Multi-Agent System for Dynamic Manufacturing Scheduling using Meta-Heuristics

ANA MADUREIRA

Computer Science Department
Institute of Engineering - Polytechnic of Porto
GECAD – Knowledge Engineering and Decision Support Research Group
Porto, Portugal

http://www.dei.isep.ipp.pt/~anamadur/

Abstract: - In this paper we will model a Manufacturing System by means of a Multi-Agent Systems, where each agent may represent a processing entity. This work has as objective to deal with the complex problem of Dynamic Scheduling in Manufacturing Systems. We want to prove that a good global solution for a scheduling problem may emerge from a community of machine agents solving locally their schedules and cooperating with other machine agents that shares some relations between the operations/jobs.

The proposed approach is in line with reality and away from the approaches that deal with static and classic or basic Job-Shop scheduling problems. In fact, in real world, where problems are essentially of dynamic and stochastic nature, the traditional methods or algorithms are of very little use. This is the case with most algorithms for solving the so-called static scheduling problem for different setting of both single and multi-machine systems arrangements. This reality, motivated us to concentrate on tools, which could deal with such dynamic, disturbed scheduling problems, for multi-machine manufacturing settings, even though, due to the complexity of these problems, optimal solutions may not be possible to find.

Key-Words: - Multi-Agent Systems, Dynamic and Distributed Scheduling, Meta-Heuristics, Manufacturing

1 Introduction

E-business trends are making it critical for manufacturers to integrate sales-order entry and production scheduling. Customers want to place their orders via Internet and have their product built to their specific needs as quickly as possible. This engineer-to-order approach requires that manufacturers build orders in small quantities to deal with an ever-increasing number of market segments. It might even require that manufacturers build orders in lot sizes of one.

The ability to deal with a larger number of smaller quantity orders increases the stress on the scheduling process. A dynamic scheduling system takes into consideration what equipment and inventory is required and available to fulfill an incoming order. In doing so, equipment could be better utilized and work-in-progress dramatically reduced.

The planning of Manufacturing Systems involves frequently the resolution of a huge amount and variety of combinatorial optimisation problems with a important impact on the performance of manufacturing organisations. Examples of those problems are the sequencing and scheduling problems in manufacturing management, routing and transportation, layout design and timetabling problems. The classical optimisation methods are not enough for the efficient resolution of those problems or are developed for specific situations.

The interest of the Meta-Heuristic (MH) approaches is that they converge, in general, to satisfactory solutions in an effective and efficient way (computing time and implementation effort).

There have been significant advances in the theory and application of Meta-Heuristics to solve hard optimization problems. The family of Meta-Heuristics includes, but it is not limited to Tabu Search, Simulated Annealing, Soft Computing, Evolutionary Methods, Adaptive Memory procedures, Ant Systems, Scatter Search and their hybrids.

This paper outlines the limitations of static approaches to scheduling in the presence of dynamic environments and gives a review of currently developing research on real-world scheduling problems, which are often complex, constrained and dynamic.

A Tabu Search based Scheduling system for solving a class of real world scheduling problems is described. Then, we will model a Manufacturing Systems by means of a Multi-Agent Systems, where each agent may represent a processing entity (machine). This system has as objective to deal with the complex problem of Dynamic Scheduling in Manufacturing

Systems. We will try to prove that a good global solution for a scheduling problem may emerge from a community of machine agents solving locally their schedules and cooperating with other machine agents that shares some relations between the operations/jobs.

The remaining sections are organized as follows: Section 2 summarizes some related work. In section 3 the scheduling problem under consideration is presented. The proposed TS based scheduling system is described on section 4. Section 5 presents the Multi-Agent System for Dynamic Manufacturing Scheduling. Finally, the paper presents some conclusions and puts forward some ideas for future work.

2 Related Work

Scheduling Problems can be seen as a decision making process, which decides when an operation should start and which resources should be used for each operation. A significant variety of constraints such as operation processing times, release and due dates, precedence constraints and resource availability can affect the scheduling decision.

The major approaches of scheduling are represented by the traditional operational research techniques as combinatorial optimization procedures, simulation, network methods, heuristic approaches and constraint analysis [5].

Traditional scheduling methods, encounter great difficulties when they are applied to some real-world situations. This is for three main reasons. Firstly, traditional scheduling methods use simplified and deterministic theoretical models, where all problem data are known before scheduling starts. However, many real world optimization problems are dynamic and non-deterministic and, in which changes may occur continually. The models seldom match the real-world scheduling environment, which is characterized by ever-changing task requirements, occurrence of a variety of unexpected occurrences and perturbations, such as new job arrivals, machine breakdowns, employees sickness, jobs cancellation and due date and time processing changes, complicated and dynamic resource constraints, uncertain processing times, and multiple or even conflicting scheduling objectives. In practice, static scheduling is not able to react dynamically and rapidly in the presence of dynamic information not previously foreseen in the current schedule.

Secondly, most of the approximation methods proposed for the Job-Shop Scheduling Problems (JSSP) are oriented methods, i.e. developed specifically for the problem in consideration. Some examples of this class of methods are the priority rules [3][7][10][12] and the Shifting Bottleneck proposed in [4].

Finally, traditional scheduling methods are essentially centralized in the sense that all the computations are carried out in a central computing and logic unit. All the information concerning every job and every resource has to go through this unit. This centralized approach is especially susceptible to problems of tractability, because the number of interacting entities that must be managed together is large and leads to a combinatorial explosion. Particularly since, a detailed schedule is generated over a long time horizon, and planning and execution are carried out in discrete buckets of time. Centralized scheduling is therefore large, complex, and difficult to maintain and reconfigure. On the other hand, the inherent nature of much industrial and service process is distributed. Consequently, traditional methods are often too inflexible, costly, and slow to satisfy the needs of real-world scheduling systems.

Several attempts have been made to modify algorithms, to tune them for optimization in a changing environment. It was observed in manufacturing all these studies, that the dynamic environment requires an algorithm to maintain sufficient diversity for a continuous adaptation to the changes of the landscape. Although the interest in optimization algorithms for dynamic optimization problems is growing and a number of authors have proposed an even greater number of new approaches, the field lacks a general understanding as to suitable benchmark problems, fair comparisons and measurement of algorithm quality.

Recently the scheduling problem in dynamic environments have been investigated by a number of authors especially in the evolutionary community, see for example [11][13].

In spite of all the previous trials the scheduling problem still known to be NP-complete. This fact incites researchers to explore new directions. Multi-Agent technology has been considered as an important approach for developing industrial distributed systems. The problem of dynamic scheduling is one that is receiving increasing attention amongst both researchers and practitioners. Researchers have begun to try solving the complex dynamic scheduling problems in a distributed way using the Multi-Agent paradigm. The Multi-Agent paradigm represents one of the most promising approaches to building complex, flexible, and cost-effective scheduling systems because of its distributed and dynamic nature [6][9]. The idea is to consider the scheduling problem in terms of conflict, concurrency and cooperation within a society of agents [8][14]. The autonomous agent approach appears to offer some significant advantages. Because overall system behavior emerges from local decisions, and the software for each agent is much shorter and simpler than would be required for a centralized approach and the result is easier to write, debug, and maintain.

3 Problem Definition

The scheduling problem can be seen as a decision making process for operations starting and resources to be used. A variety of characteristics and constraints related with jobs and production system, such as operation processing time, release and due dates, precedence constraints and resource availability, can affect scheduling decisions. For literature on this subject, see for example [3] [7] [10][12].

If all jobs are known before processing starts a scheduling problem is said to be static, while, to classify a problem as dynamic it is sufficient that job release times are not fixed at a single point in time, i.e. jobs arrive to the system at different times. Scheduling problems can also be classified as either deterministic, when processing times and all other parameters are known and fixed, or as non-deterministic or stochastic, when some or all parameters are uncertain [12].

The general Job-Shop Scheduling Problem (JSSP) of size n x m can generally be described as a decision-making process concerning about the allocation of a limited set of $m=\{0, 1, ...,m\}$ resources over time to perform a set of $n=\{0, 1, ..., n\}$ tasks or jobs. Most real-world multi-operation scheduling problems can be described as dynamic and extended versions of the classic or basic Job-Shop scheduling combinatorial optimization problem. In a Job-Shop each job has a specified processing order through the machines, i.e. a job is composed of an ordered set of operations each of which is to be processed, for certain duration, on a machines. In the basic Job-Shop scheduling problem several constraints on jobs and machines are considered: machines are always available and never breakdown; there are no precedence constraints among operations of the different jobs; each machine can process only one job at a time; the processing of each operation cannot be interrupted; each job can be processed only on a machine at a time; setup times are independent of the schedules and are included in processing times; processing, release and due times of jobs are deterministic and known in advance.

These are the most common assumptions and restrictions in classic JSSP. In practice, many scheduling problems include further restrictions and relaxation of others [5]. Thus, for example, precedence constraints among operations of the different jobs are common because, most of the times, mainly in discrete manufacturing, products are made of several components that can be seen as different jobs whose manufacturing must be coordinated.

Therefore, in this work, we define a job as a manufacturing order for a final item, that could be **Simple** or **Complex**. It may be simple, like a part, requiring a set of operations to be processed. We call it a **Simple Product** or **Simple Final Item**. **Complex Final Items**, requiring processing of several operations on a number of parts followed by assembly operations at several stages, are also dealt with.

Moreover, in practice, scheduling environment tend to be dynamic, i.e. new jobs arrive at unpredictable intervals, machines breakdown, jobs are cancelled and due dates and processing times change frequently. This non-basic JSSP, focused in our work, which we call Extended Job-Shop Scheduling Problem (EJSSP), has major extensions and differences in relation to the classic or basic JSSP. The existence of operations on the same job, on different parts and components, processed simultaneously on different machines, followed by components assembly operations, which characterizes EJSSP, is not typical of scheduling problems addressed in the literature. However, such is common in practice. This approach to job definition, emphasizing the importance of considering complex jobs, which mimic customer orders of products, is in accordance with real world scheduling in manufacturing.

4 MH based Scheduling System

A Job-Shop like manufacturing system has a natural dynamic nature observed through several kinds of random occurrences and perturbations on working conditions and requirements over time. For this kind of environment it is important the ability to efficient and effectively adapt, on a continuous basis, existing schedules according to the referred disturbances, keeping performance levels. The application of Tabu Search to the resolution of this class of real world scheduling problems seems really promising. Although, most of the known work on scheduling deals with optimization of classic Job-Shop scheduling problems, on static and deterministic environments.

In the approach proposed here, a Meta-Heuristic is used, namely Tabu Search, using a strategy known as Resource Oriented Scheduling (each job is taken in turn for scheduling on a resource or machine at a time). The Dynamic Extended Job-Shop Scheduling Problem is decomposed into a series of deterministic

Single Machine Scheduling Problems (SMSP), which are solved one at a time, consecutively, by a Meta-Heuristic (Tabu Search). The obtained individual solutions are then integrated into the main problem. So, some mechanisms and algorithms were developed, jointly named MEM (Meta-Heuristics based Scheduling Method).
Based on the MEM method and on a simple architecture, a Scheduling System for Dynamic Non-Deterministic Problems (SEPDynNDET) is proposed. The purpose of SEPDynNDET is to extend the resolution of the deterministic problem to the non-deterministic one, in which changes may occur continually. This takes into account dynamic occurrences in a manufacturing system and adapts the current schedule.
The implemented approach on the SEPDynNDET system allows obtaining a predictive schedule, based on the MEM method, which is dynamically adapted through an integration procedure selected according to the occurred disturbances.

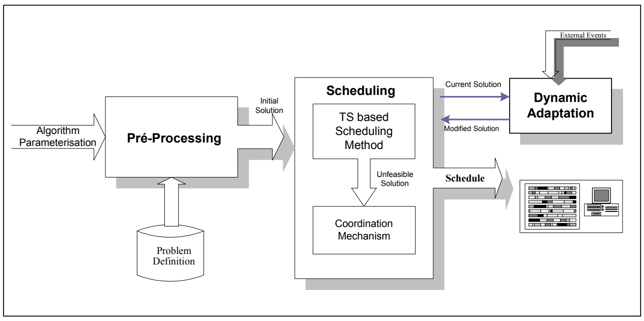

Figure 1 - SEPDynNDET Scheduling System

SEPDynNDET consists on a centralised TS-based scheduling system. One advantage of this is that the solutions (schedules) planned for single machines guarantees some consistency of manufacturing.
Some computational tests were carried out to evaluate the performance of the referred scheduling systems under different manufacturing scenarios. With a simple architecture and a small parameterization effort, it was possible to achieve good performance on the resolution of the Extended Job-Shop Scheduling Problem in a deterministic environment. For dynamic non-deterministic environments, the SEPDynNDET system behaves well, reacting efficiently and effectively, and adapting existing schedule according to disturbances and generating a new modified schedule [3].

5 Multi-Agent based Scheduling System

On section 4 we have described a centralized approach to solve real world scheduling problems. The TS based scheduling system developed adapts the resolution of the static and deterministic problem to the dynamic one in which changes may occur continually.
Considering that centralized scheduling is therefore large, complex, and difficult to maintain and reconfigure. On the other hand, the inherent nature of much industrial and service process is distributed we will try solving the complex dynamic scheduling problems in a distributed way using the Multi-Agent paradigm.
Modeling the Scheduling of Manufacturing Systems by means of two technologies like Meta-Heuristics and Multi-Agent Systems seems to be an interesting way to see Industrial Systems in the future.
In Multi-Agent System for Dynamic Manufacturing Scheduling using Meta-Heuristics (MADynScheMH) System we will model a Manufacturing Systems by means of a Multi-Agent Systems, where each agent may represent a processing entity (e.g. a machine). The system (Figure 2) has as objective to deal with the complex problem of Dynamic Scheduling in Manufacturing Systems. We want to prove that a good global solution for a scheduling problem may emerge from a community of machine agents solving locally their schedules and cooperating with other machine agents that share some relations between the operations/jobs (e.g. a precedence relation). Meta-Heuristics (Tabu Search or Genetic Algorithms) can be adapted to deal with dynamic problems, reusing and changing solutions/populations in accordance with the dynamism of the Manufacturing System.
The self-parameterization of the Meta-Heuristics allows a better adaptation to the situation being considered. The idea is that each agent adopt and provides the self-parameterization in accordance with the problem being solved (the method and parameters can change in run-time).
Meta-Heuristics (e.g. Tabu Search and Genetic Algorithms) are adequate for static problems, however real scheduling problems are quite dynamic (new orders arriving, being cancelled, machine delays or faults, etc.).
Coordination and Repair Mechanisms are used to guarantee the feasibility of schedules. Notice that solving locally problems and joining them will not guarantee the feasibility of schedules (e.g. precedence relations could not be guaranteed). The coordination mechanism will be established between machine agents involved in the execution of operations (jobs) with precedence relations in order to deal with the feasibility of the generated schedules in run-time. A repair mechanism will allow solving problems found in the feasibility of the schedules, when these problems are not solved by the coordination mechanism.

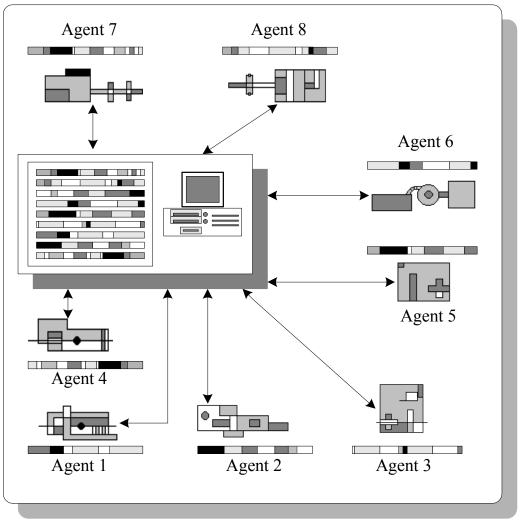

Figure 2 – MH Agents based Scheduling System

As objectives of the Multi-Agent System for Dynamic Manufacturing Scheduling using Meta-Heuristics (MADynScheMH) we identify the following assertions:

- Manufacturing Systems are well modeled by means of Multi-Agent Systems, where each agent may represent a processing entity in the Manufacturing System (e.g. a machine)
- A global solution for a scheduling problem may emerge from a community of machine agents solving locally their schedules and cooperating with other machine agents that shares some relations between the operations/jobs (e.g. a precedence relation)
- Meta-Heuristics can be adapted to deal with dynamic problems, reusing and adapting individuals/populations in accordance with the dynamism of the Manufacturing System
- The self parameterization of the Meta-Heuristics will allow a better adaptation to the situation being considered.

The main purpose is to create a Multi-Agent system where each agent represents a resource (machine) in a Manufacturing System (Machine Agents). Each machine agent is able to obtain schedules for manufacturing orders based on Meta-Heuristics (e.g. Genetic Algorithms or Tabu Search). Each machine agent is able to change the parameters of the basic algorithms according to the current situation or even to commutate from one algorithm to the other if the last start to be preferable. Meta-Heuristics algorithms used in the Machine Agents will be prepared to handle dynamism (new jobs arriving, cancelled jobs, changing jobs attributes) by adapting the solutions or population elements. A coordination mechanism will be established between machine agents involved in the execution of operations (jobs) with precedence relations in order to deal with the feasibility of the generated schedules.

In a real manufacturing system a product is produced, step by step, passing in several machines, in each machine it will be performed one operation (job) of the process plan.

In this approach we have one agent for each machine (using Meta-Heuristics). However, if we join solutions obtained by our machine agents we will observe that, possibly, they will not be feasible. In fact, if operation op1 (in machine m1) precedes operation *Op2* (in machine *m2*) and O*p2* precedes O*p3* (in machine *m3*) in a manufacturing process, it is not guaranteed that the initial time for *Op2* in *m2* will be after the end of *Op1* in *m1* nor that the end of O*p2* in *m2* will be before the start of *Op3* in *m3*.

Two possible approaches, to deal with this problem, could be used. In the first the system wait for the solutions obtained by the machine agents and then apply a repair mechanism to shift some operations in the generated schedules till a feasible solution is obtained (**Repair Approach**). In the second a coordination mechanism is established between related agents in the process. In this situation two machine agents need to cooperate if a precedence relation exists between two operations (jobs) scheduled for these machines (**Cooperative Approach)**.

The second approach seems to be more interesting, since in the first there is a trend to have several non-occupied time intervals in the schedules. In order to use the second approach we will penalize solutions in which feasibility is not guaranteed. We consider that in this way feasible solutions will emerge.

These coordination mechanisms are prepared to accept agents subjected to dynamism (new jobs arriving, cancelled jobs, changing jobs attributes).

6 Concluding Remarks

This paper has discussed an important gap between static and dynamic scheduling. Dynamic scheduling maintains schedules in a dynamic environment where a variety of unexpected events can occur at any time.

A simple and general framework exploring the potential of Meta-Heuristics, which is of practical utility, embedded in a simple framework to solve difficult problems in dynamic environments. The proposed approach is in line with reality and away from the approaches that deal with static and classic or basic Job-Shop scheduling problems. In fact, in real world, where problems are essentially of dynamic and stochastic nature, the traditional methods or algorithms are of very little use. This is the case with most algorithms for solving the so-called static scheduling problem for different setting of both single and multi-machine systems arrangements. This reality, motivated us to concentrate on tools, which could deal with such dynamic, disturbed scheduling problems, both for single and multi-machine manufacturing settings, even though, due to the complexity of these problems, optimal solutions may not be possible to find.

Considering that centralized scheduling is therefore large, complex, and difficult to maintain and reconfigure. On the other hand, the inherent nature of much industrial and service process is distributed we try to solve the complex dynamic scheduling problems in a distributed way using the Multi-Agent paradigm.

Modeling the Scheduling of Manufacturing Systems by means of two technologies like Meta-Heuristics and Multi-Agent Systems could be an interesting way to see Industrial Systems in the future.

In MADynScheMH system we model a Manufacturing Systems by means of a Multi-Agent Systems, where each agent may represent a processing entity (resource). We consider that a good global solution for a scheduling problem may emerge from a community of machine agents solving locally their schedules and cooperating with other machine agents that shares some relations between the operations/jobs.

Observing Industrial Manufacturing Systems like evolution-based social systems is very important in order to allow a better understanding and integration between the machinery and the human being.

References:

[1] Ana Madureira, Carlos Ramos, Silvio C. Silva, Toward Dynamic Scheduling Through Evolutionary Computing, *WSEAS Transactions on Systems*, Issue 4, Volume 3, ISSN 1109-2777, pp. 1596-1604, 2004.

[2] Ana M. Madureira, Carlos Ramos and Sílvio do Carmo Silva, A Genetic Algorithm for The Dynamic Single Machine Scheduling Problem, In *4th IEEE/IFIP Intl. Conf. on Information Technology for Balanced Automation Systems in Production and Transportation*,(Germany), pp. 315-323, 2000.

[3] J. Blazewicz, K. H. Ecker, E. Pesch, G. Smith and J. Weglarz, *Scheduling Computer and Manufacturing Processes*, Springer, 2nd edition, New York, 2001.

[4] Joseph Adams, Egon Balas and Daniel Zawack, The Shifting Bottleneck Procedure for Job Shop Scheduling, *Management Science*, Vol. 34, nº 3, USA, 1988.

[5] M. C. Portmann, *Scheduling Methodology: optimization and compu-search approaches, in the planning and scheduling of production systems*, Chapman &Hall, 1997.

[6] M. Kollingbaum, T. Heikkilä, P. Peeters, J. Matson, P. Valckenaers,D. McFarlane and G. Bluemink, Emergent flow shop control based on MASCADA agents, *IFAC Symposium On Manufacturing, Modeling, Management And Control (MIM2000)*, Greece, 2000.

[7] M. Pinedo, *Scheduling – Theory, Algorithms and Systems*, 2nd edition, Prentice-Hall, 2002.

[8] P. Cowling and M.Johansson, Real time information for effective dynamic scheduling, *European Journal of Operational Researc*h,139 (2), pp. 230-244, 2002.

[9] Paulo Sousa, C. Ramos and J. Neves, The Fabricare Scheduling Prototype Suite: Agent interaction and knowledge base, *Journal of Intelligent Manufacturing*, 14(5), Kluwer Academic Publishers, pp.441-455, 2003.

[10] Peter Brucker, Scheduling Algorithms, Springer, 3rd edition, New York, 2001.

[11] Report from EVONET Working Group on Evolutionary Approaches to Timetabling and Scheduling, *The state of the art in evolutionary Approaches to timetabling and scheduling*, 1999.

[12] S. French, *Sequencing and Scheduling: An introduction to the Mathematics of the Job Shop*, Ellis Horwood, Chichester, 1982.

[13] S. Jain and S. Meeran, Deterministic Job Shop scheduling: past, present and future, *European Journal of Operational Research*, nº113, 390-434, 1999.

[14] W. Shen and D. Norrie, Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey, *Knowledge and Information Systems*, 1(2), 129-156, 1999.

[15] Thomas E. Vollmann, William Berry, and D. Clay Whybark, *Manufacturing Planning and Control Systems*, McGraw-Hill, New York, 1997.