Double Fuzzy Stopping Rule for a hybrid metaheuristic for the Strip Packing Problem

Jesús David Beltrán Cano, Jose Eduardo Calderón Benito
Ignacio José García del Amo, Rayco Jorge Cabrera,
José A. Moreno Pérez and J. Marcos Moreno-Vega
Departamento de E., I.O. y Computación
Escuela Técnica Superior de Ingeniería Informática,
Universidad de La Laguna. 38271 La Laguna SPAIN.
e-mail: jamoreno@ull.es, jmmoreno@ull.es

Abstract: *In this work we propose a double fuzzy stopping rule for an hybrid metaheuristic approach to solve the Strip Packing Problem (SPP). The hybrid algorithm has two phases, in the second phase a* Variable Neighbourhood Search *(VNS) improves the quality of part of the solutions obtained with a* Greedy Randomized Adaptive Search Procedure *(GRASP) in the first phase. The stopping rule is one of the elements of the metaheuristics least studied due to, among other reasons, the imprecision in the terminology used by experts to formulate good stopping rules. We formulate in fuzzy terms a double stopping rule for solving the SPP. The double stopping rule decides when to stop each phase of the hybrid based on different characteristics of the solution found. The results achieved with the double stopping rule for the hybrid method are compared with those existing in the literature.*

Keywords: GRASP, VNS, Metaheuristics, Fuzzy Stopping Rules, Strip Packing Problem

1 Introduction

The *Strip Packing Problem* (SPP) consists of packing a finite set of the rectangles, that can be rotated 90 degrees, in the fixed width strip minimizing the height of the packing. Metaheuristic searches are general solution procedures for solving problems that must provide high quality solutions consuming moderate resource (time). A review of the application of metaheuristic algorithms to the SPP is found in [7]. Two metaheuristics with good performance are *Greedy Randomized Adaptive Search Procedure* (GRASP) [5] and *Variable Neighborhood Search* (VNS) [6]. In this work we consider a GRASP/VNS hybrid for the SPP. VNS operates on a part of the solution provided by GRASP that already has good characteristics in the other part. Specifically, VNS attempts to improve the packing of the last rectangles introduced into the solution by the GRASP, which uses a selection criterion that is quite good for packing the first rectangles but not so good for the last ones. Therefore, the first phase of our hybrid stops when it gets a solution with high quality characteristics in its first part and the second phase stops when the characteristics of the last part of the solution has also high quality. Since the concepts of *high quality* and *moderate time* are subjective, they are appropriately modeled by fuzzy sets. We propose a double fuzzy stopping criterion for the SPP for this purpose.

The rest of the paper is structured as follows. In the next section, we define the SPP and analyze the good characteristics of the first and last parts of the solutions. The GRASP and VNS metaheuristics and their adaptation for solving the Strip Packing Problem are described in section 3. The GRASP/VNS hybrid and the alternative designs of a double fuzzy stopping rules for this problem are described in section 4. In section 5 we describe the experimental comparative study and enumerate the conclusions.

2 The Strip Packing Problem

The *Strip Packing Problem* (SPP) [7] is formulated as follows. Let w be the width of a strip with infinite height and let $\mathcal{R} = \{R_i = R(w_i, h_i) : i = 1, \ldots, n\}$ be a set of n rectangles with width w_i and height h_i. The problem is to establish the optimal placing of all the rectangles. Each rectangle may be rotated 90 degrees or not rotated at all. The placing of the rectangles is feasible if all of them are completely included in the strip and there is not a pair of rectangles that overlap each other. The objective function to be minimized is the maximum height h reached by the rectangles. Equivalently, the objective is to minimize the area of the strip used that is the product of its width w by the maximum height h; $h \cdot w$.

Some infinite set of equivalent feasible solutions could be obtained by shifting horizontally or vertically some rectangles when it is possible. To avoid it, only solutions obtained by introducing successively all the rectangles following a given placement strategy are considered. We use the following usual *bottom-left* strategy: each rectangle is placed at the deepest location and, within it, in the most left possible location. So, each feasible solution of the problem is determined by the order in which the rectangles are introduced in the strip and their possible rotations.

The packing of the rectangles produces two kind of wastes. The inner one is the area enclosed by rectangles that is not occupied by other rectangles. The outer waste is the area between the top rectangles and the line that represents the maximum height of the packing. The placing of each new rectangle has to avoid inner wasted areas by adjusting its width side to the lowest segment of the upper contour. Moreover the placing of the last rectangles has to produce smooth upper contour to avoid outer wasted area.

If $A = w_1 \cdot h_1 + \ldots + w_n \cdot h_n$ is the total area of the rectangles to be packed, then the total wasted area is $W(X) = w \cdot h - A$. The upper contour of a (partial) solution is identified by the segments that constitute it from left to right. It is represented by the sequence $C = [(y^i, x_1^i, x_2^i) : i = 1, ..., c]$, where y^i is the height of the i-th segment and x_1^i and x_2^i its initial and final coordinates. The value of the objective function for a solution X with upper contour C is the maximum height of its segments; i.e., $f(X) = h = \max\{y^i : i = 1, \ldots, c\}$. The smoothness of the upper contour is measured by the outer wasted area. This area is constituted by the rectangles with width equal to the length of each segment and height equal to the corresponding distance to the maximum height. Its value is computed by:

$$W_{OUT}(X) = \sum_{i=1}^{c} (h - y^i)(x_i^2 - x_i^1).$$

Therefore the inner wasted area is: $W_{IN}(X) = W(X) - W_{OUT}(X)$. However, the inner waste is updated while the rectangles are sequentially packed.

3 GRASP and VNS metaheuristics

GRASP (Greedy Randomized Adaptive Search Procedure) [5] is a metaheuristic consisting of two phases. First, in the *constructive phase*, a solution is iteratively constructed by randomly selecting one element of an adaptive restricted list of the best candidates. Then, in the *post-processing phase*, it is attempted to improve this solution using an improved method (generally, a descent local search). We apply the Variable Neighbourhood Search (VNS) metaheuristic [6] for the second phase of the search. Common heuristic searches are based on transformations of solutions that determine a neighbourhood structure on the solution space. The VNS is a recent metaheuristic based on systematic change of the neighbourhood in a heuristic search.

In a constructive method an element is iteratively added to an initially empty structure until a solution of the problem is obtained. The choice of the item to be included in the partial solution is based on one or several heuristic evaluations that measure the convenience of considering the item as belonging to the solution. The heuristic functions depend on the problem and also on the knowledge of the decision maker about the problem. If the evaluation of an element depends on the items already in the solution, the function and the method are adaptive.

We consider measures of the wastes as heuristic functions. In addition to the heuristic function, it is necessary a strategy to select the elements. One of the most known strategy is the *greedy* rule: select the element that optimizes the heuristic function. However, this strategy shows poor performance in most cases. The alternative strategy used in GRASP is to randomly select one of the best elements. The set of best elements is called *Restricted Candidate List* (*RCL*).

At any iteration t of the constructive phase, let $\mathcal{R}(t)$ be the set of rectangles out of the partial solution $X(t)$. Let $C(t)$ be the contour at the iteration t that is determined by $X(t)$. Every rectangle in $\mathcal{R}$ is evaluated by its adjustment to the lowest segment of $C(t)$; i.e., the one with minimum y^i. Let $y^j = \min\{y^1, y^2, ..., y^c\}$ and $l = x_2^j - x_1^j$ be the length of the lowest segment. Then, given a value $\rho \in [0, 1]$, the restricted candidate list, RCL, is built as follows.

$$RCL(t) = \{R(w_i, h_i) \in \mathcal{R}(t) : (w_i \leq l \leq w_i + \rho \text{ or } h_i \leq l \leq h_i + \rho\}.$$

Only the rectangles that best fit to the length l of the lowest segment of the contour are in the restricted list. Every time a new rectangle is packed in the strip the upper contour must be updated accordingly. If the list RCL is empty, we take from $\mathcal{R}_2$ the rectangle that best fits to lowest segment. If such rectangle does not exist, a new wasted rectangle area appears and we have to rebuild the contour $C(t)$.

The GRASP strategy is conducted by the inner waste but VNS consider also the outer wasted area. It takes a solution provided by GRASP and tries to reduce the total waste area. A *neighbourhood structure* $\mathcal{N}$ on X is a function $\mathcal{N} : X \to 2^X$ that associates to each solution $x \in X$, a neighbourhood of solutions $\mathcal{N}(x) \subset X$. *Variable neighbourhood search* (VNS) [6] arose from a simple idea: change the neighbourhood structure when a local search is trapped on a local minimum. A VNS performs a combination of series of shake procedures and descent searches. When the descent search stops at a local minimum, the shake procedure gets a new starting point for a new descent search. The *General Variable Neighborhood Search* (GVNS) method applies two (possibly different) series of neighbourhood structures; one for the shaking and one for the descent. The *Variable Neighborhood Shaking* (that is equivalent to a Reduced VNS) gets iteratively a solution of the neighbourhood of the current solution and the method changes the neighbourhood structure each time a new solution is found. The *Variable Neighborhood Descent* (VND) method changes the neighbourhood structure each time a local optimum is reached. The search ends when there is not possible to improve the solution within all the neighbourhood structures.

A simple VNS can be implemented by shakes and local searches using a base move. When the improving local search stops at a local minimum, the shake procedure gets a new starting point for the local search. A basic local search consists of applying an improving base move until no such move exists. A simple shake procedure consists of applying a number of random base moves. The base move for problems where the solutions are represented by permutations is the interchange of the position of two elements of the permutation. In the SPP using the bottom-left strategy only a binary variable is needed to decide if the rectangle is rotated or not.

The series of neighbourhood structures based on a single base move involved in this procedures are nested. Given the base neighbourhood structure $\mathcal{N}$ the series of nested neighbourhood structures is defined as follows. The first neighbourhoods are the base ones, $\mathcal{N}_1(x) = \mathcal{N}(x)$ and the next neighbourhoods are defined recursively by: $\mathcal{N}_k(x) = \mathcal{N}(\mathcal{N}_{k-1}(x))$. In other words $\mathcal{N}_k(x)$ consists of the solutions that can be obtained from x by a series of k base moves.

The *Nested VNS* for the SPP comprises the steps given in the figure 1.

```
1. Initialization: Take an initial solution $x$. Set $x^* \leftarrow x$ and $k \leftarrow 1$.
2. Repeat the following until the stopping condition is met:
(a) Shake: Apply $k$ random moves to the solution $x$ to get $x'$.
(b) Local Search: Apply the best improving move to $x'$ until a local minimum $x''$ is found.
(c) Improve or not: If $f(x'') < f(x^*)$, do $x^* \leftarrow x''$ and $k \leftarrow 1$. Otherwise $k \leftarrow k+1$. Set $x \leftarrow x^*$.
```

Figure 1: Nested VNS algorithm

4 The hybrid and its double fuzzy stopping rule

The hybrid approach proposed is obtained by using a nested VNS to improve the solution provided by the constructive procedure of GRASP. Given the constructed solution provided by GRASP, the algorithm extracts the last k rectangles of the solution and applies the nested VNS to pack them considering the $n-k$ rectangles already included in the strip. Figure 2 shows a GRASP/VNS hybrid approach that is very similar to that proposed in [2]. The values for k is chosen taking into account the instance data.

```
• Apply iteratively the constructive phase of the GRASP.
• Let $[R_1, R_2, \ldots, R_n]$ be the solution provided by the GRASP.
• Let $[R'_1, R'_2, \ldots, R'_k]$ be the last $k$ rectangles of the solution.
• Apply the nested VNS to the packing of these rectangles $[R'_1, R'_2, \ldots, R'_k]$ in the free strip leaved by this packing of the rectangles $[R_1, R_2, \ldots, R_n]$.
• Let $[R^*_1, R^*_2, \ldots, R^*_k]$ be the best packing obtained.
• Return the solution $[R_1, R_2, \ldots, R_{n-k}, R^*_1, R^*_2, \ldots, R^*_k]$.
```

Figure 2: GRASP/VNS hybrid

One of the elements that most influence has in the success of the search procedures to solve a problem is the stopping criterion applied. The stopping rules must provide an appropriated trade-off between efficiency and efficacy of the solution procedure by detecting when a hight quality solution is reached. The double stopping rule decides when to stop both phases of the hybrid metaheuristic. The GRASP phase stops when it gets a solution with small inner waste. The VNS stops when it gets a solution also with smooth contour; i.e., a solution with small total waste. We use fuzzy set techniques to formalize these stopping rules.

In [9] and [10] we found the first fuzzy stopping rules appeared in the literature that are used for exact algorithms to solve the Knapsack and Travelling Salesman Problems. Those rules are based on the fact that most of the decision makers would accept solutions that are not optimal but good enough. They use membership functions depending on the objective to establish the set of good solutions. The corresponding α-cuts determine general and robust stopping criteria. The Fuzzy sets-based heuristic FANS [3], [4] is the first metaheuristic that incorporate fuzzy sets and it includes the use of a fuzzy or subjective valuation of the solutions, that is one of its most relevant elements.

The stopping rules, like the solution procedures, can be general or specific. From the point of view of of Metaheuristics field, where the generality of the procedures is one of the most relevant desirable characteristics, the stopping rules must also be general. However, as we show in [1], the stopping rules depending of the problem are more efficient and effective.

Our stopping rules for the Strip Packing Problem use the property of "high quality". The inner and outer wastes are two characteristics that indicate the quality of two parts of the solution. Solutions with a *"small"* wastes are high quality solutions. In [8] we use "small" inner waste and smooth contour (i.e., small outer contour) to define a fuzzy stopping rule for a GRASP procedure. Here we use the fuzzy concepts of small wastes to define a double fuzzy problem-dependent stopping rules.

Let $\lambda \in [0,1]$ be a value fixed by the decision maker. The set of solutions with small waste is given by the membership function μ given by $\mu(X,\lambda) = 1 - \lambda W(X)/A$. Thus, the membership function of a solution is 1 if ther is not waste ($W(X) = 0$), and decreases linearly when the waste $W(X)$ increases. A fuzzy stopping rule based on the waste stops the search when the process finds a solution with small waste. This can be applied to inner, outer and total waste. The smooth contour is given by small outer waste. Similar fuzzy stopping rules are obtained by considering a different kind of waste. For the GRASP/VNS hybrid we propose a double fuzzy stopping rule consisting in stopping the first phase when it finds a solution with small inner waste and stopping the second phase when it finds a solution also with small inner waste; i.e., with small total waste.

In the practical implementations, the decision maker fixes the level α for the α-cut. For instance, to stop the search when $\mu(X,\lambda) \geq 1-\alpha$, that is, when $W(X)/A \leq (1-\alpha)/\lambda$. The concrete values for the two parameters that appear in the stopping rules can be fixed together. For instance, to say that the percentage of inner waste must be no more than 5% of the total area A of the rectangles we set $(1-\alpha)/\lambda = 0.05$. For the total waste, since it depends directly to the height h, it is practically equivalent to fix the absolute or relative gap with respect to the ideal height; i.e. $h_{opt} = A/w$.

5 Computational Experience

The computational experiments were performed to analyze the performance of the double fuzzy stopping rule for the GRASP/VNS in solving the Strip Packing Problem. The fuzzy stopping rules bound the values for the percentages of waste. The double stoping rule stops the first phase when the inner waste is less than a given percentage of the total area of the rectangles and the second phase when total waste is less than a corresponding percentage.

Table 1 shows the results obtained with the GRASP/VNS hybrid metaheuristics using the double stopping rule compared with a Simulated Annealing [7] that also uses the bottom-left strategy. The instances used are those corresponding to the categories $C_1, C_2, \ldots, C_7$ of Hopper and Turton [7] (available at: `mscmga.ms.ic.ac.uk/jeb/orlib/stripinfo.html`). Each category consists of 3 instances with the same width and best objective value and very similar number of rectangles, as shown in the first columns of table 1. The next pairs of values show the average height and time in seconds corresponding to the GRAPS/VNS with the proposed double stopping rule. The last two values correspond to the results of average objective values and times for the SA+BL algorithm provided by Hopper and Turton [7] where the time is in minutes. They are average values on the 3 instances in each category that have been solved 10 times.

6 Conclusions

The double fuzzy stoping rule allows to get solutions with high quality that have small interior waste and smooth upper contour. This stopping rule applied to the VNS/GRASP hybrid provide solutions with similar quality in a less order of magnitude computation time.

				GRASP/VNS		SA + BL	
	n	w	h_{opt}	h (height)	t (in sec.)	h (height)	t (in min.)
C_1	16,17	20	20	21.000	1.142	20.8	0.7
C_2	25	40	20	16.667	36.116	15.9	2.4
C_3	28,29	60	30	32.167	81.302	31.5	4
C_4	49	60	60	63.467	139.023	61.8	33
C_5	72,73	60	90	94.167	90.471	92.7	1157
C_6	97	80	120	126.300	177.653	123.6	382
C_7	196,197	160	240	252.100	373.716	249.6	4181

Table 1: GRASP/VNS vs. SA+BL

Acknowledgements

This work has been partially supported by the Spanish Ministry of Science and Technology through the projects TIC2002-04242-C03-01 (70% of which are FEDER founds).

References

[1] Beltrán, Jesús David; Eduardo Calderón, Jose; Jorge Cabrera, Rayco; Moreno Vega, J. Marcos. Fuzzy Stopping Rules for the Strip Packing Problem. In IPMU 2004, 673-680.

[2] Beltrán, Jesús David; Eduardo Calderón, Jose; Jorge Cabrera, Rayco; Moreno Vega, J. Marcos. GRASP/VNS hybrid for the Strip Packing Problem. In 1st International Workshop on Hybrid Metaheuristics (2004) 79-90.

[3] Blanco, A., Pelta, D., Verdegay, J.L. A Fuzzy Valuation-based Local Search Framework for Combinatorial Problems. *Fuzzy Optimization and Decision Making.* (2002) 1, 177-193.

[4] Blanco, A., Pelta, D., Verdegay, J.L. Applying a Fuzzy Sets-based Heuristic to the Protein Structure Prediction Problem. *International Journal of Intelligent Systems.* (2002) 17, 629-643 .

[5] Feo, T.A.; Resende, M.G.C. Greedy Randomized Adaptive Search Procedures. Journal of Global Optimization (1995), 6, 109-133.

[6] Hansen, P., Mladenovic, N.: Variable neighborhood search: Principles and applications. *European Journal of Operational Research.* (2001) 130, 449–467.

[7] Hopper, E., Turton, B.C.H.: A Review of the Application of Meta-Heuristics Algorithms to 2D Strip Packing Problems. *Artificial Intelligence Review* (2001) 16, 257–300

[8] Moreno Pérez, José A., Moreno-Vega, J. Marcos: Fuzzy Constructive Heuristics. En *Fuzzy Sets Based Heuristics for Optimization*, José L. Verdegay (Ed.). Springer Verlag. Series: Studies in Fuzziness and Soft Computing, Volumen. 126 (2003) 83–96

[9] Sancho-Royo, A., Verdegay, J.L., Vergara-Moreno, E.: Some Practical Problems in Fuzzy Sets-based Decision Support Systems. *Mathware and Soft Computing* (1999) Vol. VI, N. 2-3, 173–187.

[10] Verdegay, J.L., Vergara-Moreno, E.: Fuzzy Termination Criteria in Knapsack Problem Algorithms. *Mathware and Soft Computing.* (2000) Vol. VII, N. 2-3, 89–97