Combining Metaheuristics and Integer Linear Programming: A Hybrid Methodology Applied to the Container Loading Problem

Napoleão V. Nepomuceno, Plácido R. Pinheiro, André L. V. Coelho

Mestrado em Informática Aplicada – Universidade de Fortaleza
Av. Washington Soares, 1321, CEP 60811-905 – Fortaleza – CE – Brazil

napoleao@edu.unifor.br, {placido,acoelho}@unifor.br

Abstract. *This paper presents a hybrid framework for coping with hard combinatorial optimization problems, which is based upon a methodology combining heuristic and exact methods. In this framework, a metaheuristic engine works as a generator of reduced instances for the problem, which are formulated as mathematical programming models. These instances, in turn, are solved by an exact optimization technique, and the performance measures accomplished by the respective models are interpreted as score values by the metaheuristic, thus guiding its search process. As a means to assess the potentialities behind the novel approach, we provide an instantiation of the framework to deal specifically with the container loading problem.*

1. Introduction

The inherent difficulty and the enormous practical relevance of solving hard real-world problems have led to a large research effort in the development of heuristic methods aimed at finding good approximate solutions. Usually, metaheuristics have succeeded in providing a proper balance between the efficiency and effectiveness criteria while tackling many non-trivial optimization problems. However, it is also quite consensual that there is still much room for the conceptualization of more sophisticated solution methods in this research area.

In recent years, it has become ever more evident that a skilled combination of concepts stemming from different metaheuristics can be a very promising strategy to confront complicated optimization tasks. The hybridization of metaheuristics with other operations research methods has been shown great appeal as well, as they typically represent complementary perspectives over the problem solving process as a whole. In general, combinations of components coming from different metaheuristics and/or from more conventional exact methods into a unique optimization framework have been referred to by the label of “hybrid metaheuristics” [2].

This paper presents an optimization framework, which is based upon a methodology that complies directly with the aforementioned hybridization precept. The framework is instantiated for dealing specifically with the container loading (CL) problem, as a means to assess the potentialities behind the novel approach. Some simulation results we have achieved by experimenting with a series of benchmark problems, as well as with the practical case study developed in a metallurgic factory, are presented. Some final remarks with indication for future work conclude the paper.

2. Hybrid Framework

The hybrid framework prescribes the integration of two distinct components, according to the setting illustrated in Figure 1. The first component is named as the Generator of Reduced Instances (GRI), which is in charge of producing reduced instances of the original problem. In theory, any metaheuristic can be recruited for such a purpose, provided that the reductions carried out always respect the constraints imposed by the problem at hand. This is necessary to ensure that an optimal solution found for any instance would also be a feasible solution to the original problem. Conversely, the role of the second element, referred to as the Solver of Reduced Instances (SRI), is to interpret and solve any of the generated instances coming out of the GRI. The optimal objective function values achieved with the solving of these sub-problems will serve as feedback to the GRI, in a manner as to guide its search process. Any exact optimization technique could be, in thesis, a candidate to act as SRI (see [4] [8]).

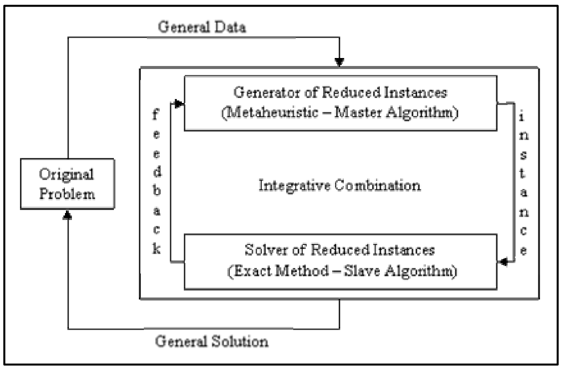

Figure 1. Hybrid framework under investigation.

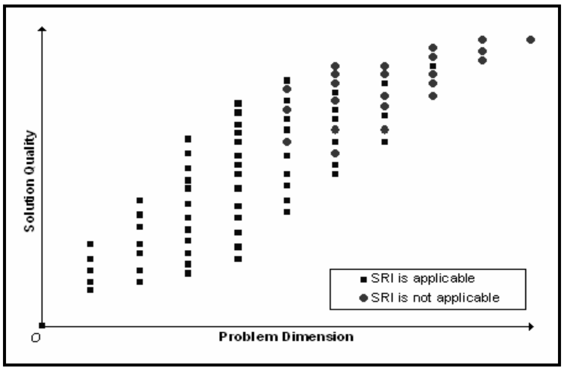

Figure 2. Hypothetical search space of the GRI.

According to the classification proposed in [9], the methodology falls into the category of integrative combinations. The quality of the solutions to the instances generated by the metaheuristic is determined when the sub-problems are solved by the exact method. However, there are cases where it is not possible to solve, or even generate, the reduced problem instance, due to its complexity. Thus, roughly speaking, the hybrid framework aims at determining sub-problems which could be solved by an exact method and present efficient solutions. The best solution obtained throughout the whole process is deemed to be the final solution to the original problem. Figure 2 represents, hypothetically, the search space of the GRI metaheuristic, as well as demonstrates the distribution of sub-problems according to its dimension (size), solution quality, and the applicability of the SRI. Thus, the solution space of the framework is restricted to the sub-problems where the SRI is applicable.

Basically, the methodology, as informally described so far, roughly comprehends a sequence of three distinctive steps, discussed in the following [4] [7].

Mathematical Formulation of the Problem

In this stage, the aspects to be incorporated in the formal model and the suppositions that can be made relative to the problem itself need to be specified. The desired objectives, the decision variables, and the considered constraints, all need to be made explicit in the mathematical formulation adopted.

Identification of the Reducible Structures

In principle, it would be possible to setup the entire model and solve the original problem using an exact method. However, the number of variables and constraints tends to grow extraordinarily in practical cases. In this step, the task is to identify the main entities that contribute somehow to the increase in the problem dimension. Thus, the idea is to generate and solve only sub-problems in an efficient way.

Specification of the Metaheuristic Sub-problem Generator

Finally, the choice of the metaheuristic technique to act as GRI should be done. This issue, like the previous one, deserves greater attention by the framework user, since it is the GRI component that is in charge to discover the reduced version of the original problem that could still provide the more adequate solution to it.

In the following, the methodology is revisited, providing information pertaining exclusively to the CL problem. The satisfactory tackling of this problem comes to be a challenging task, thus serving well the purposes of the novel approach evaluation.

3. Coping with the Container Loading Problem

The CL problem alludes to the task of packing boxes into container. More precisely, given the dimensions of the container and the boxes which need to be loaded, the problem can be defined as to find such an arrangement of boxes that optimizes a given objective function – in general, the maximum volume of the loaded boxes. To formulate the CL problem, we resort to an extension of the integer linear programming (ILP) model proposed in [3], modified in order to allow the rotation of the items (see [4] [5]).

Among the main entities that control the number of packing patterns of this model are the different types of boxes, their spatial dimensions, as well as the orientation constraints posed to these boxes. Several reduced instances of the original problem can then be eventually generated by excluding some of the box modes and/or some positions demarcated by the discretization sets (see [4] [6]).

A modified genetic algorithm has been implemented to account for the GRI task. Each of its individuals is represented by a binary chromosome, which encodes the discretization sets to the aforementioned ILP model. In this encoding, each gene represents one possible element of discretization along a particular dimension. Assigning a null value to a bit means that its associated element of discretization will not be used for the construction of the reduced problem; by flipping the value of this gene to one, the element will take part in the reduced problem instance (see [4] [7]).

4. Computation Experiments

To evaluate the potentialities behind the novel approach, a series of experiments have been carried out by our group, and some of the results achieved so far are discussed in the following subsections.

Benchmarking Problems

Some experiments have been performed over problems of the OR-Library available at http://people.brunel.ac.uk/~mastjjb/jeb/info.html. In Tables 1 and 2, we contrast the performance of the hybrid framework (HF) with that exhibited by the well-known combined heuristic B/R [1], in those problem instances where the latter have presented the worst and best performance indices, respectively. The hybrid framework beats the heuristic B/R in the worst-case instances, whereas, in the best-case instances, the heuristic B/R prevails. On average, our framework has shown superior performance, reaching the mark of 86.53% of effective volume utilization of the container across all these problem instances against the 83.53% score achieved by the heuristic B/R. Overall, our methodology has obtained satisfactory results. For instance, on average, considering all the problem instances of the Thpack1 group, we have achieved the mark of 87.52% (±3.49) of effective usage of the container space, outperforming the 85.40% (±4.30) score achieved by the combined heuristic B/R (see [4] [7]).

Table 1. Comparative results for the worst case instances of B/R.

Group	Problem	B/R (%)	HF (%)
Thpack1	44	73.72	76.01
Thpack2	11	73.79	86.19
Thpack3	46	75.33	87.88
Thpack4	93	78.38	88.10
Thpack5	70	78.71	86.94
Thpack6	93	75.22	87.60
Thpack7	74	75.73	87.89

Table 2. Comparative results for best case instances of B/R.

Group	Problem	B/R (%)	HF (%)
Thpack1	39	94.36	89.15
Thpack2	33	93.76	87.47
Thpack3	49	92.63	90.54
Thpack4	22	90.06	85.26
Thpack5	41	90.39	87.18
Thpack6	58	89.15	86.15
Thpack7	51	88.28	85.11

Case Study

The hybrid framework has also been applied in the ambit of a large metallurgic factory. This particular industrial unit has usually made use of a load planning and accommodation system that is quite well known in the market, namely *MaxLoad Pro* (ML). Among the six problem instances considered, the hybrid framework was able to find better solutions in four of the cases, with a tie in the other two (see [4] [7]). In Figures 3 and 4, the packing patterns produced by both approaches are illustrated.

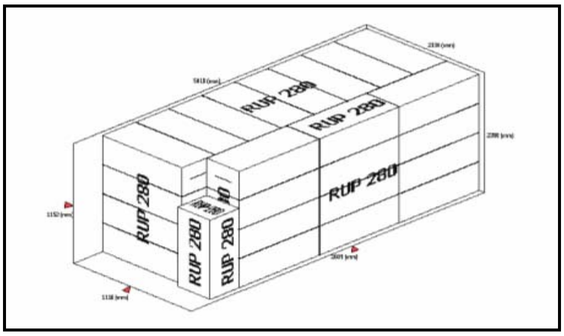

Figure 3. ML packing pattern (with 45 loaded boxes).

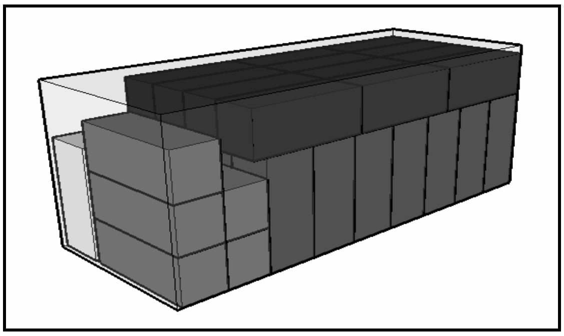

Figure 4. HF packing pattern (with 47 loaded boxes).

4. Final Remarks

We have presented a novel methodology for coping with hard combinatorial optimization problems, which is based upon a particular type of hybridization between approximate and exact algorithms. A particular instantiation for dealing specifically with the CL problem was delivered. Overall, the optimization performance achieved with the new methodology has been very satisfactory, taking as reference the results achieved by other approaches and taking into account the inherent difficulties associated with the problem – indeed, some problem instances could not be directly solved through the application of the exact optimization package alone. Likewise, in a real case study, the hybrid methodology has presented competitive solutions to those provided by a well-known commercial tool, prevailing in general over it. In order to evaluate the generalization level behind the novel methodology, we plan, as future work, to develop and test other instantiations of the conceptual framework for coping with other types of hard combinatorial optimization problems. Actually, we have already successful results for another cutting and packing problem [8]. Likewise, it is in our plans to investigate other types of metaheuristics (like trajectory-based or even hybrid ones) to play the role of the GRI component in the framework.

References

[1] Bischoff, E. and Ratcliff, M. (1995). "Issues in the Development of Approaches to Container Loading," Omega 23, p. 377-390.

[2] Blesa, M.J. *et al*. (2005). "Hybrid Metaheuristics." In: 2nd International Hybrid Metaheuristics Workshop, LNCS 3636, p. VI-VII.

[3] Morabito, R. and Arenales, S. (1997). "Abordagens para o Problema do Carregamento de Contêineres," Pesquisa Operacional 17, p. 29-56.

[4] Nepomuceno, N.V. (2006). "Combinação de Metaheurísticas e Programação Linear Inteira: uma Metodologia Híbrida Aplicada ao Problema de Carregamento de Contêineres." M.Sc. Dissertation, UNIFOR, Available at http://www.unifor.br /pls/oul/p_tese_defendida_ncm?p_nr_curso=83.

[5] Nepomuceno, N.V., Pinheiro, P.R. and Coelho, A.L.V. (2006). "Aplicação de uma Metodologia Híbrida ao Problema de Carregamento de Contêineres." In: XXXVIII Simpósio Brasileiro de Pesquisa Operacional, p. 1596-1603.

[6] Nepomuceno, N.V., Pinheiro, P.R. and Coelho, A.L.V. (2006). "Metaheurística e Programação Linear Inteira: Um Algoritmo Híbrido para o Problema de Carregamento de Contêineres." In: XIII Congreso Latino-Iberoamericano de Investigación Operativa.

[7] Nepomuceno, N.V., Pinheiro, P.R. and Coelho, A.L.V. (2007). "Tackling the Container Loading Problem: A Hybrid Approach Based on Integer Linear Programming and Genetic Algorithms." In: 7th European Conference on Evolutionary Computation in Combinatorial Optimization, LNCS 4446, p. 154-165.

[8] Nepomuceno, N.V., Pinheiro, P.R. and Coelho, A.L.V. "The 'Generate-and-Solve' Hybrid Framework: Combining Heuristic and Exact Methods for Tackling Cutting and Packing Problems." (Invitation to submit a full version for a special issue of Journal of Heuristics devoted to selected works from CLAIO XIII).

[9] Raidl, G.R. (2006). "A Unified View on Hybrid Metaheuristics." In: 3rd International Hybrid Metaheuristics Workshop, LNCS 4030, p. 1-12.