

COPSolver: open source software for solving combinatorial optimization and other decision problems - library for solving the multi-product batch processing time maximization problem

Tatiana Balbi Fraga, Federal University of Pernambuco, Avenida Marielle Franco, Bairro Nova Caruaru, Caruaru, 55014-900, PE, Brazil, tatiana.balbi@ufpe.br

Abstract

This paper presents the software COPSolver, originally developed to solve decision and/or optimization problems, especially from the combinatory optimization area. The first version of COPSolver has a application of Fraga's exact method for solving the multi-product batch processing time maximization problem. This paper address the COPSolver_1.0-1 and the next steps that will be implemented for the software's applications development.

combinatorial optimization problem; exact method; multi-product batch problem; processing time maximization.

Code metadata

Nr.	Code metadata description	Please fill in this column
C1	Current code version	COPSolver_1.0-1
C2	Permanent link to code/repository used for this code version	https://github.com/tbfraga/COPSolver/releases/tag/v1.0-1
C3	Permanent link to Reproducible Capsule	https://codeocean.com/capsule/4837209
C4	Legal Code License	Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International Public License
C5	Code versioning system used	git
C6	Software code languages, tools, and services used	c++, CodeBlocks, LINGO, Ubuntu 22.04.1, GitHub
C7	Compilation requirements, operating environments & dependencies	elementary OS (can be adapted to Ubuntu)
C8	If available Link to developer documentation/manual	http://tbfraga.github.io/COPSolver/documentation/
C9	Support email for questions	tbfraga@proton.me

1. Motivation and significance

A decision problem arises when we need to choose an option from a set of known options so that the chosen decision meets a set of pre-established constraints. This problem is configured as an optimization problem when the decision taken must, in addition to meeting the constraints, be the one that best contributes to the achievement of a given group of objectives.

Decision and/or optimization problems are frequent in industrial and organizational environments, as well as in our daily lives. Usually the resolution of these problems brings great benefits, such as improving the efficiency of industrial processes, reducing costs, increasing productivity, increasing the useful life of equipments, helping to make better choices, bringing better cost-effectiveness, improving customer satisfaction, etc.

Given the enormous variety of decision-making and/or optimization problems and the complexity of such problems, the scientific literature brings numerous scientific materials that deal with the various problems and propose different solution methodologies for these problems. However, it is rare to find software that solves such problems. Usually such software is closed code, developed directly to solve very specific organizational problems, and the rights of use are kept by the developers and organizations involved in the software development project. Even when this does not occur, many pieces of software developed in research projects are simply not disclosed and end up being lost over time.

For this reasons I have developed COPSolver (Combinatorial Optimization Problems Solver), to make available to everyone cientific tools that can help improve the decision management process as well as improve the efficiency of various organizational processes.

2. Software description

COPSolver software was (and is being) developed by Dr. Tatiana Balbi Fraga with the purpose of solving several decision and optimization problems in the most efficient and robust way possible.

The first version of the software, COPSolver_1.0-1, applies the exact solution method developed by Fraga et al. (2023) to solve the Multi-product Batch Processing Time Maximization (MBPTM) problem. This problem arises in production process operations where a set of products are processed simultaneously on the same machine, but with production rates that can be different for each product. There is a production limit for each product and also for the products group, as well as a limit for the batch processing time. The problem consists of defining the maximum processing time of the multi-product batch respecting the known restrictions.

The exact method of Fraga et al. (2023) is extremely efficient, proposing solutions very quickly and with low computational cost.

COPSolver package also contains the MBPTM.lng file, which is a code written in LINGO language to solve the model proposed by Fraga et al. (2023) for the MBPTM problem. When run, COPSolver automatically generates a data.ldt file to be used with this LINGO code. However, the MBPTM.lng file can only be used through the LINGO software. More details about the LINGO software are available at <https://www.lindo.com>.

2.1 Software architecture

COPSolver is an open source software written in C++ language. Its architecture has been planned as follows: solver will be formed by a main.cpp file along with a library set (.h) and their respective source codes (.cpp). Each library will be developed to define and solve a single distinct type of problem. All libraries will contain two main structures called problem and solution, which will be used to define the parameters of the problem and the desired solution, respectively. Each library will also contain a class, called cop (for combinatorial optimization problems) or otherwise according to the nature of the problem addressed. Such class will basically contain functions for constructing/reading the problem, functions for solving the problem, and printing functions. Functions with similar purposes will be named with the same name using the functionalities of polymorphism. The libraries will be distinguished by a distinct namespace containing an acronym that identifies the problem addressed in the library itself. The definition of the problems instances can be done in three distinct ways: 1) through the data.txt file; 2) using solver predefined problems; and 3) generating a problem statically or randomly through pre-established functions of the solver. The config.txt file should be adjusted in order to: 1) inform the type of problem addressed; 2) select the method of defining the problem and the parameters that should be adjusted according to the chosen method; and 3) select the method for solving the problem. In the future, an extra library will be developed for automatic identification of the problem according to the definitions presented and for solution by the best method. For better suitability to the codeocean repository, the computational codes will be allocated within a folder named code, the files containing the input data (config.txt and data.txt) will be allocated within a folder named data, and the output files will be inserted into a folder named results. Also, files containing .h computational codes will be inserted inside a folder named lib while .cpp files (except main.cpp) will be inserted inside a folder named src. The lingo-language solvers developed for COPSolver validation will be available in specific subfolders within the folder named LINGOSolver.

3. Economic, technological, scientific and educational impact

The determination of the maximum processing time of multi-product batches is extremely important for some industries, as it is directly related to inventory management as well as to the correct fulfillment of demand. Therefore, COPSolver_1.0-1 can serve as an important tool for inventory control, reducing costs and improving demand compliance. As future works, we will be applying COPSolver to assist in planning the production of extruders, in companies of the plastic bag production sector. With this we hope to improve inventory management, increasing the competitive potential of these companies.

COPSolver_1.0-1 can also be used to conduct case studies, assisting in teaching and research. Also, since COPSolver is built on the basis of problem-oriented libraries, it can also be reused in the development of other more complex software in research and technological projects.

4. Next steps

Professor Fraga, author of this paper, and, more generally, her research group GAMOS (Group of Analysis, Modeling and Optimization of Systems), have developed many works focused on the identification, modeling and solution/optimization of real problems in industrial/organizational environments. Therefore, the development of COPSolver will be done in conjunction with the development of these works. Our intention is that the software can, in the near future, cover a wide range of methodologies to solve real problems and standard problems found in the literature. Later we intend to develop a pattern identification method to recognize problems patterns and solve different problems automatically. In the future, we will also be developing extension projects to train

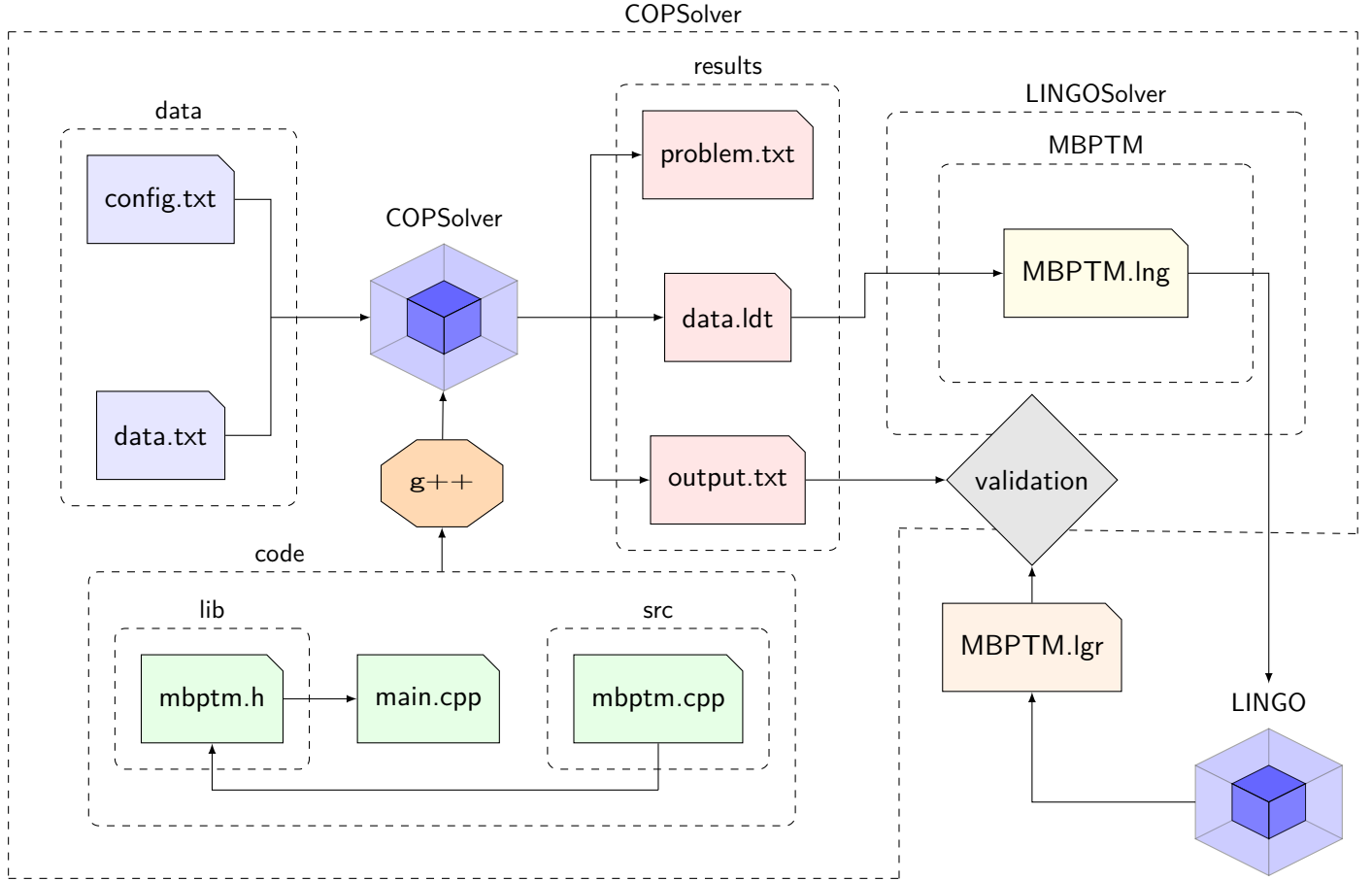


Figure 1: Software architecture.

micro and small companies to use COPSolver. We hope that COPSolver can help to improve several processes and, in particular, the productive and administrative processes of micro and small companies, offering them a tool to improve their competitive potential.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Huge thanks to the teams responsible for developing elementary OS, as well as google translator, duckduckgo, gummi, CodeBlocks, LINGO, git, GitHub, gcc, and TeXLive libraries. Without the availability of these tools, COPSolver could never have been developed. I am also very grateful to my co-workers Regilda da Costa e Silva Menêzes and Marcos Luiz Henrique for their constant support to the projects I develop and to my dear advisees Aldênia Karla Barrêto Candido, Lays Cristiane Cavalcante de Alcântara Aguiar, Edwedja de Lima Silva and Ítalo Ruan Barbosa de Aquino, among others, for having participated so enthusiastically in the projects that gave rise to this software, in particular for having made the effort to describe real processes in such detail. I would also like to thank the company in the plastic bag sector and other companies that made the studies possible as well as the identification of the new optimization problem addressed in COPSolver_1.0-1.

References

Fraga, T.B., Aquino, Í.R.B. and Menêzes, R.C.S. (2023). Multi-product Batch Processing Time Maximization Problem. Manuscript submitted for publication.