EVOLUTIONARY APPROACH FOR DYNAMIC SCHEDULING IN MANUFACTURING

Ana Madureira, Carlos Ramos
Instituto Superior de Engenharia do Porto, GECAD & Computer Science Department, Porto, Portugal

Sílvio do Carmo Silva
Universidade do Minho, Dept. de Produção e Sistemas, Braga, Portugal

Keywords: Dynamic scheduling, Extended Job-Shop Scheduling, Manufacturing, Genetic Algorithms

Abstract: This paper presents a simple and general framework exploring the potential of evolutionary algorithms, which is of practical utility, embedded in a simple framework to solve difficult problems in dynamic environments. The proposed evolutionary 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. We decided to address the problem drawing upon the potential of Genetic Algorithms to deal with such complex situations.

1 INTRODUCTION

Evolutionary algorithms have often been shown to be effective for difficult combinatorial optimisation problems appearing in various industrial, economical, and scientific domains. Prominent examples of such problems are scheduling, timetabling, network design, transportation and distribution problems, vehicle routing, travelling salesperson, other graph problems, satisfiability, packing problems, planning problems, and general mixed integer programming.

Considering that natural evolution is a process of continuous adaptation, it seemed us appropriate to consider Genetic Algorithms for tackling real Non-Deterministic Scheduling Problems. Thus, the GA based scheduling system developed adapts the resolution of the static and deterministic problem to the dynamic one in which changes may occur continually. A population regenerating mechanism is put forward, for adapting the population of solutions, according to disturbances, to a new population, which increases or decreases according to new job arrivals or cancellations.

It is widely accepted that manufacturing scheduling problems are generally difficult or hard to solve to optimality (Baker, 1974), (French, 1982) and (Blazewick, 2001). Most of the approximation methods proposed for the Job-Shop Scheduling Problems are oriented methods, i.e. developed specifically for the problem in consideration. Some examples of this class of methods are the priority rules (Baker, 1974), (French, 1982) (Blazewick, 2001) and the Shifting Bottleneck proposed in (Adams, 1988).

The problem of finding good solutions to scheduling problems is very important to real manufacturing systems because the production rate and production costs are very dependent on the schedules used for controlling the flow of work through the system. Most research in scheduling focuses on optimisation of static and dynamic **deterministic** problems where all problem data are known before scheduling starts. However many real world optimisation problems are **non-deterministic**, in which changes may occur continually.

Madureira A., Ramos C. and do Carmo Silva S. (2004).
EVOLUTIONARY APPROACH FOR DYNAMIC SCHEDULING IN MANUFACTURING.
In *Proceedings of the First International Conference on Informatics in Control, Automation and Robotics*, pages 288-291
DOI: 10.5220/0001145002880291

Due to their dynamic nature, real scheduling problems have an additional complexity in relation to static ones. In many situations these problems, even for apparently simple situations, are hard to solve, i.e. the time required to compute an optimal solution increases exponentially with the size of the problem (Morton and Pentico, 1993).
Recently the scheduling problem in dynamic environments have been investigated by a number of authors especially in the evolutionary community, see for example, (Jain, 1999), (Dimopoulos et al., 2000) and (Madureira, 2002).
The proposed approach deals with these two cases of dynamic scheduling: deterministic and stochastic. For such class of problems, the goal is no longer to find a single optimum, but rather to continuously adapt the solution to the changing environment. The purpose of this paper is to describe an approach based on GA for solving dynamic scheduling problems.
The paper is structured as follows: section 2 provides a description of the scheduling problem to be solved. In section 3, we describe a method based on GA for the resolution of the Extended Job-Shop Scheduling Problems (EJSSP) and describe an approach to solve the Non-deterministic version of the Extended Job-Shop Scheduling Problems. Finally, some conclusions are drawn and some ideas for future work are presented.

2 PROBLEM DEFINITION

Most real-world multi-operation scheduling problems can be described as dynamic and restricted or relaxed versions of the classic Job-Shop scheduling combinatorial optimisation problem (Morton and Pentico, 1993). In a Job-Shop problem 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, during a given time, i.e. the operation processing time, in a machine of the system. In the classic Job-Shop Scheduling Problems (JSSP) several constraints on jobs and machines are considered: machines are always available and never break down; there are no precedence constraints among operations of the different jobs; the operations processing can not be interrupted and each machine can process only one job at a time; 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; technological constraints are deterministic and known in advance.
In practice, many scheduling problems include further constraints. This means that problems can become more complex and more general, i.e. non-basic (Portmann, 1997). 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. This non-basic JSSP, focused in our work, which we called **Extended Job-Shop Scheduling Problem** (EJSSP), has major extensions and differences in relation to the classic or basic JSSP (Madureira et al., 2002).

3 THE PROPOSED APPROACH

The scheduling approach presented in this section is applicable to both static and dynamic manufacturing environments for any optimizing criteria that are possible to establish. It is based on the decomposition of the problems into Single Machine Scheduling Problems(SMSP), one for each machine involved in processing, ant later integration for obtaining a solution to the original problem, i.e. to the Extended JSSP.
This work is concerned with the solution of problem instances of the EJSSP. It starts focusing on the solution of the dynamic deterministic JSSP problems. For solving these we developed a framework, leading to a dynamic scheduling system having as a fundamental scheduling tool a GA-based scheduling method with two main pieces of intelligence. One such piece is a GA-based method for deterministic scheduling. This includes a Genetic Algorithm for single machine problems and an inter-machine activity coordination mechanism that attempts to ensure a good feasible solution for the deterministic EJSSP. The other piece is a rescheduling mechanism that includes a method for population regeneration under dynamic environments, increasing or decreasing it according new job arrivals or cancellations.
This approach to solve the dynamic EJSSP consists on generating a predictive schedule in advance using the information available. When disruptions occur in the system during the execution, the predictive schedule is modified or revised in order to consider the recent modifications.

3.1 Dynamic Deterministic EJSSP

Initially, we start by decomposing the deterministic EJSSP problem into a series of deterministic Single Machine Scheduling Problems. We assume the existence of different and known job release times r_j, prior to which no processing of the job can be done

and, also, job due dates d_j. Based on these, release dates and due dates are determined for each SMSP and, subsequently, each such problem is solved independently by the Genetic Algorithm.
Afterwards, the solutions obtained for each SMSP are integrated to obtain a solution to the main problem instance, i.e. the instance of the EJSSP. The integration of the SMSP solutions may give an unfeasible schedule to the EJSSP. This is why schedule repairing may be necessary to obtain a feasible solution. The repairing mechanism named Inter-Machine Activity Coordination Mechanism (IMACM) carries this out (Madureira et al., 2002). The repairing is carried out through coordination of machines activity, having into account job operation precedence and other problem constraints. This is done keeping job allocation order, in each machine, unchanged. The IMACM mechanism establishes the starting and the completion times for each operation. It ensures that the starting time for each operation is the highest of the two following values: the completion time of the immediately precedent operation in the job, if there is only one, or the highest of all if there are more, and the completion time of the immediately precedent operation on the machine.

3.2 Dynamic Non-Deterministic EJSSP

For non-deterministic problems some or all parameters are uncertain, i.e. are not fixed as we assumed in the deterministic problem. Non-determinism of variables has to be taken into account in real world problems. For generating acceptable solutions in such circumstances our approach starts by generating a predictive schedule, using the available information and then, if perturbations occur in the system during execution, the schedule may have to be modified or revised accordingly, i.e. rescheduling is performed. Therefore, in this process, an important decision must be taken, namely that of deciding if and when rescheduling should happen. The decision strategies for rescheduling may be grouped into three categories: continuous, periodic and hybrid rescheduling. In the continuous one rescheduling is done whenever an event modifying the state of the system occurs. In periodic rescheduling, the current schedule is modified at regular time intervals, taking into account the schedule perturbations that have occurred. Finally, for the hybrid rescheduling the current schedule is modified at regular time intervals if some perturbation occurs.
In the scheduling system for Extended JSSP, implementing our approach, rescheduling is necessary due to two classes of events (Madureira et al., 2001a):
a) **Partial events** imply changes in jobs or operations attributes such as processing times, due dates and release times.
b) **Total events**, imply changes in neighbourhood structure, resulting from either new job arrivals or job cancellations.
While, on one hand, partial events only require redefining job attributes and re-evaluation of the objective function of solutions, total events, on the other hand, require a change on solution structure and size, carried out by inserting or deleting operations, and also re-evaluation of the objective function. Therefore, under a total event, the modification of the current solution is imperative. In this work, this is carried out by mechanisms described in (Madureira et al., 2001b) for SMSP.
Considering the processing times involved and the high frequency of perturbations, rescheduling all jobs from the beginning should be avoided. However, if work has not yet started and time is available, then an obvious and simple approach to rescheduling would be to restart the scheduling from scratch with a new modified solution on which takes into account the perturbation, for example a new job arrival. When there is not enough time to reschedule from scratch or job processing has already started, a strategy must be used which adapts the current schedule having in consideration the kind of perturbation occurred.
The GA-based scheduling system, here proposed, for non-deterministic EJSSP is structured, in three modules, namely the modules for pre-processing, scheduling and dynamic schedule adaptation:
1. Pre-Processing Module - The pre-processing module deals with processing input information, namely problem definition and instantiation of algorithm components and parameters, such as, the initial individual and population generation mechanisms, size of population, genetic operators and respective probabilities.
2. Scheduling Module - The scheduling module is concerned with the application of the GA-based scheduling method for deterministic EJSSP presented above, considering that all release dates, processing times and due dates are known in advance. Whenever a new event occurs which disturbs the schedule, in such a way that rescheduling is to be done, the dynamic adaptation module generates a new deterministic problem. Then, the scheduling module solves this new problem.
3. Dynamic Adaptation Module - The occurrence of a partial event requires redefining job attributes and a re-evaluation of the schedule objective function. A change in job due date requires the re-

calculation of the operation starting and completion due times of all respective operations. However, changes in the operation processing times only requires re-calculation of the operation starting and completion due times of the succeeding operations. A new job arrival requires definition of the correspondent operation starting and completion times and a regenerating mechanism to integrate all operations on the respective single machine problems. In the presence of a job cancellation, the application of a regenerating mechanism eliminates the job operations from the SMSP where they appear. After the insertion or deletion of positions, neighbourhood regeneration is done by updating the size of the neighbourhood and ensuring a structure identical to the existing one. Then the scheduling module can apply the search process for better solutions with the new modified solution.

4 CONCLUDING REMARKS

In most practical environments, scheduling is an ongoing reactive process where the presence of real time information continually forces reconsideration and revision of pre-established schedules.

The handling of uncertainty is an important issue in real-world scheduling problems. Uncertainty can arise through incomplete knowledge of the problem, through incomplete knowledge of how the problem will change over time or may be inherent to the problem itself.

This work is concerned with the resolution of realistic Job-Shop Scheduling Problems (EJSSP). Thus, not only it focuses on the solution of the dynamic EJSSP problem but also on both the deterministic and non-deterministic versions of it. Moreover, it is concerned with integrated scheduling of jobs which are products composed by several parts or components which may be submitted to a number of manufacturing and multi level assembly operations, having as the main criterion meeting due dates. We call these problems Extended EJSSP.

Considering that natural evolution is a process of continuous adaptation, it seemed us appropriate to consider Genetic Algorithms for tackling real Non-Deterministic Scheduling Problems. Thus, the GA based scheduling system developed adapts the resolution of the static and deterministic problem to the dynamic one in which changes may occur continually. A population regenerating mechanism is put forward, for adapting the population of solutions, according to disturbances, to a new population, which increases or decreases according to new job arrivals or cancellations.

We recognize the need for further testing, particularly for better evaluation of the suitability the proposed framework and mechanisms under dynamic Extended Job-Shop environments. We also recognize that this is not an easy task because it is difficult to find in the literature test problems with the job structure that we selected and think important, in industrial practice, namely jobs made from several parts to be manufactured and assembled through several assembly operations and stages.

REFERENCES

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

Baker, K.R., 1974, *Introduction to sequencing and scheduling*, Wiley, New York.

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

Dimopoulos, C. and Zalzala, Ali M. S., 2000, Recent Developments in Evolutionary Computation for Manufacturing Optimization: Problems, Solutions and Comparisons, *IEEE Trans. on Evol. Computation*, nº4, vol.2, 93-113.

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

Madureira, Ana M., Ramos, Carlos and Silva, Sílvio do Carmo, (2001a), A Genetic Approach for Dynamic Job-Shop Scheduling Problems, *4th MetaHeuristics International Conference (MIC'2001)*, Porto (Portugal).

Madureira, Ana M., Ramos, Carlos and Silva, Sílvio do Carmo, (2001b), A GA Based Scheduling System for The Dynamic Single Machine Scheduling Problem, *ISATP'2001 (IEEE International Symposium Assembly and Task Planning)*, Fukuoka (Japão).

Madureira, Ana M., Ramos, Carlos and Silva, Sílvio do Carmo, (2002), A Coordination Mechanism for Real World Scheduling Problems Using Genetic Algorithms, *IEEE 2002 IEEE World Congress on Computational Intelligence*, Honolulu - Hawaii (EUA).

Morton, E. Thomas and Pentico, David W., 1993, *Heuristic Scheduling Systems*, John Wiley & Sons.

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