Control and Cybernetics

vol. **25** (1996) No. 1

Assortment problems with cutting policies

by

Pedro Castro Borges* and **José Soeiro Ferreira****

* Universidade Católica Portuguesa, and
Institute of Mathematical Modelling (IMM), Bldg. 321,
The Technical University of Denmark,
DK - 2800 Lyngby, Denmark,

** Faculdade de Engenharia da Universidade do Porto, and
Instituto de Engenharia Sistemas e Computadores (INESC),
Largo Mompilher 22 – Apartado 4433,
4007 Porto, Portugal

Abstract: Assortment selection problems with cutting policies play an important role in several industries but, possibly due to their combinatorial characteristics, they did not receive as much attention as cutting stock problems did, though the two problems are closely related. In this paper we graph several examples of cost functions of one and two-dimensional problems which reveal that many local optima with cost close to optimum exist. Several implementations of known meta-heuristics are tested with a real problem. Two different neighbourhood structures are considered and the different performance of the implemented heuristics is briefly discussed.

Keywords: assortment problem, cutting stack problem, meta-heuristics, simulated annealing, tabu search.

1. Introduction

When a product or raw material is produced in large quantities, some decisions must be made regarding the choice of characteristics of the units to be produced or stocked. Only a limited number of types can be produced or stocked and these units must meet client requirements while minimising waste or maximising profit. If the clients' needs do not exactly match the produced units then something must be done to satisfy their demand with the existing stock, generating costs or value loss. The problem of determining the best set of units to offer or stock is generally known as the assortment problem and is frequently associated with cutting stock problems. A survey on assortment problems can be found in Borges and Ferreira (1994).

The assortment selection problems are briefly presented in Section 2 while Section 3 discusses some issues that arise in assortment selection problems with cutting stock policies. In Section 4 we study the solution space by graphing some objective functions for both one and two-dimensional problems (multi-pattern and multiple stock sizes). This is important to better compare solutions and solution procedures. It is shown that, in some cases, solutions very close to the global optimum may be easy to find, as long as an optimising approach is used to plan the cut operation.

In an attempt to find a method that is able to deal with assortment problems, some meta-heuristics are employed, including simulated annealing and tabu search as briefly described in Section 5. Two strategies for neighbour generation are used and the results are discussed in Section 6.

2. Assortment selection

Assortment problems, or assortment selection problems, are often associated with raw material production, product planning and production planning and appear in many industries. Although different problems have been classified as assortment problems, it is possible to present a general formulation that groups them: Some products may be produced or bought in a set of different sizes or qualities, represented in Figure 1 as "candidates". The produced or stocked articles are used to satisfy a deterministic or stochastic demand but economic and/or logistical reasons forbid that all the demanded types be produced or stocked; instead a subset of the "candidates" set must be chosen. This subset, referred to as *assortment* or *catalogue*, is used to supply the orders.

The demand for products that are not stocked is supplied from the larger, or in some way better or more valuable products in the assortment. This is done by simple substitution or through cutting operations, implying an additional cost or a loss of value.

The objective is to find the assortment that gives a balance between the stocking or production costs and the substitution or waste costs. The solution of the assortment problem might also include the planning of the operation referred to in Figure 1 as "Demand Supply Process".

There are often practical restrictions that influence the decisions and generate problems with different characteristics, some of which are equivalent to problems in other areas, under other designations. A recent survey and bibliography on assortment selection problems can be found in Borges and Ferreira (1994) and older ones in Hinxman (1980) or Pentico (1986).

In this paper we deal with assortment problems with deterministic demand and where cutting policies are used, i.e., where several of the demanded units can be obtained from one unit of the catalogue. It is important to distinguish the number of dimensions relevant for the assortment problem from the dimensionality of the cutting operation, because cases exist in which they are different.

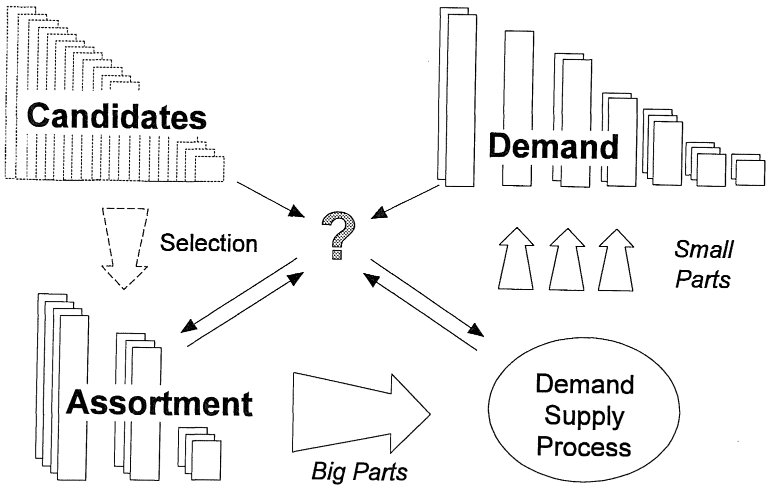

Figure 1. Schematic representation of assortment problems

In the example shown in Figure 2 the cutting operation is two-dimensional but only one value can be chosen for the height of the stock parts.

Problems that imply only substitution policies are usually easier than those involving cutting policies. Examples solved optimally can be found in Vidal (1994) or Pentico (1971).

3. Cutting policies

In assortment problems involving cutting stock operations, it is important to study the relation between the two decisions (assortment selection and cut planning) and to evaluate the consequences of the assortment selection. Assortment problems with cutting policies can be seen as a generalisation of cutting stock problems where the stock sizes are no longer fixed and so the selections of sizes and of cutting patterns are addressed together. This accentuates the combinatorial nature of the problem.

In this paper we deal with one and two-dimensional orthogonal guillotine cuts, allowing the existence of multiple stock sizes. These classes of cutting problems are very common and we face them in our co-operation with metal and furniture industries, which motivated the investigation on assortment selection problems.

To classify and distinguish different kinds of assortment problems we use a subset of a simple notation described in Borges and Ferreira (1994) which indicates the number of dimensions relevant for the assortment selection and the type of demand supply process which in this paper is always represented

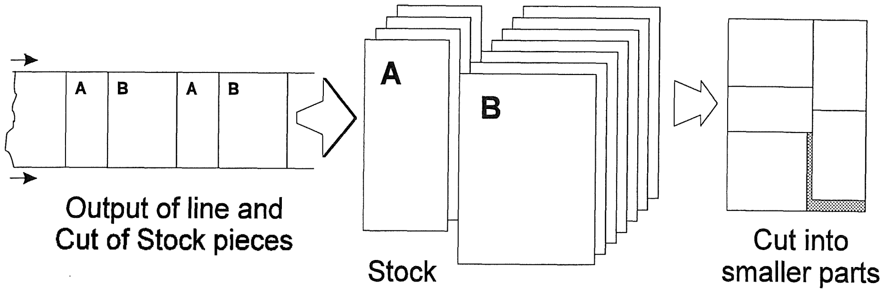

Figure 2. All stock sizes share the same height but several widths are allowed

by the dimensionality of the cut operation. So we employ the code 1/1 for problems where only one dimension is relevant for both the assortment and for the cut planning. We use the notation 2/2 for problems where two dimensions are relevant to both decisions. In this notation the example in Figure 1 would be noted 1.5/2 (this problem was considered in Chambers (1976)).

The problems addressed here can be stated in the following way:

From a set of sizes defined as the integers or pairs of integers inside intervals, choose a subset of N sizes that, when used to supply a demand of smaller parts with integer measures, yields the lower cost. The cutting process consists of up to three-phase guillotine cuts and no pre-assumptions are made on the cost of the stock parts.

In most cases, as in the examples presented ahead, the stock parts' cost is proportional to their weight, i.e., proportional to length in one-dimensional cases and proportional to area in two-dimensional examples.

Several methods have been proposed to address assortment problems with cutting policies but the combinatorial characteristic of the problems forces the methods to only engage 'limited' versions of the problems. No algorithm has been proposed, to the knowledge of the authors, to solve practically and optimally the general 2D/2D assortment problem with multiple stock sizes.

Gemmill and Sanders (1990) generate approximate solutions for a stochastic version of the problem, though that is not the case addressed here.

The quest for a general method that might be able to solve most assortment problems demands further efforts and the comparison of solution procedures; here some basic difficulties arise:

- The proposed methods are generally conceived to deal with specific problems that are particular in relation to demand satisfaction constraints, type of assortment, dimensionality or cutting procedures. This implies comparison biases that are difficult to eliminate.

- The methods to optimise nesting and cutting procedures used when choosing assortments influence the behaviour of heuristics for assortment selection. When comparing methods for assortment selection, a dilemma arises, between two arguments related to which cut planning methods one should use: One argument is that it is not fair to compare heuristics for assortment selection using other cut planning procedures than the ones used by the original authors, in particular when the assortment selection heuristic relies on characteristics of the cut planning procedure, as in the heuristic of Chambers and Dyson (1976); the other argument is that in order to make a fair and consistent comparison, the same cutting method should be used for different assortment selection procedures, as done in Gemmill and Sanders (1991).
- What merit should be awarded to the solutions found by any method? When one has several solutions it is easy to compare them quantitatively, but how should one compare them qualitatively? Up to which degree a specific method gives better solutions than another? The answer to these problems demands a general knowledge of their structure that in many cases does not exist.

The above mentioned difficulties lead us to approach these problems in the following way: first we take a look at some objective functions in order to gain knowledge about the solution space and as a pre-requisite to evaluate and compare solutions and solution procedures; then we employ some general search procedures and compare the results. Several methods are used: descent methods, simulated annealing, tabu-search and a variant of tabu-search, as described further on in Section 5.

4. Objective function

The analysis of any optimisation problem benefits from the knowledge of the solutions space which, in many cases, cannot be represented. In the examples of 1/1 and 2/2 assortment problems, it is possible to generate a graphical representation of the cost or objective function. To do so one must find the optimal cutting strategies for each possible assortment.

Calculating the cost functions demands intensive use of computer resources. To solve the cutting stock problems, both in one and two-dimensional cases, the PLACORTE software developed and commercialised by INESC was used. This software is based on a fast column generation technique (Oliveira and Ferreira (1994)) which improves the traditional Gilmore and Gomory (1965) delayed column generation by solving the LP problems to optimality in about half of the time. This algorithm uses specialised heuristics for the placement of the last pieces, in order to obtain integer solutions. Integer programming could have been used to solve the cutting stock problems, but the time cost would have been prohibitive for this kind of analysis, particularly in 2D problems. Furthermore, it is known that the round up property holds in many instances

Demanded width	Units demanded
16	2
67	32
75	1
84	1
182	10
194	7
207	5
220	5
245	7

Table 1. Problem AÇOS1

of cutting stock problems (see Marcotte 1985 and Wäsher and Gau 1993). A problem is said to have that property if the number of stock parts used in the optimal integer solution can be obtained by rounding up the number of parts used in its relaxed (LP) version. In our examples this property holds for most assortments. For the example shown ahead under the designation AÇOS1, the round-up property is verified in more than 97% of the assortments, as shown in Borges (1994) by comparing the solutions of the relaxed problems with upper bounds for the integer ones.

As an example of a simple 1/1 problem we take the problem AÇOS1, whose data is presented in Table 1. This example deals with the width of steel rolls and originated from the metal industry (F.Ramada S.A.). Figure 3 shows the cost values for the possible stock widths when only one width is to be used. The data for its generation was obtained by solving a cutting stock problem for each possible stock width. The cost of a solution with no waste is represented by a horizontal line and an arrow indicates the stock size originally used.

The cost function exhibits a saw pattern behaviour, because it measures the cost or total width, which increases with the stock size but drops abruptly each time the cutting patterns accomplish using one stock unit less. The graphic has two regions: when the stock size is small the cost behaves more irregularly; and for bigger stock sizes it becomes regular and predictable. These two regions correspond to different zones in the solution of the relaxed cutting stock problem, an similarly irregular first region followed by an almost flat line as shown in Borges (1994). Contrary to what is observed for smaller stock sizes, the minima for larger stock parts have very similar costs which are very close to the ideal cost (no waste), though they never reach it. As would be expected, the distance between local minima increases as the stock size increases, as do the maximum costs.

The original roll width (1030 mm) is clearly inadequate to the demand since

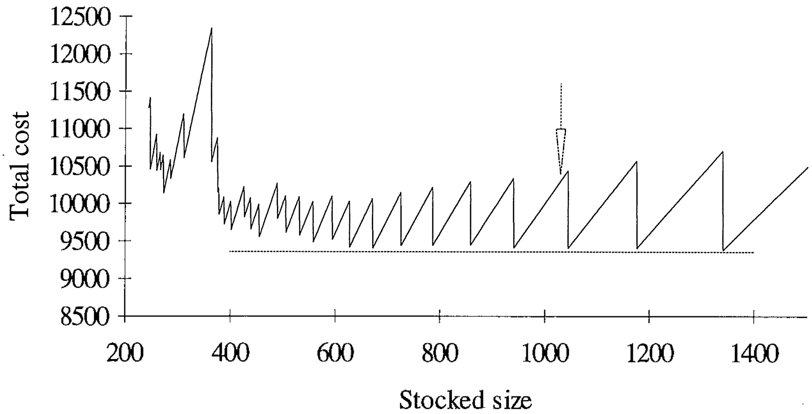

Figure 3. Objective function for problem AÇOS1

it leads to a cost so close to a maximum that the company could reduce the steel cost by about 8% if a better choice of the roll width was made, i.e., by improving the assortment.

As the next example, we graph the total cost of a problem using the list of small sizes from Gilmore and Gomory (1963), reproduced in the test data collected in Wäsher and Gau (1993). The problems considers 30 different demanded sizes, which makes the cut and stock problems much bigger and difficult to solve. That data generated Figure 4, where the displayed zone corresponds to the initial part shown in Figure 3. This curve also shows high initial costs but here the amplitude of the jigsaw patterns is like noise if compared with the global variations. The original stock size and the value of the ideal solution are again indicated by an arrow and a horizontal line, respectively. We can observe that the stock size used by Gilmore and Gomory is good for the demand considered.

Cost values for two-dimensional assortment problems can be represented as 3D polygon meshes. To represent those surfaces the number of cutting stock problems that need to be solved is bigger than in the previous examples since two dimensions must be swept. The time necessary to solve a two-dimensional cutting stock problem is also longer.

Table 2 presents data for a 2/2 example, named PINHO30, that originates from the furniture industry (Móveis Machado S.A.). In the cutting operation no rotations are allowed. Figure 5 shows the corresponding cost surface whose behaviour is of the same type that was observed in the 1/1 examples. In two-dimensional problems found in this kind of industry the size of the stock parts

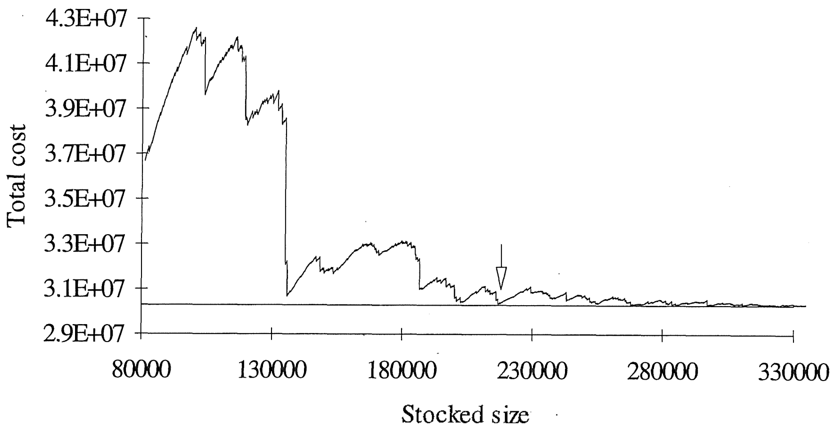

Figure 4. Objective function when using Gilmore and Gomory data

Width (cm)	Height (cm)	Quantity
900	380	30
800	510	30
1870	600	60
1340	600	30
920	410	60
900	38	58

Table 2. Demanded sizes for example PINHO30

is seldom of a much higher magnitude than the sizes of the demanded parts. Because of that, the working size is usually in the more mountainous area of the graph, i.e., in the region where the waste could be higher if precautions are not taken in the choice of a good assortment.

The examples discussed until now have assumed that only one stock size is selected but in many examples the assortment is composed of several sizes. If only material cost is taken into account, any assortment of two sizes a an b will have a lower or equal cost than the lowest of the costs obtained if size a or size b are considered alone. The cost function of an assortment with n sizes provides an upper bound of the cost functions obtained with assortments composed of more than n sizes. The more sizes we are allowed to have, the closer to zero waste we can get!

Figure 6 shows the costs for problem AÇOS1 when two stock sizes are considered. Two areas can still be identified and a diagonal cordillera can be seen, corresponding to the solutions where the two sizes are similar, the peak values corresponding to the values found in Figure 3.

In Figure 6, one can identify more flat areas corresponding to bigger but still different stock sizes, where a high density of local minima exists. In these areas it seems almost if any search procedure would find solutions that are 'relatively' close to the optimum. Here it would be necessary to choose a poor search strategy to end up in an assortment with 1% more waste than the optimal! In some problems the situation could be more extreme, for example in a small problem named EIL60-1 after data collected in Wäsher and Gau (1993) and shown in Borges (1994), where most solutions lie in an interval of about 3‰ from the optimal, when allowing for two stock sizes. From an optimisation point of view, heuristics whose stopping rules rely on a comparison between the best solution found so far and a lower bound (see for example Yanasse, 1994) demand some knowledge about the problem in order to chose an adequate value for the admissible error. This kind of observations shows the importance of the study of the objective function to allow the comparison of solution procedures. The global optimum can still be difficult to find and, in many cases, one must work in the more mountainous zone of the cost functions, as in many 2/2 examples.

5. Meta-heuristics

It is theoretically possible to extensively calculate the cost of all feasible assortments if software to plan the cutting operation is available; this was done in Section 4. However, such a solution is impracticable because of the time it demands, particularly if multiple stock sizes are considered.

There are many factors that complicate real problems; the existence of different cost structures, non-linearities, suppliers' behaviour or specific costs per cutting pattern. This discourages, at an early stage, the investment in problem-specific methods. For that reason we chose to use two well-known general search procedures, which have gained popularity in the last decade, namely *simulated annealing* and *tabu search*. It is not our intention in this article to tune the methods to a particular problem, but simply to compare the different approaches with some fixed parameter sets, which were believed to be adequate for the problem we address, as would happen in a real application.

The search procedures we will compare are:

- simulated annealing;
- descent methods;
- tabu search;
- a variant of tabu search.

All these methods take an assortment as their starting point. From that solution, described as a set of one or two-dimensional stock sizes, the method will iterate, moving to a new solution in each iteration until the process stops.

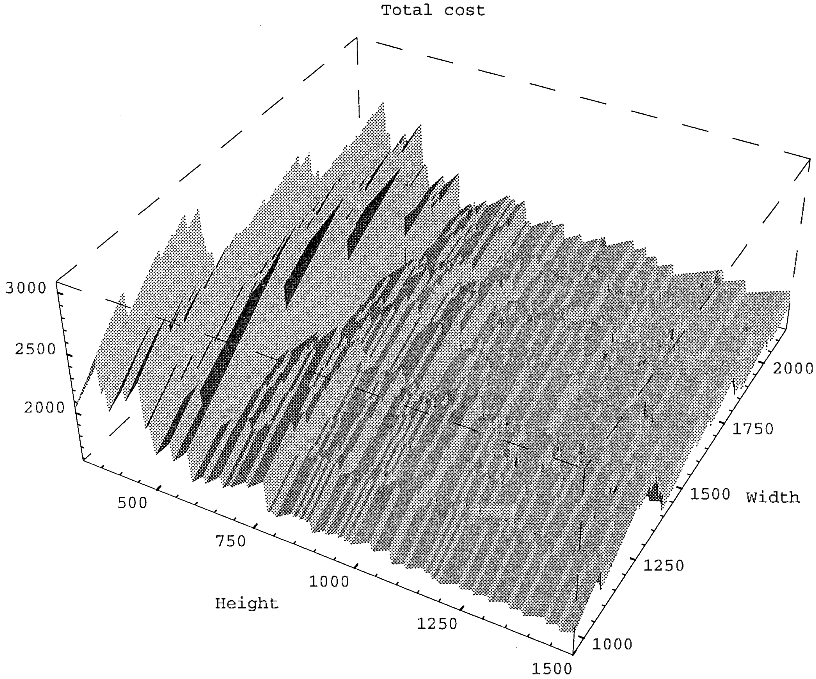

Figure 5. Cost function for example PINHO30 with one stock size

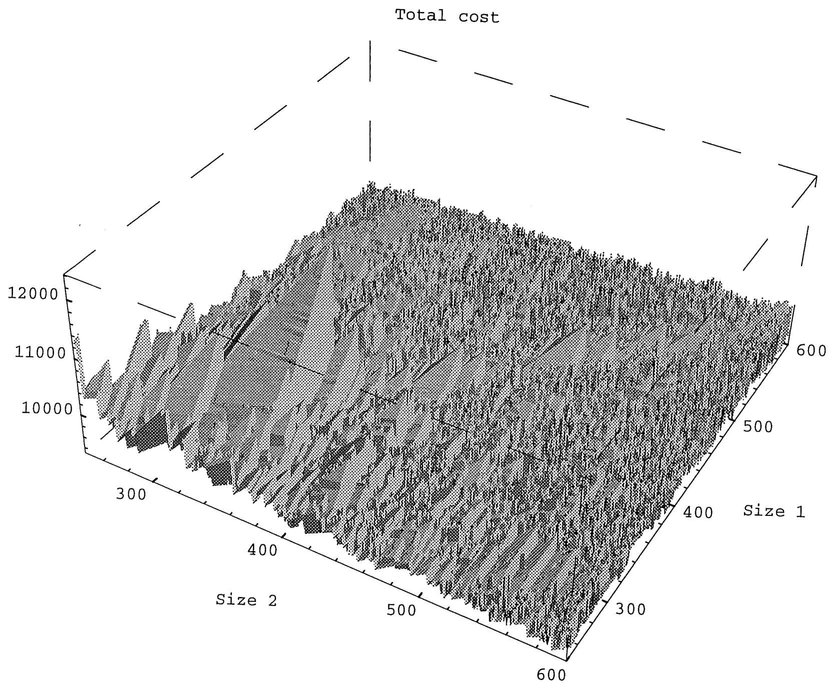

Figure 6. Total cost for example AÇOS1, with two stock sizes

The assortment obtained in each step is called the current solution and can be 'modified' to a number of other assortments we call neighbours of the current solution.

The performance of these search procedures is influenced by many factors; the number of neighbours considered in each iteration, the way those neighbours are generated and the cost of evaluating a solution (in time or other resources). The time required to evaluate each solution can be long in assortment problems since one cutting stock problem must be solved to evaluate each solution. This means that the kind of cutting operation is significant.

As the methods are well known (see for instance Pirlot, 1992) we describe the implemented procedures only briefly:

5.1. Descent Methods

In descent methods, each successive current solution must have a lower cost than the precedent. In our implementation a greed parameter was introduced (named GREED in sequel). While searching the neighbourhood of the current solution, if the number of already searched neighbours that would lead to a cost improvement is equal to GREED, then the selection of a new current solution is forced without searching the current neighbourhood any further. If the GREED parameter is equal to one, the procedure behaves as a greedy algorithm and if set to infinity, the procedure becomes a steepest descent method, searching the entire neighbourhood before moving on.

5.2. Tabu Search

Tabu search includes mechanisms to overcome being trapped in local minima and to avoid cycling. To 'step out' of local minima it allows adopting the best neighbour, even if that leads to a cost degradation. To avoid cycling, a tabu condition is attributed to the most recently visited solutions so that moving to a solution with tabu status is prohibited. In practice this can be done by placing the last visited solutions in a tabu list that is looked up when generating neighbours. There are many variants of this method.

In the case of assortment problems, it was easy and efficient to keep a tabu list with the complete solutions and so the basic version of the search procedure was implemented. As a stopping criterion, we set a limit for the number of iterations without improving the best solution so far. A greed parameter with the same role as in descent methods was also introduced.

5.3. Tabu Search with infinite list of calculated solutions

Taking advantage of the existing code, we can efficiently store and search for all assortments we calculate along the way. This avoids solving the same cutting stock problems more than once and also allows simulating a tabu list of infinite length. We made the search even more restrictive by assigning tabu conditions to all calculated solutions, including the ones that were never selected to be 'current solutions'. In each iteration, if a neighbour was calculated but not chosen to be the new current solution since a better solution existed 'near' it, then that worse solution is also assigned a tabu status. This procedure results in pushing the search forward to unexplored areas. Again the greed parameter was introduced.

5.4. Simulated Annealing

We do not describe simulated annealing; descriptions of this heuristic be found in Pirlot (1992) or Kirkpatrick, Gelatt and Vecchi (1983).

From the many possible cooling strategies we chose a convex quadratic decreasing function as suggested in Andersen (1993). The only reason for this choice was that it achieved better results while solving the problem considered in that paper in comparison with other cooling strategies.

The temperature C_k in each iteration K is given by:

$$C_k = aK^2 + bK + c$$

Where the parameters a, b, and c are calculated as follows:

$$a = \frac{C_0 - C_F}{(\max I)^2} \quad b = \frac{2(C_f - C_0)}{\max I} \quad c = C_0$$

The maximum number of solutions to search ('stopping rule') is represented by '$\max I$'. The parameters C_0 and C_F (initial and final temperatures) were, as in Andersen (1993), determined after an initial phase where a few solutions were tested. We chose to test three different temperature values that we think are reasonable and cover the range the thumb-rule used by Andersen would suggest.

The simulated annealing procedure can be written in the following way:

Function *Simulated annealing:*
$x \leftarrow j$ // Current solution x is initialised with initial solution J
$a \leftarrow (c0 - cf)/(\max I^2)$
$b \leftarrow 2(cf - c0)/\max I$
$c \leftarrow c0$
iteration $\leftarrow 0$
do
    do
        temperature $\leftarrow a \cdot \text{iteration}^2 + b \cdot \text{iteration} + c$
        iteration $\leftarrow$ iteration $+ 1$
        $t =$ neighbour(x) // Random sequence without repetition.
        will_change $\leftarrow 1$
        if cost_of_solution(t) $>$ cost_of_solution(x) and random$[0, 1] >$

```
                exp(difference_cost_between_solutions(t, x)/temperature)
            will_change ← 0
    while there_are_more_neighbors(x) and will_change = 0
    x ← t
    if global_improvement()
            best_x ← x
while will_change = 1 and iteration < max I
return best_x
```

5.5. Neighbour generation

To evaluate the importance of the neighbour generation we tested the same examples with two different ways of generating neighbours, one being an enhancement of the other. We call the way neighbours are generated the *neighbourhood structure.*

The first neighbourhood structure is very simple: a neighbour is generated by increasing or decreasing one of the current assortment parts' size in one of its dimensions by a step that could be equal either to the highest common factor of the measures of the demanded sizes in that dimension or to the smallest value observed in that dimension.

In the second neighbourhood structure, the solution parts' sizes can also be modified in each dimension by the highest demanded value in that dimension or by the size of the most demanded part. This second structure allows bigger jumps and the use of the modal measures in the demand can also be seen as an intelligent attempt to explore the distance between 'good' sizes, which is eventually more frequent.

Section 6 presents numerical tests and results using both neighbourhood structures.

6. Numerical results

We tested the meta-heuristics mentioned above with several parameterizations, resulting in a total of 13 particular implementations. The parameters used are indicated, together with the designation we will use in the rest of the paper to refer to each of the particular heuristics:

- Descent methods (D)
  - D1 GREED $=\infty$ Steepest descent
  - D2 GREED $=1$ Greedy algorithm
  - D3 GREED $=3$
- Tabu search (TS)
  - TS1 GREED $=\infty$
  - TS2 GREED $=1$ Greedy tabu search

D1	D2	D3	TS1	TS2	TS3	kTS1	kTS2	kTS3	SA1	SA2	SA3	SA4
10	8	10	52	36	136	297	33	198	183	153	115	309
0.83%	0.67%	0.83%	4.33%	3.00%	11.33%	24.75%	2.75%	16.50%	15.25%	12.75%	9.58%	25.75%

Table 3. Frequencies of being the best among the 13 heuristic (1200 initial assortments)

  - TS3 GREED $= 3$
- Variant of tabu search (kTS)
  - kTS1 GREED $= \infty$
  - kTS2 GREED $= 1$
  - kTS3 GREED $= 3$
- Simulated Annealing (SA)
  - SA1 $c0 = 2000$ MAXI $= 1500$
  - SA2 $c0 = 1000$ MAXI $= 1500$
  - SA3 $c0 = 300$ MAXI $= 1500$
  - SA4 $c0 = 1500$ MAXI $= 2500$

The size of the tabu list in the TS heuristics is of 7 solutions. In TS and kTS heuristics, the limit for the number of iterations without global improvement was set to 120. In the SA heuristics, the final temperature CF was set to zero.

As a test problem, we use the data presented in Table 1 (AÇOS1) and fix the number of sizes in the assortment to two. The maximum width allowed is 600, which makes the objective function correspond approximately to the one presented in Figure 6. 1200 initial solutions were randomly generated (uniform distribution) and used as starting assortments for the 13 heuristics described above, i.e., the same initial set was used with all the heuristics.

The boxplots in Figure 7 summarise the results distributions. A cross indicates the mean values and small dashes signal the 10th and 90th percentiles.

The mean values lie inside an interval of relatively small amplitude (a little above 1% of the cost values). The SA heuristics behaved well, namely SA4, and show the lower results dispersion. The variant of tabu search seems to behave better than the other TS heuristics. Table 3 presents the number of runs in which each heuristic was the best (i.e.: achieved solutions with lowest cost, including ties) among the thirteen heuristics, when starting from the same initial assortments. Analogously, Table 4 gives the number of runs in which the global optimum was found. Relative frequencies are also given.

Finally, in Figure 8, we compare the effort required by the implemented procedures. Descent methods behave well, especially if the small number of cutting

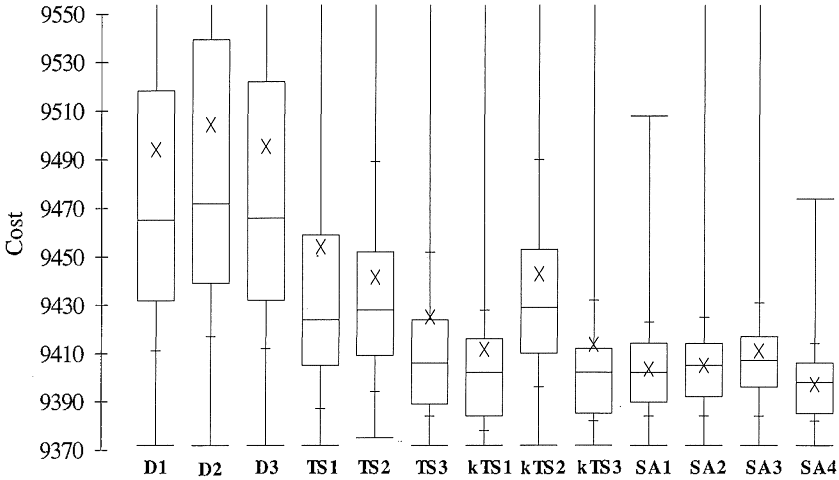

Figure 7. Results with the first neighbourhood structure

D1	D2	D3	TS1	TS2	TS3	kTS1	kTS2	kTS3	SA1	SA2	SA3	SA4
1	3	1	10	0	18	27	2	28	7	9	7	14
0.08%	0.25%	0.08%	0.83%	0.00%	1.50%	2.25%	0.17%	2.33%	0.58%	0.75%	0.58%	1.17%

Table 4. Frequencies of finding global optima (1200 initial assortments)

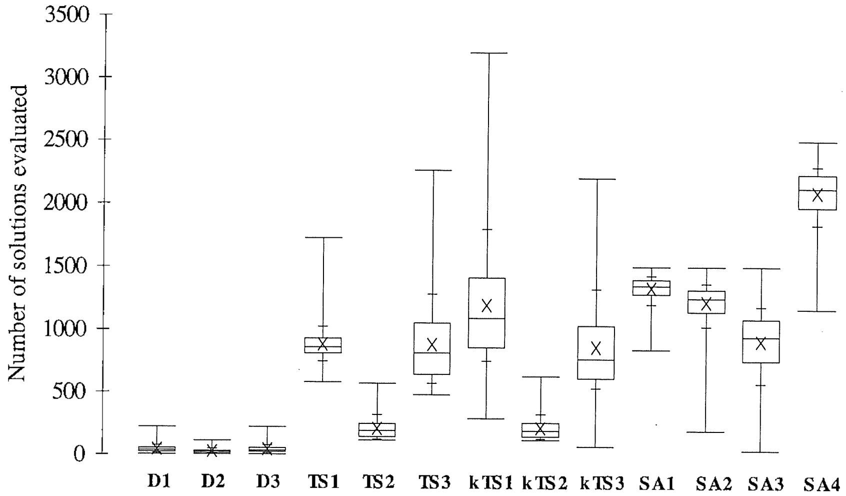

Figure 8. Number of cutting stock problems

stock problems is taken into account. This behaviour is somehow expected after the discussion about the cost functions taken in Section 4. It would be of interest to experiment with descent methods with multiple initial solutions and choosing only the best results. Could it be preferable? That issue is discussed further on.

The costs obtained using the second neighbourhood structure are summarised in Figure 9 while the number of cutting stock problems solved by each implementation are represented in Figure 10. The new results are generally better than the ones obtained employing the first, more simple, neighbourhood structure.

The tabu search heuristic with intermediate greed (TS3) outranks the steepest descent implementation of tabu search (TS1). The kTS3 heuristic also behaves well but it was the steepest descent implementation of the kTS heuristic (kTS1) that found an optimal solution in 21.5% of the runs, as shown in Table 6. This advantage is derived from the stronger effort this implementation demanded (see Figure 10). The simulated annealing implementations reacted modestly to the improvement of the neighbourhood structure; SA4 demanded more effort but yielded better results than SA1, SA2 or SA3.

One conclusion is that the implementations inspired on tabu search improved further when the neighbourhood structure was enriched, in comparison with simulated annealing, where improvements were more modest. It seems that tabu search heuristics took more advantage of the knowledge about the problem that

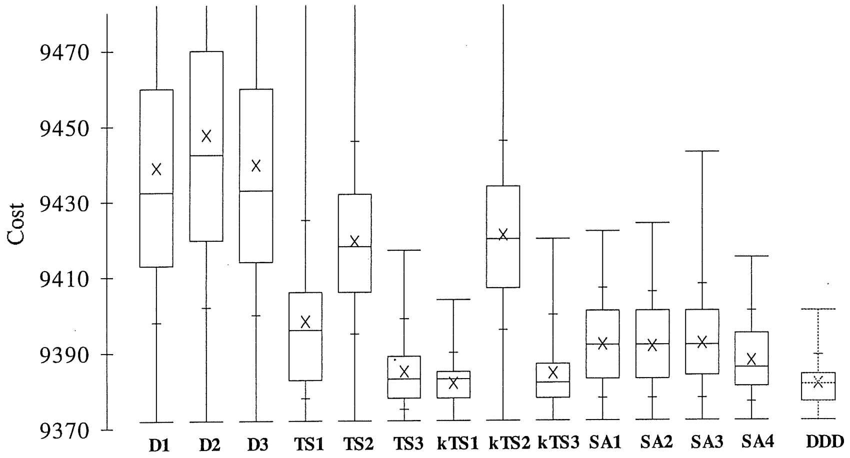

Figure 9. Results using the second neighbourhood structure

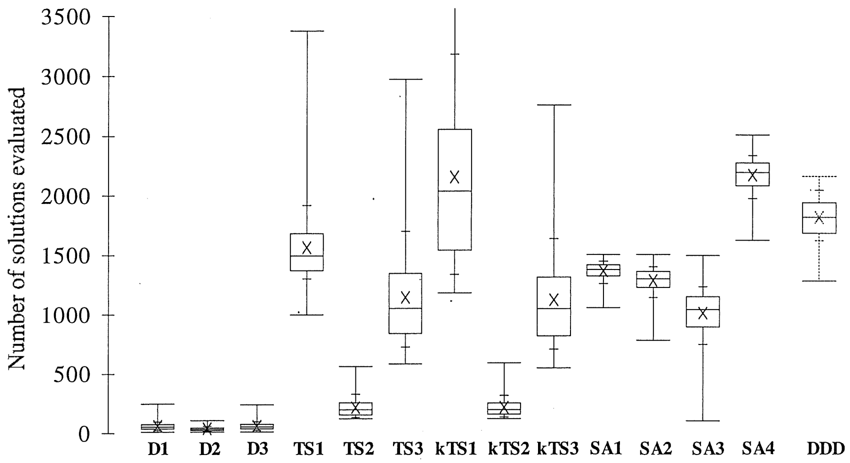

Figure 10. Number of solutions evaluated

D1	D2	D3	TS1	TS2	TS3	kTS1	kTS2	kTS3	SA1	SA2	SA3	SA4
17	14	17	100	9	290	350	6	287	109	125	121	173
1.42%	1.17%	1.42%	8.33%	0.75%	24.17%	29.17%	0.50%	23.92%	9.08%	10.42%	10.08%	14.42%

Table 5. Frequencies: best among the 13 heuristics (second neighbourhood structure)

D1	D2	D3	TS1	TS2	TS3	kTS1	kTS2	kTS3	SA1	SA2	SA3	SA4
9	10	11	32	2	112	258	1	126	33	37	45	51
0.75%	0.83%	0.92%	2.67%	0.17%	9.33%	21.50%	0.08%	10.50%	2.75%	3.08%	3.75%	4.25%

Table 6. Frequencies: global optimum found (second neighbourhood structure)

was introduced in the search process. Tabu search would be placed somewhere between simulated annealing and problem dependent specialised heuristics as far as use of problem knowledge is concerned.

The performance of the descent methods suggested using them with several initial solutions and selecting the best assortment found in a group of runs. We use the same 1200 starting assortments considered in the comparisons above and group them sequentially in groups of 30 elements. Inside each group, the steepest descent algorithm is used and the best result obtained within each group is chosen. This yields 40 final assortments whose statistics are also shown in Figure 9 and Figure 10, under the designation DDD. Care should be taken in comparing those boxplots with the others as they only represent 40 solutions and not 1200. Descent methods do seem to perform well; this is due to the shape of the objective function in this kind of assortment problems (see Figure 6). However TS3 and kTS3 still solve less cutting stock problems for a similar result.

7. Final remarks

The decisions that need to be made for assortment selection have a complementary role to the ones in cutting stock problems. Both problems are relevant to the operation of many industries although assortment selection problems have not received as much attention as cutting stock problems.

The graphical inspection of 1D and 2D cost functions indicates that in many cases, discovering a reasonable assortment is not difficult if an optimal cutting procedure is used, though the optimal assortment might still be difficult to find. There may exist many minima close to the global optimum and still other solutions that impose a considerable amount of unnecessary waste. In these cases, extra care must be taken when comparing solution procedures because the simple cost of the solutions alone might be a poor comparison criterion.

The existence of distinct variants of assortment problems leads to experimentation with meta-heuristic methods instead of investing in the conception of specialised problem dependent heuristic methods for each problem variant. If the neighbourhood structure is poor, simulated annealing might have some advantage but when the neighbourhood structure is enriched with problem dependent knowledge, then the implementations based on tabu search seem to improve more, in particular a variant with infinite list of calculated solutions. Anyway, it was not the intention to tune the heuristics' parameters and the quality of the solutions seems to be quite correlated with the computing effort they demanded. This suggested trying the simple descent method with several initial points; and the obtained results were not worse than the ones produced by other heuristics. This was due to the kind of cost function in the considered assortment problems.

Writing of the present paper was partially supported by grant CIENCIA/BD/2510

References

ANDERSEN, K., VIDAL, R.V. and IVERSEN, V.B. (1993) Design of a communication network using simulated annealing. *Applied Simulated Annealing*, Springler Verlag, 201-215.

BEASLEY, J.E. (1985) An algorithm for the two-dimensional assortment problem. *European Journal of Operational Research*, **19**, 253-261.

BORGES, P. and FERREIRA, J.S. (1994) Overview of assortment problems. *Proceedings the Third Meeting of the Nordic Section of The Mathematical Programming Society.*

BORGES, P. (1994) *Assortment Selection - Characterisation and Solutions*, MSc dissertation, Faculdade de Engenharia, Porto University (in Portuguese).

CHAMBERS, M.L. and DYSON, R.G. (1976) The cutting stock problem in the flat glass industry - selection of stock sizes. *Operational Research Quartely*, **27**, 4, 949-957.

CHEN, C., SARIN, S. and RAM, B. (1993) A mixed-integer programming model for a class of assortment problems. *European Journal of Operational Research*, **63**, 362-367.

DIEGEL, A. and BOCKER, H.J. (1984) Optimal dimensions of virgin stock in cutting glass to order. *Decision Sciences*, **15**, 260-274.

DOWSLAND, K.A. and DOWSLAND, W.B. (1992) Packing problems. *European Journal of Operational Research*, **56**, 2-14.

DYCKHOFF, H. (1990) A typology of cutting and packing problems. it European Journal of Operational Research, **44**, 145-159.

EILON, S. (1960) Optimizing the shearing of steel bars. *Journal Mechanical Engineering Science*, **2**, 129-142.

GEMMILL, D.D. and SANDERS, J.L. (1990) Approximate solutions for the cutting stock 'portfolio' problem. *European Journal of Operational Research*, **44**, 167-174.

GEMMILL, D.D. and SANDERS, J.L. (1991) A comparison of solution methods for the assortment problem. *International Journal of Production Research*, **29**, 12, 2521-2527.

GILMORE, P.C. and GOMORY, R.E. (1963) A linear programming approach to the cutting stock problem part II. *Operations Research*, **11**, 863-888.

GILMORE, P.C., GOMORY, R.E. (1965) Multistage cutting stock problems of two and more dimensions. *Operations Research*, **13**, 94-120.

HINXMAN, A.I. (1980) The trim-loss and assortment problems: a survey. *European Journal of Operational Research*, **5**, 8-18.

KIRKPATRICK, S., GELATT, C.D., VECCHI, M.P. (1983) Optimization by simulated annealing. *Science*, **220**, 671-680.

MARCOTTE, O. (1985) The cutting stock problem and integer rounding. *Mathematical Programming*, **33**, 82-92.

PENTICO, D.W. (1971) *Extensions of the Assortment Problem*, Ph.D. Thesis, Carnegie-Mellon University, Pittsburgh, Pa.

PENTICO, D.W. (1986) Comments on 'On the optimal choice of sizes' by Peter Tryfos. *Operations Research*, **34**, 2, 328-329.

PIRLOT, M. (1992) General local search heuristics in combinatorial optimisation: a tutorial. *Belgian Journal of Operations Research*, Statistics and Computer Science, **32**, 1-2, 7-68.

OLIVEIRA, J.F. and FERREIRA, J.S. (1994) A faster variant of the Gilmore and Gomory technique for cutting stock problems. *JORBEL*, **34**, 1, 23-38.

WÄSCHER, G. and GAU, T. (1993) *Test data for the one-dimensional cutting stock problem*, Arbeitsbericht des Instituts für Wirtschaftswissenschaften der Technischen Universität Braunschweig, Nr. 93/9.

VIDAL, R.V.V. (1994) On the optimal sizing problem. *Journal of the Operational Research Society*, **45**, 6, 714-719.

YANASSE, H., ZINOBER, A. and HARRIS, H. (1994) A search strategy for the one-size assortment problem. *European Journal of Operational Research*, **74**, 135-142.