Genetic Algorithms for a Discrete Lot Sizing Problem

Pr Norbert HOUNKONNOU
Professor of Mathematics
Department of Mechanical Engineering
University of Abomey-Calavi

Dr Ing. Ratheil HOUNDJI *
Ing., MSc, Ph.D. Associate Professor
Department of Software Engineering
University of Abomey-Calavi

Tafsir GNA
Research Assistant

Lot sizing takes an important place in production planning in industry. It consists in determining a production plan that meets the orders and at the same time takes into account the financial objectives of the enterprise. Recent researches have experimented an NP-Hard variant of lot sizing problem: the Pigment Sequencing Problem (PSP). Several methods have been applied to PSP. None of the applied methods is based on genetic algorithms whereas they showed their efficiency in solving optimization problems. In this document, we apply two solving methods based on genetic algorithms to PSP which are the hierarchical and parallel genetic algorithms. The experiments allow us to compare the results obtained in applying these last solving methods to the ones obtained of the application of other methods made in ealier researches. These very first results show that genetic algorithms could be efficient in solving PSP.
Key-words*: Genetic algorithm, production planning, pigment sequencing problem, lot sizing.*

1 Introduction

Lot sizing problem consists in identifying items to produce, when to produce and on which machine in order to meet the orders while taking into account financial goals. Such a problem has been studied these recent decades. In fact, solving a lot sizing problem has a lot of challenges. Not only several types of items are required to be produced but the production planning has to meet often opposite goals such as serving customer needs and minimizing production and stocking costs.
Several versions of lot sizing problems have been proposed in the literature. Lately, Houndji et al. [18] et Ceschia et al. [6] have worked a NP-Hard variant known as *Pigment Sequencing Problem* (Pochet et Wolsey [28]) and included in the CSPlib library (Gent and Walsh, [11]). It consists in producing several items on a single machine whose production capacity is restricted to one item per period. The planning horizon is discrete and finite with stocking costs and setup costs from one item to another.
Pigment Sequencing problem, like any lot sizing problem can be formalized and solved with genetic algorithms. Genetic algorithms are heuristic search methods inspired by the natural evolution of living species. Based upon the concept of the survival of the fittest, genetic algorithms are able over multiple generations to find the best solution to a problem. Several researches [14] [26] have showed how effecient they can be in solving optimization problems.
In this paper, we expose a search method based on genetic algorithms. This method known as *Hierarchical Coarsed-grained and Master-slave Parallel Genetic Algorithms (HCM-PGAs)* divide the global population into small set of population [...]. We use and experiment this approach and the results obtained show that genetic algorithms are a promising method in solving a discrete lot sizing problem such as Pigment Sequencing problem.
This paper is organized as follows: Section 2 expose some background on the Pigment Sequencing problem, Section 3 gives details on our method based on genetic algorithms, Section 4 presents some experimental results obtained off the application of our method and Section 5 concludes and provides some perspectives.

2 Pigment Sequencing problem (PSP)

In this section, we present the Pigment sequencing problem and give a formal description of the problem.

2.1 Literature review

PSP belongs to the category of Discrete Lot Sizing Problems (DLSP). PSP is a problem in which all capacity avail-

*Address all correspondence related to ASME style format and figures to this author.

able for a period is used to produce one item.
Miller and Wolsey [25] formulated the DLSP with setup costs not dependent of sequence as a network flow problem. They exposed MIP formulations for various modifications (with backlogging, with safety stock, with initial stock). In addition, several more MIP formulations and variants have been proposed and discussed by Pochet and Wolsey [28].
Gicquel and al. [13] exposed a formulation and derived valid inequalities for the DLSP with several items and sequential setup costs and periods, which is a modification of the problem proposed by Wolsey [33]. Furthermore, Gicquel and al. [12] proposed a new approach to modelisation of the DLSP with several items and sequential setup costs and periods that take into account relevant physical attribute such as color, dimension and level of quality. This allowed them to effectively reduced th number of variables and constraints in the MIP Models. Houndji and al. [18] introduced a new global constraint that they named *stocking cost* in order to effectively solve the PSP with constraint programming. They tested it on new instances and published them on CSPlib (Gent and Walsh [11]). The experimental results showed that *stocking cost* is effective in filtering compared to other constraints largely used in the community of constraint programming.
Lately, Ceschia and al. [7] applied the simulated annealing to the PSP. They introduced an approach that guide the local search and applied it to new instances available on Opthub library [6].

2.2 Description

Several studies [16] [7] have already been conducted on PSP. It can be described as a problem which consists in finding a production planning of various item on one machine with setup costs. Setup costs are costs necessary for the transition from on item production i to the one of item j so that $i \neq j$. The production planning needs to meet the customer orders while:

- not exceeding the production capacity of the machine.
- minimizing the setup and stocking costs.

It is assumed that the production period is short enough to produce only one item per period and all orders are normalized i.e. The machine's production capacity is restricted to one item per period and $d(i,t) \in \{0,1\}$ with i the item and t the period. It is a production planning problem with the following specifications: a discrete and finite planning horizon, some capacity constraints, a deterministic and static order, several items and small bucket, setup costs, only one level, without shortage.

Instance: Be a problem with data as follows:

- Number of items: $NI = 2$;
- Number of periods: $NT = 5$;
Order per period. Be d(i, t) the order of item i in the period t: $d(1,t) = (0,1,0,0,1)$ and $d(2,t) = (1,0,0,0,1)$;
- Stocking cost. Be *h(i)* the stocking cost of the item i, $h(1) = h(2) = 2$

Be xT the production planning which represents a potential solution to the problem. It is a table of size NT. A possible solution to the problem is $xT = (2,1,2,0,1)$ with a cost of $q(2,1)+q(1,2)+q(2,1)+2h(2) = 15$. The optimal solution is : $xT = (2,1,0,1,2)$ with a cost of $q(2,1)+q(1,2)+h(1) = 10$.

3 Our Method

In this section, we explain what genetic algorithms are, how they work in pratice and expose our method based on this category of algorithms in solving a Discrete Lot Sizing Problem.

3.1 Genetic Algorithms

Genetic algorithms are algorithms designed to mimic the natural eveolution of living species and reproduction mecanisms. They have been proposed for the first time by John Holland [15] in 1970. One of the main principles of these algorithms is the concept of the "*survival of the fittest*" whiche states that one individu whose features fit the best with the environement is more likely to survive.

4 Experimental results [1]

In this section, we firstly present the tools used for the implementation and tests, then the instances on which we applied our approach of hierarchical genetic algorithms, the hyperparameters we defined for the aforementioned tests and finally we expose the experimental results obtained from the tests.

4.1 Implementation and tools

Our approach is implemented using the python programming language and specifically the version 3.5. Python is well suited for this kind of implementation thanks to the vast amount of packages available for handling such data.
We implement the tests on a computer with the following specifications:

- Operating system : Linux Ubuntu 20.04 LTS
- Processor : Intel® Core TM i7 CPU L 640 @ 2.13GHz * 4 ;
- Memory : 3,7 Gio
- Type of the operating system : 64 bits

5 Conclusions and perspectives

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse

[1] Examine the input file, asme2ej.tex, to see how a footnote is given in a head.

Table 1. Figure and table captions do not end with a period

Example	Time	Cost
1	12.5	$1,000
2	24	$2,000

cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

$$f(t) = \int_{0_+}^{t} F(t)dt + \frac{dg(t)}{dt} \tag{1}$$

6 Figures

7 Tables

All tables should be numbered consecutively and centered above the table as shown in Table 1. The body of the table should be no smaller than 7 pt. There should be a minimum two line spaces between tables and text.

8 Citing References

The ASME reference format is defined in the authors kit provided by the ASME. The format is:

> *Text Citation*. Within the text, references should be cited in numerical order according to their order of appearance. The numbered reference citation should be enclosed in brackets.

The references must appear in the paper in the order that they were cited. In addition, multiple citations (3 or more in the same brackets) must appear as a " [1-3]". A complete definition of the ASME reference format can be found in the ASME manual [**?**].

The bibliography style required by the ASME is unsorted with entries appearing in the order in which the citations appear. If that were the only specification, the standard BIBTEX unsrt bibliography style could be used. Unfortunately, the bibliography style required by the ASME has additional requirements (last name followed by first name, periodical volume in boldface, periodical number inside parentheses, etc.) that are not part of the unsrt style. Therefore, to get ASME bibliography formatting, you must use the `asmems4.bst` bibliography style file with BIBTEX. This file is not part of the standard BibTeX distribution so you'll need to place the file someplace where LaTeX can find it (one possibility is in the same location as the file being typeset).

With LATEX/BIBTEX, LATEX uses the citation format set by the class file and writes the citation information into the .aux file associated with the LATEX source. BIBTEX reads the .aux file and matches the citations to the entries in the bibliographic data base file specified in the LATEX source file by the `\bibliography` command. BIBTEX then writes the bibliography in accordance with the rules in the bibliography .bst style file to a .bbl file which LATEX merges with the source text. A good description of the use of BIBTEX can be found in [**?**, **?**] (see how two references are handled?). The following is an example of how three or more references [?, ?, ?] show up using the `asmems4.bst` bibliography style file in conjunction with the `asme2ej.cls` class file. Here are some more [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**] which can be used to describe almost any sort of reference.

9 Conclusions

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

10 Discussions

This template is not yet ASME journal paper format compliant at this point. More specifically, the following features are not ASME format compliant.

1. The format for the title, author, and abstract in the cover page.
2. The font for title should be 24 pt Helvetica bold.

If you can help to fix these problems, please send us an updated template. If you know there is any other non-compliant item, please let us know. We will add it to the above list. With your help, we shall make this template compliant to the ASME journal paper format.

Acknowledgements

ASME Technical Publications provided the format specifications for the Journal of Mechanical Design, though they are not easy to reproduce. It is their commitment to ensuring quality figures in every issue of JMD that motivates this effort to have authors review the presentation of their figures.

Thanks go to D. E. Knuth and L. Lamport for developing the wonderful word processing software packages TEX and LATEX. We would like to thank Ken Sprott, Kirk van Katwyk, and Matt Campbell for fixing bugs in the ASME style file `asme2ej.cls`, and Geoff Shiflett for creating ASME bibliography stype file `asmems4.bst`.

Appendix A: Head of First Appendix

Avoid Appendices if possible.

Appendix B: Head of Second Appendix

Subsection head in appendix

The equation counter is not reset in an appendix and the numbers will follow one continual sequence from the beginning of the article to the very end as shown in the following example.

$$a = b + c. \tag{2}$$