that is identified in the organization model. The goals, tasks, and mental states have to be associated for each agent. Three types of agents have been identified: Planner, Scheduler and Worker. Each one plays the role associated with its name.

We have addressed the focus on Scheduler agent. Figure 5 shows the scheduler agent model. The scheduler agent plays the scheduler role and this example has two main goals (e.g., set a schedule and increase the production rate average) and a set of tasks (e.g., check and load data, parameter adjustment, calculate predictive/reactive scheduling, tuning the schedule, validate/ reject the results, notify a new schedule).

The interaction diagram gives a name to the interaction and identifies goals and agents involved and identify the interaction specification that describes the negotiation between agents. An interaction model has traceability with each use case. Figure 4 shows the defined interactions, the involved agents, and the interaction unit (i.e., message) that is used.

The specification of the interaction describes the nature of the interaction, the steps that compose the iteration protocol, agents involved at each step, messages, mental steps of the agents, and so on.

They are achieved by means of a set of mental states. In the same way, a planner and a worker agent model exist. The first one plays the role of planner and coordinator, and the second plays the role of worker.

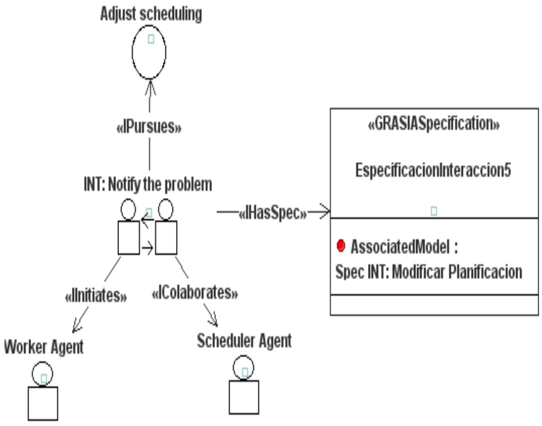

Figure 4. Example of interaction diagram.

Figure 5 shows how a scheduler agent could be in a different mental state ("AHasMS"). For example, "waiting" for a proposal/message, "scheduling" a batch of jobs, "ask for modifying" a batch of jobs, etc.

From a specific mental state, the agent scheduler must reach his goals by using the available functions and tools.

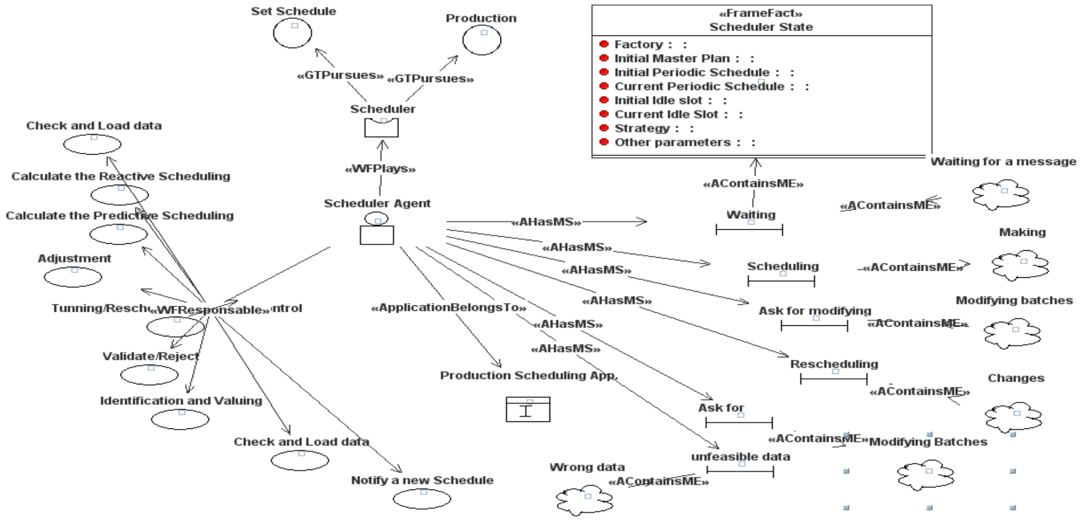

Figure 5. Scheduler Agent Model.

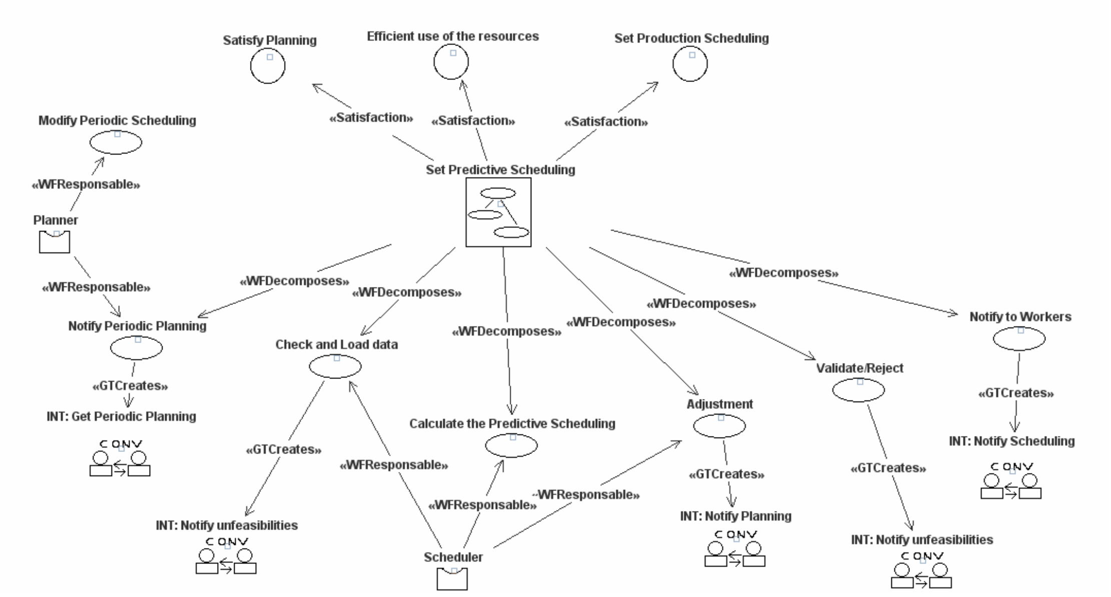

Figure 6. Predictive Scheduling Tasks/Goals Model.

Task/Goal model

This model attempts to answer the questions of why, who, and how throughout the analysis process. "Why" refers to the goals that are defined for the system; "who" refers to the agents which are responsible for the goal fulfilment; and "how" refers to the set of tasks which are defined to achieve the goals.

Two major goals have been established: "predictive scheduling" and "scheduling adjustment or rescheduling".

Predictive scheduling comes from the functional view of the production scheduling domain, which is carried out in CIMOSA analysis.

Figure 6 shows how this workflow must satisfy three main goals (i.e., satisfy planning, efficient use of resources, and set production scheduling). The "conversation" items that represent interactions between two agents (e.g., scheduler-planner and scheduler-worker) could be highlighted in the figure. These conversations are the core of the negotiation process that occurs in almost all functional processes.

The functionality "Calculate the predictive scheduling" is supported by SMAGA algorithm. It is a Genetic Algorithm developed specifically to generate a predictive schedule for the problem describes in the paragraph 3. In the Fgure 7 is shown the general process.

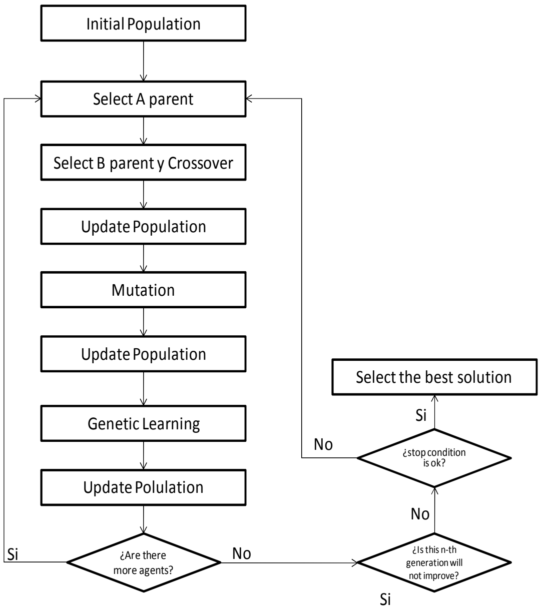

Figure 7. Basic process of SMAGA Algorithm

Regarding our analysis, the decomposition of the schedule creation workflow is shown in Figure 8 as an example of the tasks/goals model. The associated tasks are the different steps of this workflow, which is accomplished according to concrete roles.

Scheduling adjustment comes from the functional view of the production scheduling domain, which is carried out in CIMOSA analysis that is named “reactive scheduling.” Figure 8 shows how this workflow adds one additional goal, “keep a stable scheduling,” and displays what is relevant and characteristic about this function. This task maintains “conversations” as core elements in the process.

In this case the functionality “Calculate the reactive scheduling” is supported by a set of heuristics that perform partial or full rescheduling.

One of them is selected according with the problem status using a decision rule.

Environment model

The environment model of the production scheduling organization is shown in order to consider that an internal application is managed by the planner to store and update executed plans that are created by the organization.

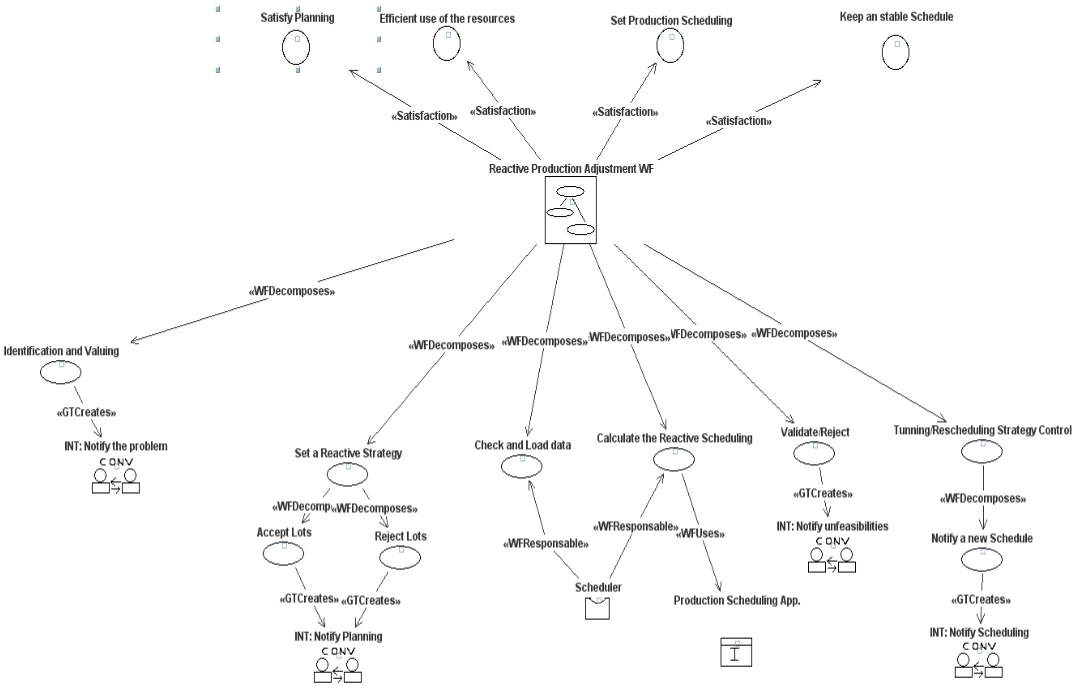

Figure 8. Scheduling Adjustment Tasks/Goals Model.

Model overview

In the Fgure 9 is shown a general platform overview. In the figure three agents (Planner, Scheduler and Worker) are collaborating using five basic interaction protocols: Some of them have been used between planner agent and scheduler agent: Notify Planning, Notify Infeasibilities and Modify Planning. Others interactions have been used between scheduler agent and worker agent: Notify the Problem and Notify Scheduling.

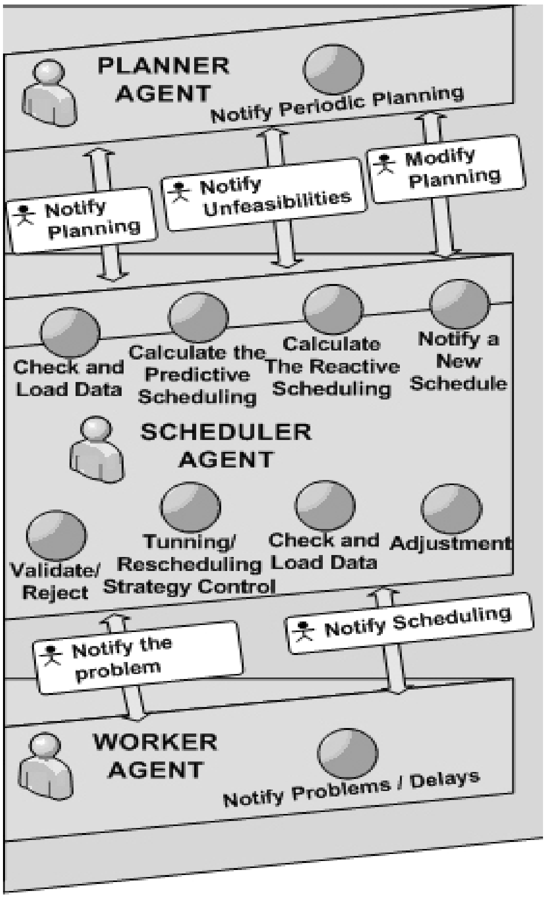

Figure 9. Platform Overview.

Circular icons represent task that agents must execute after mental state is analysed and decision rules select the suitable one.

Planner agent has been modelled partially. Only interaction, task, etc. that can potentially improve the scheduling role have been taken into account. Worker agent is included only as residual but needed function.

It must be highlighted that scheduler agent model could be implemented with several agents. It scheduler agent is responsible only for a factory, so in a multi-plant enterprise case it should be replicated.

5. Implementation and Testing

The resulting proposed framework software has been transformed in a prototype in order to be used in a typical make-to-stock industry, such as a ceramic tile enterprise. The programming and deployment software have been carried out by using the JADE platform (http://jade.tilab.com/) and JAVA language. The methods and algorithms have been developed in order to solve a hybrid flow-shop problem with sequence-dependent setup time [9], [10], [18].

After a Design of Experiment (DoE) process where 21 type of event, 4 levels in one operative factor and 120 sets of jobs types were considered for each of 8 different scenarios. During the experiment 4 runs by combination were used reaching 322.560 runs.

Comparing the current processes at the selected enterprise and the new options that are provided by the prototype, one could observe that, as the top productivity rate is attained (i.e., 10%-25%, depending on the scenario) in the second case, it can be attributed to the overall reduction of idle time. These times have reduced as much for the relocation/sequencing of the lots in process as they have for the incorporation of new lots.

6. Conclusions

This paper presents a framework, based on multi-agent technology. It should help to improve production planning-scheduling collaboration.

The proposal address solutions not only initial solutions (predictive) but modification of this one (reactive) if changes appear. That is, the concept of scheduling problem as a decisional process where several actors are involved is the basic change form traditional OR approach.

The problem has been analysed considering the four views of CIMOSA methodology. According with the actual dynamic production processes requirements that allow the selling process to be realized in an online platform. Our approach is based on a medium-time project. This proposal is based on the system capability that offers the most suitable product alternative introducing a powerful planning-scheduling interact means a several communication protocol and functionality.

In order to design the system it is necessary to integrate the various distributed production steps in a flexible, adaptable, versatile, robust, and natural way. Agent/multi-agent systems technology has been used in the resolution of this problem, since it provides the required characteristics for manufacturing systems. INGENIAS methodology has supported the design step. We have currently centred our design on agent entities (tasks, interactions, integration in the organization, etc.) from the point of view of the scheduling level.

Once the framework has been implemented using JADE environment and JAVA language the resulting software has been tested in the real ceramic tile industry.

Acknowledgements

This work has been carried out as part of the project “DPI2008-06788-C02-01 (PERMACASI)” funded by Ministerio de Ciencia e Innovación of the Spanish Government and as part of the project “PAID-06-10-2396 (NegoSol-MAS)” funded by Vicerrectorado de Investigación of Universidat Politècnica de València.

REFERENCES

1. BLAZEWICZ, J., K. ECKER, G. SCHMIDT, J. WEGLARZ, **Scheduling in Computer and Manufacturing Systems,** Springer, Berlin, 1993.
2. BIUNDO, S., R. AYLETT, M. BEETZ, D. BORRAJO, A. CESTA, T. GRANT, L. KCCLUSKEY, A. MILANI, G. VERFAILLIE, **Technological Roadmap on IA Planning and Scheduling**. Planet (ENoE in AI Planning), 2003.
3. BRANDIMARTE, P., M. RIGODANZA, L. ROERO, **Conceptual Modeling and Object-oriented Scheduling Architecture Based on Shifting Bottleneck Procedure,** IIE Transactions, vol. 32(10), 2000, pp. 921-929.
4. DAOUAS, T., K. GHEDIRA, J. P. MULLER, **How to Schedule a Flow Shop Plant by Agents, in Applications of Artificial Intelligence in Engineering**, Billerica, MA: Computational Mechanics Inc., 1995, pp. 73–80.
5. ESPRIT Consortium AMICE. CIMOSA, 1993, **Open System Architecture for CIM**.
6. FRAMIÑAN, J. M., R. RUIZ, **Architecture of Manufacturing Scheduling Systems: Literature Review and an Integrated Proposal**, European Journal of Operational Research, vol. 205, 2010, pp. 237-246.
7. GÓMEZ-GASQUET, P., R. RODRIGUEZ-RODRIGUEZ, R. D. FRANCO, A. ORTIZ, **A Collaborative Scheduling GA for Products-packages Service within Extended Selling Chains Environment**, Journal of Intelligent Manufacturing, 2010 DOI 10.1007/s10845-010-0434-z
8. GOMEZ-GASQUET, P., **Programación de la producción en un taller de flujo híbrido sujeto a incertidumbre: arquitectura y algoritmos. Aplicación a la industria cerámica**, Dtors. Lario-Esteban, F. C., Andrés-Romano C. UPV, 2010, http://hdl.handle.net/10251/7728.
9. GOMEZ-GASQUET, P., C. ANDRES, J. P. GARCÍA-SABATER, **Dynamic Hybrid Flow-Shop Scheduling with Due Dates and Sequence Dependent Setup Times,** Production Planning and Scheduling (PMS '04), Nancy, France. Proceedings, 2004, pp. 254-259.
10. GUPTA, J. N. D., **Flowshop Schedules with Sequence Dependent Setup Times**, Journal of the Operations Research Society of Japan, vol. 29(3), 1986, pp. 206-219.
11. HUANG, S. H., H. C. ZHANG, M. L. SMITH, **A Progressive Approach for the Integration of Process Planning and Scheduling**, IEEE Trans., vol. 27(4), 1995, pp. 456–464.
12. KEMPENAERS, J., J. PINTE, J. DETAND, J. P. KRUTH, **A Collaborative Process Planning and Scheduling System**, Advanced Engineering Software, vol. 25(1), 1996, pp. 3–8.

13. KOSANKE, K., **Cimosa - Overview and Status**, Computers in Industry, vol. 27(2), 1995, pp. 101-109.

14. LIU, J., K. P. SYCARA, **Distributed Problem Solving through Coordination in a Society of Agents**, 13th Intl. Workshop on DAI, 1994.

15. MAMALIS, A. G., I. MALAGARDIS, K. KANBOURIS, **Online Integration of Process Planning Module with Production Scheduling**, International Journal of Advanced Manufacturing Technology, vol. 12(5), 1996, pp. 330-338.

16. MONCH, L., S**cheduling Framework for Jobs on Parallel Machines in Complex Manufacturing Systems**, Wirtschaftsinformatik, vol. 46(6), 2004, pp. 470-480.

17. PAVON, J., J. GOMEZ-SANZ, **Agent Oriented Software Engineering with INGENIAS, Multi-Agent Systems and Applications II**, Proceedings, 2691, 2003, pp. 394-403.

18. PINEDO, M. L., **Planning and Scheduling in Manufacturing and Services**, Third ed. Springer, Berlin, 2007.

19. RAJKUMAR, R., P. SHAHABUDEEN, P. NAGARAJ, S. ARUNACHALAM, T. PAGE, **A BiCriteria Approach to the M-machine Flowshop Scheduling Problem,** Studies in Informatics and Control, Vol. 18(2), 2009, pp. 127-136.

20. RELVAS, S., H. A. MATOS, A. P. F. D. BARBOSA-POVOA, J. FIALHO, **Reactive Scheduling Framework for a Multiproduct Pipeline with Inventory Management**, Industrial & Engineering Chemistry Research, vol. 46(17), 2007, pp. 5659-5672.

21. SCHIEGG, P., **Bibliography on Multi-Agent Scheduling in Manufacturing Systems**., http://farm.ecs.umass.edu/˜pschiegg/bib/lit.html

22. SCHMIDT, G., **Modelling Production Scheduling Systems,** International Journal of Production Economics, vol. 46, 1996, pp. 109-118.

23. SHAKERI, S., R. LOGENDRAN, **A Mathematical Programming-based Scheduling Framework for Multitasking Environments**, European Journal of Operational Research, vol. 176(1), 2007, pp. 193-209.

24. SHEN, W., **Distributed Manufacturing Scheduling using Intelligent Agents**, IEEE Expert/Intell. Syst., vol. 17(1), 2002, pp. 88–94.

25. SMITH, S., **Is Scheduling a Solved Problem? The Next Ten Years of Scheduling Research**, Cowling, P. and Kendall, G. Ed. San Francisco, CA, 2001, pp. 116-120.

26. SUN, D., L. LIN, **A Dynamic Job-Shop Scheduling Framework - A Backward Approach,** International Journal of Production Research, vol. 32(4), 1994, pp. 967-985.

27. WADHWA, S., J. MADAAN, R. RAINA, **A Genetic Algorithm Based Application for a Flexible System**, Studies in Informatics and Control, vol. 16(2), 2007, pp. 171-184.

28. WANG, L., W. SHEN, **DPP: An Agent-based Approach for Distributed Process Planning**, Journal of Intelligent Manufacturing, vol. 14(5), 2003, pp. 429–440.

29. WOOLDRIGE, M., N. R. JENNINGS, **Intelligent Agents: Theory and Practice,** Knowledge Engineering Review, vol. 10(2), 1995, pp. 115-152.

30. ZELM, M., F. VERNADAT, K. KOSANKE, **The CIMOSA business Modelling Process**, Computers in Industry, vol. 21(2), 1995, pp.123-142.

31. ZWEBEN, M., M. S. FOX, **Intelligent Scheduling**, Morgan Kaufmann, San Francisco, CA, 1994.