**A Modified Ant Colony Algorithm to the P|Prec|C_max Scheduling Problem: A Comparative Study**

Mohamed Messaoudi-Ouchene, LAMDA-RO Laboratory, Mathematics Department, Faculty of Science, University Saad Dahlab of Blida, Blida, Algeria

Ali Derbala, LAMDA-RO Laboratory, Mathematics Department, Faculty of Science, University Saad Dahlab of Blida, Blida, Algeria

**ABSTRACT**

This paper investigates a comparative study which addresses the P/prec/C_max scheduling problem, a notable NP-hard benchmark. MLP_SACS, a modified ant colony algorithm, is used to solve it. Its application provides us a better job allocation to machines. In front of each machine, the jobs are performed with three priority rules, the longest path (LP), a modified longest path (MLP) and a maximum between two values (MAX). With these three rules and with both static and dynamic information heuristics called “visibility”, six versions of this ant colony algorithm are obtained, studied and compared. The comparative study analyzes the following four meta-heuristics, simulated annealing, taboo search, genetic algorithm and MLP_SACS (a modified ant colony system), is performed. The solutions obtained by the MLP_SACS algorithm are shown to be the best.

**Keywords:** Ant Colony Optimization, Meta-heuristic, Parallel scheduling, Priority List, Visibility

**1. INTRODUCTION**

Scheduling is a decision-making process that is used on a regular basis in many manufacturing and services industries. It deals with the allocation of resources to jobs over given time periods and its goal is to optimize one or more objectives. The first scheduling algorithms were formulated in the mid fifties. Since then, there has been a growing interest in scheduling. During the seventies, computer scientists discovered scheduling as a tool for improving the performance of computer systems. Scheduling problems have been investigated and classified with respect to their computational complexity. During the last few years, new and interesting scheduling problems have been formulated.

The resources and jobs in an organization can take many different forms. The resources may be machines in a workshop, runways at an
airport, processing units in a computing environment, and so on. The jobs may be operations in a production process, take-offs and landings at an airport, stages in a construction project, executions of computer programs, and so on.

Most parts of the Bruckers’s book (Brucker, 2007) are devoted to the discussion of polynomial algorithms. In addition, enumerative procedures based on branch & bound concepts and dynamic programming, as well as local search algorithms are presented. Methods of combinatorial optimization that are relevant for the solution procedures and computational complexity theory are stated. Each job may have a certain priority level, an earliest possible starting time and a due date. One objective may be the minimization of the completion time of the last job and another may be the minimization of the number of jobs completed after their respective due dates (Pinedo, 2008).

The scheduling problem, on identical parallel machines, to precedence constraints denoted $\text{P|prec|C}_{\text{max}}$ is often stated to be a NP-hard problem (Brucker & Knust, 1998). Such scheduling is done in two phases: the allocation of jobs to machines and the sequencing of jobs on each machine. An exact solution is only effective if the number of jobs is reduced. Our goal is to minimize the scheduling length or the completion time of the last job. This criterion is important because the scheduler ensures load balancing between machines. We consider only the non-preemptive scheduling. Once a job begins execution, it can not be interrupted. The time required to execute a given job is known in advance and does not depend on the machine used. Precedence relations may exist between jobs and are given by a graph. In scheduling on parallel machines, the most commonly used algorithms are lists of types. They determine, to a range of jobs given by a list, the corresponding scheduling. They consider the jobs one by one and make the decision to schedule based on a partial ordering of already sequenced jobs. For many difficult problems in a wide variety of areas, meta-heuristics have received considerable interest and have proven effective in various fields such as industrial, economic, logistical, engineering, trading, science, etc.. Phelipeau-Gelineau (Gelineau, 1996) developed a taboo search algorithm to solve the $\text{P|prec|C}_{\text{max}}$ problem. Alba (2005) brings both the parallelism and metaheuristic fields with a main focus on optimization. It contains a methodological part dealing with a specified technique, discusses how parallel models can be derived for this technique to become more efficient. Some experimental analysis is included. Complex optimization problems abound in the real world. In the face of these challenges, established methods often fall short of providing good solutions. However, ‘exact’ and ‘heuristic’ techniques are dramatically enhancing our ability to solve significant practical problems in the world of optimization.

They are changing the landscape in the field, broadening the frontiers of the possible, and allowing us to engage effectively with a whole new range of challenges. This monograph (Martí & Reil, 2011) sets out state-of-the-art optimization methods. The exact and heuristic techniques outlined in these pages can be put to use in any number of combinatorial optimization problems. It is also a tutorial on how to design effective and successful implementations of exact and heuristic procedures alike. Examples of new meta-heuristics are evolutionary algorithms and ant colonies (Dorigo, Maniezzo, & Colorni, 1991; Dorigo & Caro, 1999). Ant Colony Optimization (ACO) has been successfully applied to a wide range of combinatorial optimization problems. Solnon (2010) describes ACO and studies its efficiency for solving hard combinatorial problems, with a specific focus on constraint programming. Beyond the ant metaphor, it describes the mechanisms which allow artificial ants to converge towards solutions and, more particularly, those used to balance diversification and intensification. The main goal of Talbi’s book (Talbi, 2013) is to provide a state of the art of hybrid meta-heuristics. It provides a complete background that enables readers to design, implement and solve complex optimization problems under uncertainty. Readers learn to solve large scale problems quickly and efficiently with complementary mathematical programming, constraint
Related Content

Multi-Objective Optimal Power Flow using BAT Search Algorithm with Unified Power Flow Controller for Minimization of Real Power Losses

A Local Search Approach to Solve a Financial Portfolio Design Problem
[www.igi-global.com/article/a-local-search-approach-to-solve-a-financial-portfolio-design-problem/125863?camid=4v1a](www.igi-global.com/article/a-local-search-approach-to-solve-a-financial-portfolio-design-problem/125863?camid=4v1a)

Reinforcement Learning for Improving Gene Identification Accuracy by Combination of Gene-Finding Programs
[www.igi-global.com/article/reinforcement-learning-improving-gene-identification/64643?camid=4v1a](www.igi-global.com/article/reinforcement-learning-improving-gene-identification/64643?camid=4v1a)
(Approximate) Frequent Item Set Mining Made Simple with a Split and Merge Algorithm
www.igi-global.com/chapter/approximate-frequent-item-set-mining/38572?camid=4v1a