The G4-Heuristic for the Pallet Loading Problem

Author(s): Guntram Scheithauer and Johannes Terno

Source: *The Journal of the Operational Research Society*, Apr., 1996, Vol. 47, No. 4 (Apr., 1996), pp. 511-522

Published by: Palgrave Macmillan Journals on behalf of the Operational Research Society

Stable URL: https://www.jstor.org/stable/3010727

Journal of the Operational Research Society (1996) 47, 511–522  0160-5682/96 $12.00

# The G4-Heuristic for the Pallet Loading Problem

GUNTRAM SCHEITHAUER and JOHANNES TERNO
Technical University Dresden, Germany

A new heuristic for the well-known (two-dimensional orthogonal) pallet loading problem (PLP) is proposed in this paper. This heuristic, referred to as G4-heuristic, is based on the definition of the so-called G4-structure of packing patterns. The G4-structure is a generalization of the common used block structure of packing patterns which requires the same orientation of packed boxes within each block. The G4-heuristic yields in approximately 99% of the test instances an optimal solution and solves all instances exactly where at most 50 boxes are contained in an optimal packing. Although the algorithm is pseudo-polynomial the computational experiments reported show that also instances with more than 200 packed boxes in an optimal solution can be handled with a small amount of computational time. Moreover, so far there is not known any instance where the gap between optimal value and the value obtained by the G4-heuristic is larger than one box.

*Key words*: packing, pallet loading problem, cutting stock problem, combinatorial analysis, dynamic programming, heuristics

## INTRODUCTION

The paper deals with the (standard orthogonal) two-dimensional pallet loading problem (PLP) of finding a maximal layout for identical small rectangles (pieces, boxes) on a larger rectangle (pallet). This problem often arises in distribution when many items of a small product must be transported. An increase in the number of items on a pallet leads directly to a decrease in transportation cost. For that reason the PLP is of considerable economic importance.

Therefore a lot of work dealing with the PLP is published (cf. Refs 1, 2). A comprehensive survey of the state of the art of handling PLP is given in Ref. 3. Since so far there is not known any efficient exact algorithm for the PLP most of the work is connected with heuristic approaches. Using the classification of Dyckhoff[4] the PLP belongs to the class 2/B/O/C.

Although there exist some attempts to develop tight (upper) bounds (cf. Refs 5–8) unfortunately there exists no general useful criterion to decide whether a packing pattern (layout, packing arrangement) is optimal or not. Therefore, already in cases of instances having no more than 100 items in an optimal packing, the computational amount of a branch and bound algorithm for solving an instance may be too large in practice (cf. Ref. 3). This is because of the very rapid increasing number of combinatorial opportunities of different packings. Investigations with respect to equivalence and dominance of packing patterns can be found in Refs 3 and 9 which are useful to reduce the computational effort in a branch and bound algorithm.

In this paper we will analyse the structure of optimal packing patterns and we will introduce the notation of the G4-structure. Then we propose a new heuristic for solving the PLP. Since the algorithm is based on dynamic programming it is of pseudo-polynomial type. The computational experiments show that instances having more than 200 items in an (optimal) packing pattern can be handled with a small computational amount of time.

## PROBLEM FORMULATION AND NOTATIONS

The PLP may be defined more formally as follows: Given a pallet with length $L$ and width $W$ and a rectangular box (piece) with length $l$ and width $w$, how many boxes can be packed orthogonally on the pallet?

*Correspondence: J. Terno, Technische Universität Dresden, Institut für Numerische Mathematik, TU Dresden, D-01062, Dresden, Germany*

Throughout the paper we will denote an instance of the PLP by a quadruple $(L, W, l, w)$. Without loss of generality we may assume that $L$, $W$, $l$ and $w$ are positive integers and that $L \geqslant W \geqslant l > w$. Moreover we will assume that the greatest common divisor $q$ of $l$ and $w$ equals 1 since in the other case the instance $(L, W, l, w)$ can be reduced to $([L/q], [W/q], l/q, w/q)$ where $[x]$ gives the greatest integer not greater than $x$.

A horizontally oriented box will be referred to as H-box, and a vertically oriented box as V-box. Positions of the boxes will be given with respect to an $(x, y)$-cartesian system whose origin is placed at the lower left corner of the pallet. Moreover we identify the $x$-direction with the length direction and the $y$-direction with the width direction. Hence, the packing of the $i$th box occupies a rectangular area in the plane. For the sake of simplicity we use the following identifications:

$$(x_i, y_i, H) = \{(x, y) \in \mathscr{R}^2 : x_i \leqslant x \leqslant x_i + l,\ y_i \leqslant y \leqslant y_i + w\}$$

and

$$(x_i, y_i, V) = \{(x, y) \in \mathscr{R}^2 : x_i \leqslant x \leqslant x_i + w,\ y_i \leqslant y \leqslant y_i + l\}.$$

Then a packing pattern $A$ of $n$ packed boxes may be described in the form of a set

$$A = \{(x_i, y_i, o_i) : i = 1, \ldots, n\}$$

where $(x_i, y_i)$ denotes the so-called allocation point (lower left corner) of the $i$th packed box and $o_i \in \{H, V\}$ characterizes its orientation.

Clearly, the packing pattern $A$ is feasible for the pallet $L \times W$ if and only if the following conditions are fulfilled (here and in the following let int($S$) denote the interior of set $S$):

$$\operatorname{int}(x_i, y_i, o_i) \cap \operatorname{int}(x_j, y_j, o_j) = \varnothing \qquad \text{for } i, j = 1, \ldots, n,\ i \neq j,$$

$$(x_i, y_i, o_i) \subset P(L, W) \qquad \text{for } i = 1, \ldots, n,$$

where $P(L, W) = \{(x, y) \in \mathscr{R}^2 : 0 \leqslant x \leqslant L, 0 \leqslant y \leqslant W\}$.

## THE G4-STRUCTURE

In order to introduce the G4-structure of a packing pattern we need to define some more notations. Let $A = \{(x_i, y_i, o_i) : i = 1, \ldots, n\}$ be a packing pattern of the pallet $P(L, W)$. For simplicity in the following considerations we define the length $l_i$ and width $w_i$ of the $i$th packed box as follows:

$$l_i := \begin{cases} l, & \text{if } o_i = H, \\ w, & \text{if } o_i = V, \end{cases} \qquad w_i := \begin{cases} w, & \text{if } o_i = H, \\ l, & \text{if } o_i = V. \end{cases}$$

A subset $\tilde{A} = A(\tilde{I}) = \{(x_i, y_i, o_i) : i \in \tilde{I}\}$ of $A$ is called a block pattern if there exists a subset $\tilde{I}$ of $I = \{1, \ldots, n\}$ and a rectangle

$$R(\underline{x}, \underline{y}, \bar{x}, \bar{y}) := \{(x, y) \in \mathscr{R}^2 : \underline{x} \leqslant x \leqslant \bar{x},\ \underline{y} \leqslant y \leqslant \bar{y}\}$$

such that $\bigcup_{i \in \tilde{I}} (x_i, y_i, o_i) \subset R(\underline{x}, \underline{y}, \bar{x}, \bar{y})$ and $\bigcup_{i \in I \setminus \tilde{I}} (x_i, y_i, o_i) \cap \operatorname{int} R(\underline{x}, \underline{y}, \bar{x}, \bar{y}) = \varnothing$.

If $\tilde{A}$ is a block pattern then the smallest containing rectangle is given by

$$\tilde{R}(\tilde{A}) := \left\{(x, y) : \min_{i \in \tilde{I}} x_i \leqslant x \leqslant \max_{i \in \tilde{I}} \bar{x}_i,\ \min_{i \in \tilde{I}} y_i \leqslant y \leqslant \max_{i \in \tilde{I}} \bar{y}_i\right\}$$

where $\bar{x}_i = x_i + l_i$ and $\bar{y}_i = y_i + w_i$.

The block length $L(A(\tilde{I}))$ and the block width $W(A(\tilde{I}))$ are defined as follows:

$$L(A(\tilde{I})) := \max_{i \in \tilde{I}} \bar{x}_i - \min_{i \in \tilde{I}} x_i, \qquad W(A(\tilde{I})) := \max_{i \in \tilde{I}} \bar{y}_i - \min_{i \in \tilde{I}} y_i.$$

When all of the packed boxes of a packing pattern $A = \{(x_i, y_i, o_i) : i = 1, \ldots, n\}$ have the same orientation we call $A$ a homogeneous (packing) pattern (cf. Ref. 10). Trivially the given packing pattern itself is a block pattern.

In order to characterize the structure of a packing pattern the following definitions will be introduced.

*Definition 1*

A packing pattern $A = \{(x_i, y_i, o_i) : i = 1, \ldots, n\}$ of a pallet $P(L, W)$ is said to be of guillotine structure (G-structure) if $n$ is not greater than 3 or if there exist a partition $I_1$ and $I_2 = I \backslash I_1$ of $I = \{1, \ldots, n\}$ such that both $A(I_1)$ and $A(I_2)$ form block patterns which are of G-structure.

*Definition 2*

A homogeneous pattern is said to be of 1-block structure. For $k \geqslant 2$, a packing pattern $A = \{(x_i, y_i, o_i)\colon i = 1, \ldots, n\}$ is said to be of $k$-block structure if there exists a partition of $I = \{1, \ldots, n\}$ in $q$ disjunctive subsets $I_j, j = 1, \ldots, q, q \leqslant k$, such that each packing pattern $A(I_j)$ forms a pattern of $p_j$-block structure with $p_j \leqslant k$.

According to this definition the set of packing patterns having $k$-block structure contains any packing pattern with a $p$-block structure for $p \leqslant k$.

Note, there is an important difference between the definition of patterns having a $k$-block structure and the $k$-block patterns used in the literature (cf. Ref. 8) since for the latter it is required that each of the corresponding $k$ block patterns consists of a homogeneous pattern.

Per definition, a packing pattern of a pallet $P(L, W)$ with $n$ boxes is of $n$-block structure. But the question is: what is the smallest $k$ for a given pattern $A$ such that $A$ is of $k$-block structure? Moreover, the following problem is unsolved so far: Given an instance $(L, W, l, w)$ of the PLP, what is the smallest $k$ such that the set of packing patterns having $k$-block structure contains an optimal pattern?

In the following we will consider a set of instances of the PLP (referred to as problem of type I (cf. Ref. 3), characterized by $1 \leqslant L/W \leqslant 2$, $1 \leqslant l/w \leqslant 4$, $1 \leqslant LW/(lw) < 51$) which is of large practical importance and we will show that for any instance of type I there exists an optimal packing pattern having G4-structure.

At first we state the following:

*Assertion 1*

A packing pattern $A = \{(x_i, y_i, o_i) : i = 1, \ldots, n\}$ has a $k$-block structure with $k \leqslant 3$ if and only if the pattern is of G-structure.

To see this one has only to make use of the recursive definition of G-structure and 3-block structure.

*Definition 3*

A packing pattern $A = \{(x_i, y_i, o_i) : i = 1, \ldots, n\}$ of a pallet $P(L, W)$ is said to have the G4-structure if $A$ has $k$-block structure with a $k \leqslant 4$.

The notation 'G4'-structure should be interpreted as guillotine- or 4-block structure or as generalized 4-block patterns (see Figure 1). Note, the recursive definition of the G4-structure allows an arbitrary number of blocks for a packing pattern $A$ with G4-structure.

In order to compare packing patterns of G4-structure (referred to also as G4-patterns) with those obtained by the common used heuristics we refer always to the excellent and comprehensive work of J. Nelissen[3] for the definition and computational specifications of the considered heuristics.

Obviously, any $k$-block pattern with $k \leqslant 4$ has G4-structure. Moreover, it is easy to see that any 5-block pattern also has G4-structure.

The 7-block patterns obtained by a 7-block heuristic (proposed by Dowsland and Dowsland[11] and Exeler[12]) have G4-structure. To see this we follow Ref. 3, p. 35: The generation of the 7-block

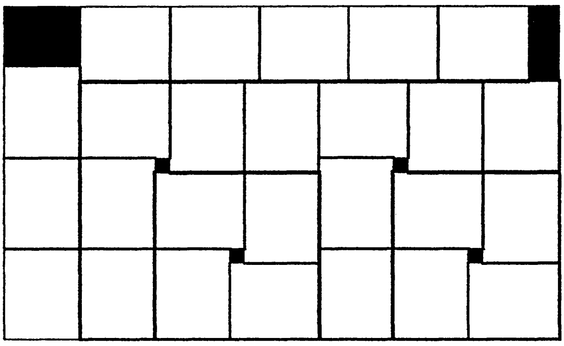

Fig. 1. *G4-structure.*

patterns leads especially to three homogeneous blocks in the lower left, upper left and upper right corners of the pallet and a remaining rectangular area in the lower right corner which is packed with a 4-block pattern. Hence, always a packing pattern with G4-structure is obtained.

The 9-block heuristic developed by Exeler[12] also yields packing patterns of G4-structure since in addition to the 7-block heuristic only two further homogeneous blocks in the form of guillotine-blocks are allowed.

It is also easy to see (by inspecting the examples given[3] and a more formal analysis of the algorithmic approaches) that the so-called diagonal heuristics proposed by Exeler[12] and Naujoks[13] yield packing patterns having G4-structure.

Nelissen[3,8] proposes the so-called angle heuristic, the recursive angle heuristic and the recurrence heuristic. These heuristics also generate packing patterns of G4-structure as a result of their definition. The main difference in comparison to the general G4-pattern is that various blocks in the heuristics are restricted to be homogeneous.

Last but not least, the so-called complex block heuristic proposed[8] also generates packing patterns of G4-structure because of its similarity to the 7-block heuristic.

Summarizing, any packing pattern obtained by one of the heuristics considered above, belongs to the set of G4-patterns. Hence, if we compute the best G4-pattern for a given instance of the PLP then we get a pattern at least as good as the best pattern which can be obtained by the other heuristics.

## THE NEW HEURISTIC

In this section we give a detailed description to compute a packing pattern having G4-structure with a maximal number of boxes (maximal G4-pattern).

### *The basic alogorithm*

Because of the definition of the G4-structure, a G4-pattern consists of either two disjunctive G4-patterns (if a guillotine-(G-)cut occurs) or of four disjunctive G4-patterns (if there is no G-cut). In order to compute maximal G4-patterns we need maximal G4-patterns for smaller pallets (rectangles).

Let $n(L', W')$ denote the number of packed boxes (with length $l$ and width $w$) of a maximal G4-pattern for a pallet with length $L'$ and width $W'$ ($0 \leqslant L' \leqslant L$, $0 \leqslant W' \leqslant W$).

Using the notations in Figure 2 with

$$e := L' - a, \quad f := W' - d, \quad g := L' - c, \quad h := W' - b,$$

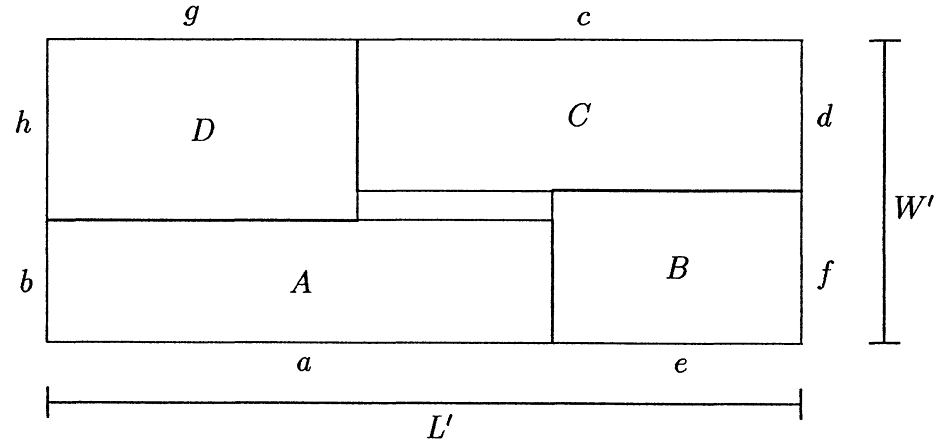


FIG. 2. *Notations for the recurrence formulas.*

the following recurrence formula is valid:

$$n(L', W') = \max\Bigg\{ \max_{1 \leqslant a \leqslant L'/2} n_V(L', W', a), \max_{1 \leqslant b \leqslant W'/2} n_H(L', W', b),$$

$$\max_{1 \leqslant a < L'} \max_{1 \leqslant b < W'} \max_{L'-a < c < L'} \max_{1 \leqslant d < W'-b} \{n(a, b) + n(e, f) + n(c, d) + n(g, h)\}\Bigg\}, \quad (1)$$

$$1 \leqslant W' \leqslant W, \qquad 1 \leqslant L' \leqslant L,$$

$$n_V(L', W', a) := n(a, W') + n(L' - a, W'),$$

$$n_H(L', W', b) := n(L', b) + n(L', W' - b),$$

$$n(0, b) := 0, \quad 0 \leqslant b \leqslant W, \quad n(a, 0) := 0, \quad 1 \leqslant a \leqslant L, \quad n(l, w) := n(w, l) := 1.$$

Now the heuristic in its basic version simply consists of calculating $n(L', W')$ for all $L'$ and $W'$ with $0 \leqslant W' \leqslant W$ and $0 \leqslant L' \leqslant L$. But the recurrence formula and the basic algorithm involve some opportunities to reduce the computational amount.

Firstly, note that it is not necessary to consider G4-patterns with $b + d > W'$ (the rectangular waste is not above but right of block $A$) since such G4-patterns are reflections of packings of the other type with respect to the width direction.

Using the ideas of normalization (cf. Herz[10]) and raster points (cf. Ref. 14) we define the set

$$S(L) = S(L, l, w) = \{r : r = \alpha l + \beta w, r \leqslant L, \alpha, \beta \in \mathscr{Z}_+\}$$

of potential $x$-coordinates of an allocation point. Moreover, using

$$\langle s \rangle_L := \max\{r \in S(L) : r \leqslant s\}$$

the reduced set $\tilde{S}(L)$ of raster points is given by

$$\tilde{S}(L) = \tilde{S}(L, l, w) = \{\langle L - r \rangle_L : r \in S(L)\}$$

(cf. Ref. 14). The usage of $\tilde{S}(L)$ realizes the application of the Nicholson principle of Dynamic Programming and is based on the fact that any $r \in S(L)$ for which an $r' \in S(L)$ exists with $r < r'$ and $\langle r \rangle_L = \langle r' \rangle_L$ can be omitted because of dominance considerations.

Analogously $S(W) = S(W, l, w)$ and $\tilde{S}(W) = \tilde{S}(W, l, w)$ can be defined.

Note, that the definition of $\tilde{S}$ leads in general to a smaller set in comparison to the approach of K. Dowsland (cf. Ref. 3, p. 45, e.g. instance (65, 65, 13, 11)).

Using the raster point sets $S(L)$ and $S(W)$ with

$$e := \langle L' - a \rangle_L, \quad f := \langle W' - d \rangle_W, \quad g := \langle L' - c \rangle_L, \quad h := \langle W' - b \rangle_W,$$

we obtain

$$n(L', W') = \left\{ \max_{a \in S_0(L'/2)} n_V(L', W', a), \quad \max_{b \in S_0(W'/2)} n_H(L', W', b), \right.$$

$$\left. \max_{a \in S_a(L')} \max_{b \in S_b(W')} \max_{c \in S_c(L', a)} \max_{d \in S_d(W', b)} n(a, b) + n(e, f) + n(c, d) + n(g, h) \right\}, \quad (2)$$

$$W' \in S(W), \qquad L' \in S(L),$$

where

$$S_0(L) := S(L) \backslash \{0\},$$

$$S_a(L') := \{r \in S(L) : w \leqslant r \leqslant L' - w\}, \quad S_b(W') := \{r \in S(W) : w \leqslant r \leqslant W' - w\},$$

$$S_c(L', a) := \{r \in S(L) : L' - a < r \leqslant L' - w\}, \quad S_d(W', b) := \{r \in S(W) : w \leqslant r < W' - b\}.$$

A further recurrence can be obtained using $\tilde{S}(L)$ and $\tilde{S}(W)$:

$$n(L', W') = \max\left\{ \max_{a \in \tilde{S}_0(L'/2)} n_V(L', W', a), \quad \max_{b \in \tilde{S}_0(W'/2)} n_H(L', W', b), \right.$$

$$\left. \max_{a \in \tilde{S}_a(L')} \max_{b \in \tilde{S}_b(W')} \max_{c \in \tilde{S}_c(L', a)} \max_{d \in \tilde{S}_d(W', b)} n(a, b) + n(e, f) + n(c, d) + n(g, h) \right\}, \quad (3)$$

$$W' \in \tilde{S}(W), \qquad L' \in \tilde{S}(L),$$

where

$$e := \langle L' - a \rangle_L, \quad f := \langle W' - d \rangle_W, \quad g := \langle L' - c \rangle_L, \quad h := \langle W' - b \rangle_W,$$

$$\tilde{S}_0(L') := \{r \in \tilde{S}(L) : 0 < r \leqslant L'\},$$

$$\tilde{S}_a(L') := \{r \in \tilde{S}(L) : w \leqslant r \leqslant L' - w\}, \qquad \tilde{S}_b(W') := \{r \in \tilde{S}(W) : w \leqslant r \leqslant W' - w\},$$

$$\tilde{S}_c(L', a) := \{r \in \tilde{S}(L) : L' - a < r \leqslant L' - w\}, \qquad \tilde{S}_d(W', b) := \{r \in \tilde{S}(W) : w \leqslant r < W' - b\}.$$

Both recursions (2) and (3) have the same structure but the latter works on a reduced set of allocation points. Therefore the recursion (3) is of advantage with respect to computational amount. On the other hand, using (2) an optimal solution is obtained for any pallet $P(L', W')$ with $L' \leqslant L$ and $W' \leqslant W$ when $n(L, W)$ is computed.

Note, when some overlap is allowed, i.e. a pallet $P(L + \Delta L, W + \Delta W)$ may be used, a recurrence similar to (3) may be applied using the following sets

$$\hat{S}(L, \Delta L) := \bigcup_{L \leqslant L' \leqslant L + \Delta L} \tilde{S}(L') \quad \text{and} \quad \hat{S}(W, \Delta W) := \bigcup_{W \leqslant W' \leqslant W + \Delta W} \tilde{S}(W').$$

*Bounds from the literature*

Several upper bounds of the maximal number $n^*(L, W)$ of boxes which can be placed on a pallet $P(L, W)$ without overlap are known. A comprehensive survey of such bounds is contained in Refs 8 and 3; and we refer interested readers to these papers for details.

In order to reduce computational effort we use the following upper bounds:

1. The continuous or area bound $u_c(L, W)$ equals the integer part of the quotient of pallet area to box area.
2. A second bound $u_B(L, W)$ was proposed by Barnes[5]. This bound is based on the consideration that every packing of an $l \times w$ box on an $L \times W$ pallet is also a packing of either an $l \times 1$ or a $1 \times w$ box on the pallet.
3. The upper bound $u_A(L, W)$ is based on the minimization of the area bound on the corresponding equivalence class. Work in this field was done by Dowsland[6] and Exeler[15]. Note, using the development of this bound (cf. Ref. 8) the number of efficient partitions can be reduced essentially by applying of dominance considerations and hence the amount to compute this bound decreases.

4. The fourth bound $u_L(L, W)$ was proposed by Isermann[7] and is based on the solution of a linear programming problem which represents a relaxation of the PLP. Note, the relative large number of variables can easily be handled using a column generation technique.

Since we have a different amount for calculating one of the bounds we use them in a special order as described in the next subsection.

*Bounds for the main loop*

In the following we will discuss several possibilities of using bounds within the recurrence (2) which can be applied similarly to recursion (3).

Firstly, the initialization of the $n$-array can be done in accordance with

$$n(w, W') := \lfloor W'/l \rfloor,\ W' \in S(W), \qquad n(L', w) := \lfloor L'/l \rfloor,\ L' \in S(L).$$

Since the recurrence consists of some parts and since the computations of the different upper bounds require a different amount, the sequence of calculating and bounding is of importance. Therefore the following strategy is used to compute $n(L', W')$ for a pallet $P(L', W')$ with $L' > w$ and $W' > w$. (We assume that $n(p, q)$ has already been computed for all $q \in S(W)$, $p \in S(L' - 1)$ and for $q \in S(W' - 1)$, $p = L'$.):

1. Symmetry: If $W' < L'$ and $L' \leqslant W$ then $n(L', W') := n(W', L')$, Exit (that is, the computation of $n(L', W')$ is terminated).
2. Compute the best homogeneous pattern (let $n_1$ be the number of boxes in this pattern) and the continuous bound $u := u_c(L', W')$. If $n_1 = u$ then Exit.
3. Comparison with best solutions for smaller pallets:
   Set $n_2 := \max\{n(L' - 1, W'), n(L', W' - 1), n_1\}$. If $n_2 = u$ then Exit.
4. Improving the upper bound with Barnes' bound:
   Set $u := \min\{u, u_B(L', W')\}$. If $n_2 = u$ then Exit.
5. Guillotine structure: Compute
   $$n_3 := \max\left\{n_2,\ \max_{a \in S_0(L'/2)} n_V(L', W', a),\ \max_{b \in S_0(W'/2)} n_H(L', W', b)\right\}.$$
   If $n_3 = u$ then Exit.
6. Improving the upper bound with other bounds:
   Set $u := \min\{u, u_A(L', W')\}$. If $n_3 = u$ then Exit.
   Set $u := \min\{u, u_L(L', W')\}$. If $n_3 = u$ then Exit.
7. 4-block structure: Calculate the maximal number $n_4$ of boxes for patterns having 4-block structure.

Note, during the computation of $n_3$ and $n_4$, respectively, it is of advantage to test for Exit when an improved value is calculated.

Since the calculation of $n_4$ requires the variation of four parameters, possibilities to reduce the range of one or several parameters is of advantage. For the following investigations we underlay a consideration of the parameters $a$, $b$ and $c$ in an increasing order and of $d$ in a decreasing order. Let

$$\langle s \rangle_L^+ := \min\{r \in S(L) : r > s].$$

Consider the situation in Figure 3 where $e := \langle L' - \alpha \rangle_L$. If

$$\lfloor (L'W' - w(L' - a - e) - (\langle L' - a \rangle_L^+ - e))/(lw) \rfloor \leqslant n_3$$

then no improved packing can be found for that $a$. Hence, continue with the next $a \in S_a(L')$ if exists.

Consider the situation in Figure 4 where additionally $b$ is fixed and $h := \langle W' - b \rangle_W$. Since when $d = h$ a G-cut occurs we may restrict $d$ to be not greater than $\langle h - 1 \rangle_W$. Hence, if

$$\lfloor (L'W' - b(L' - a - e) - (\langle L' - a \rangle_L^+ - e))/(lw) \rfloor \leqslant n_3$$

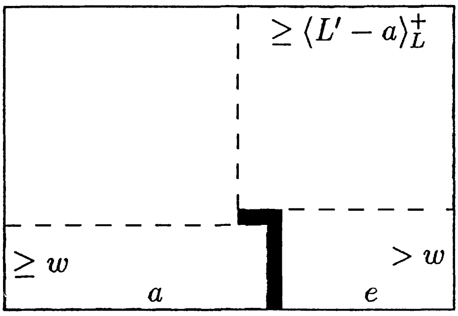


FIG. 3. *Termination criterion in the a-loop.*

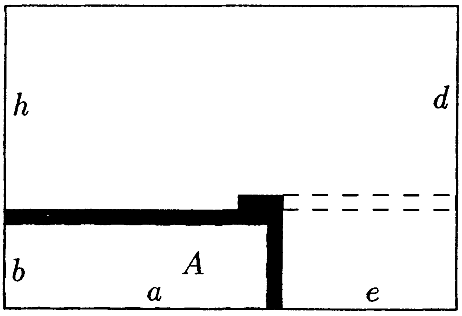


FIG. 4. *Termination criterion in the b-loop.*

then no improved packing can be found for that $a$ and $b' \geqslant b$. Continue with the next $a \in S_a(L')$ if one exists.

If

$$n(a, b) + \lfloor (L'W' - (L' - e)(W' - h) - (\langle L' - a \rangle_L^+ - e)(h - \langle h - 1 \rangle_W))/(lw) \rfloor \leqslant n_3$$

then no improved packing can be found for that $a$ and $b$. Continue with the next $b \in S_b(W')$ if exists.

Consider the situation in Figure 5 where additionally $c$ is fixed and $g := \langle L' - c \rangle_L$. Let $\Delta P := L'W' - (L' - e)(W' - h)$. If

$$n(a, b) + \lfloor (\Delta P - (c - e)(h - \langle h - 1 \rangle_W))/(lw) \rfloor \leqslant n_3$$

then no improved packing can be found for that $b$ and $c' \geqslant c$. Continue with the next $b \in S_b(W')$ if one exists. If

$$n(a, b) + n(g, h) + \lfloor (\Delta P - (c - e)(h - \langle h - 1 \rangle_W) - (L' - c))/(lw) \rfloor \leqslant n_3$$

then no improved packing can be found for that $c$. Continue with the next $c \in S_c(L', a)$ if one exists.

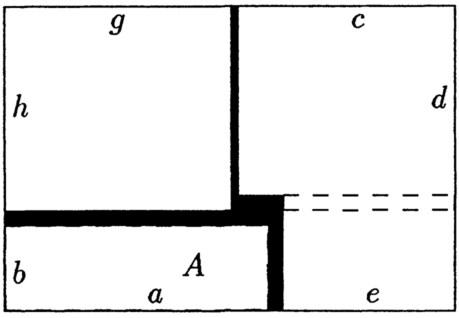


FIG. 5. *Termination criterion in the c-loop.*

Consider the situation in Figure 6 where additionally $d$ is fixed and all $d' \in S_d(W', b)$ with $d' \leqslant d$ have still to be considered. Let $\Delta P := L'W' - (L' - e)(W' - h) - h(L' - c)$. If

$$n(a, b) + n(g, h) + \lfloor (\Delta P - (c - e)(h - d))/(lw) \rfloor \leqslant n_3$$

then no improved packing can be found using $d' \leqslant d$. Continue with the next $c \in S_c(L', a)$ if one exists.

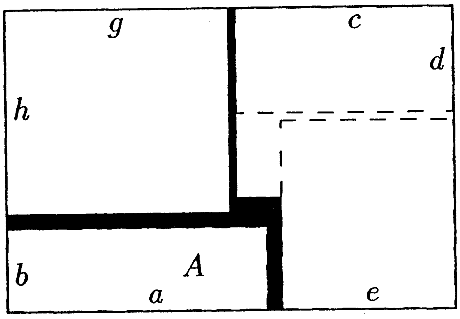


FIG. 6. *Termination criterion in the d-loop.*

## COMPUTATIONAL EXPERIENCE

In this section we report the behaviour of the new heuristic with respect to efficiency and computational effort. The G4-heuristic was implemented in Turbo-Pascal, Version 6.0, and the tests were done on a Pentium 90 PC.

First, let us consider instances of type I defined in a previous section. As reported[3] for any instance (except for four, cf. p. 163 in Ref. 3) of a set of representatives of the PLP of type I (called COVER I) at least one of the heuristics used yields an optimal solution for the representatives of COVER I. Since the G4-heuristic leads to patterns at least as good in comparison to that obtainable with the other heuristics, it remains to investigate the four unsolved instances of COVER I. Figures 7–9 show the corresponding G4-patterns which are also optimal. Hence, using the G4-heuristic an optimal pattern can be obtained for any instance of type I.

Nelissen[3] also considered PLP of type II which differ from those of type I by $51 \leqslant LW/(lw) < 101$. Applying the heuristics considered in Ref. 3, only 206 (hard) instances of COVER II

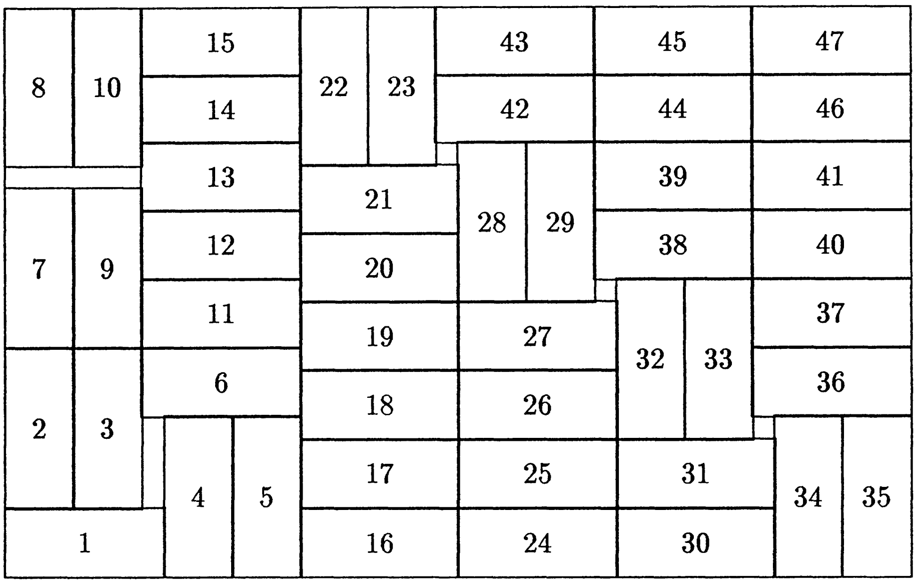


FIG. 7. *Optimal patterns for the instances (40, 25, 7, 3) and (52, 33, 9, 4).*

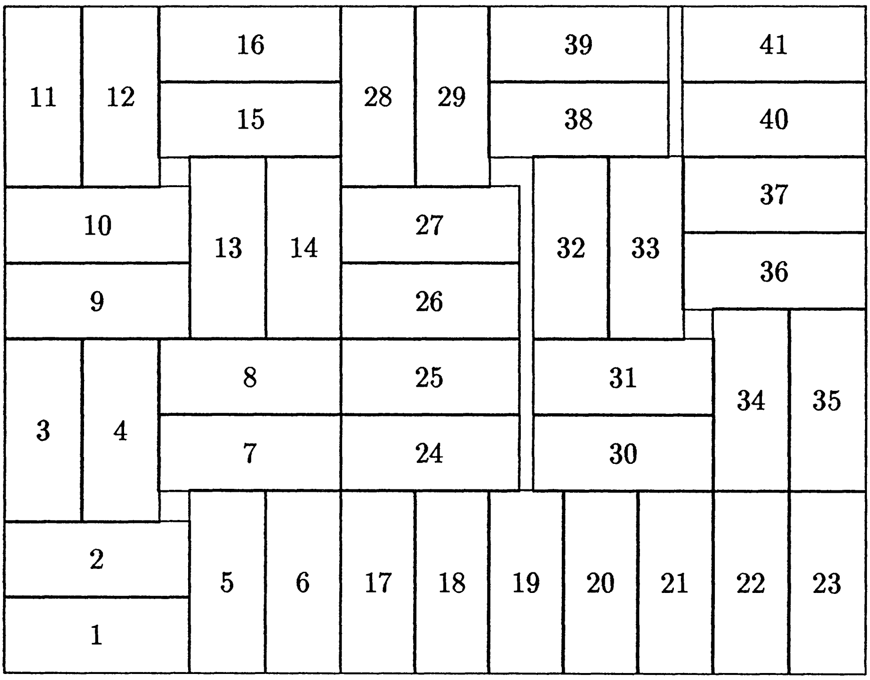


FIG. 8. *Optimal pattern for instance (57, 44, 12, 5).*

(which contains representatives of the PLP of type II) cannot be solved to optimality. But the G4-heuristic solves 167 of the 206 instances optimally. Moreover the gap between optimal value and value of the G4-solution was 1 for the remaining 39 instances. (J. Nelissen kindly placed the data of the 206 instances at our disposal.)

In order to give an impression on the amount of computer time needed for recurrence (3) when applied to the 206 hard instances of COVER II we give the average and maximal time (in seconds) in Table 1 (row 'G4-heuristic'). Moreover we consider two further variants of the G4-heuristic: the

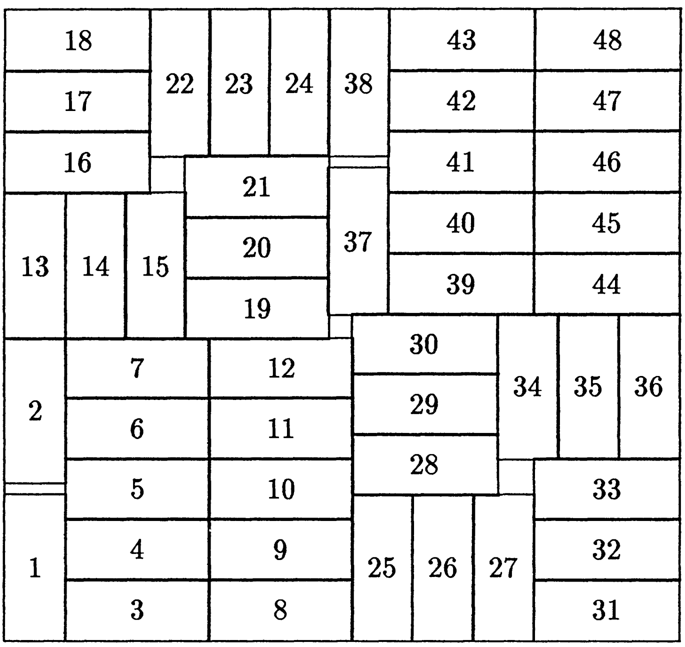


FIG. 9. *Optimal pattern for instance (56, 52, 12, 5).*

TABLE 1. *Computer time for the 206 hard instances of COVER II*

| | Average | Maximal |
|---|---|---|
| G4-heuristic | 0.22 | 0.55 |
| Without LP-bound | 0.12 | 0.33 |
| Without LP- and equivalence bound | 0.25 | 1.43 |

G4-heuristic without applying the LP-bound (row Without LP-bound) and the G4-heuristic without the LP bound and the 'equivalence' bound (row Without LP- and equivalence bound).

The usefulness of the LP-bound and the equivalence bound and the dependence of computer time on the data size is illustrated by means of six instances with $W = 200$, $l = 21$, $w = 19$ and different $L$-values, $L = 200, 250, \dots, 450$. Table 2 gives the value of the G4-patterns and the computer time for the three variants already considered in Table 1.

TABLE 2. *Comparison of variants in dependence of the L-size*

| | | | | | | |
|---|---|---|---|---|---|---|
| $L$: | 200 | 250 | 300 | 350 | 400 | 450 |
| G4-value: | 100 | 125 | 149 | 175 | 200 | 225 |
| G4-heurisic | 0.22 | 0.28 | 1.15 | 2.20 | 3.40 | 20.54 |
| Without LP-bound | 0.11 | 0.22 | 0.88 | 1.97 | 3.24 | 21.42 |
| Without LP- and equivalence bound | 0.49 | 2.52 | 6.65 | 9.17 | 33.84 | 47.13 |

In order to illustrate the computational behaviour of the G4-heuristic we give some details for the representative instance (300, 200, 21, 19). Table 3 contains the following information for the G4-heuristic using recurrence (2) (columns $R2$) and for using recurrence (3) (columns $R3$): Row $L$-raster gives the cardinalities of $S(L)$ and $\tilde{S}(L)$ minus 1, and row $W$-raster gives the cardinalities of $S(W)$ and $\tilde{S}(W)$ minus 1. The columns denoted with 'Ter' contain absolute quantities whereas the columns 'Rest' give the remaining number of calculations of $n(L', W')$-values. This number (row $n$-values) equals ($L$-raster $-$ 1)($W$-raster $-$ 1). The entries in row 'Symmetry' and columns 'Ter' give the number of times the $n$-calculation is not necessary because of symmetry. The entries in row '1-block pattern' and columns 'Ter' give the number of times a 1-block pattern (homogeneous pattern) is proved to be optimal using only the area bound. The number of times an optimal solution of a smaller pallet can be proven to be optimal also for the current pallet (using area bound) are shown in row '"Smaller" pallets'. The number of times the current solution is shown to be optimal because of Barnes's bound are given in row 'Barnes's bound'. For a large part of the remaining ($L'$, $W'$)-combinations a solution with G-structure is found (row 'G-structure'). Applying the equivalence bound and the LP bound leads to further terminations (rows 'Equivalence bound' and 'LP-bound'). The row '4-block structure' gives the number of times an improved solution is found with G4-structure and which is optimal. For the remaining ($L'$, $W'$)-combinations it is not proven that the obtained G4-solution is optimal. An optimal G4-pattern of instance (300, 200, 21, 19) contains 149 boxes and is shown in Figure 10.

TABLE 3. *Detailed analysis for instance (300, 200, 21, 19)*

| | R2 | | R3 | |
|---|---|---|---|---|
| $L$-raster: | 127 | | 71 | |
| $W$-raster: | 60 | | 35 | |
| | Ter | Rest | Ter | Rest |
| $n$-values | 7434 | 7434 | 2380 | 2380 |
| Symmetry | 1711 | 5723 | 561 | 1819 |
| 1-block pattern | 677 | 5046 | 240 | 1579 |
| 'Smaller' pallets | 924 | 4122 | 158 | 1421 |
| Barnes's bound | 210 | 3912 | 77 | 1344 |
| G-structure | 1435 | 2477 | 472 | 872 |
| Equivalence bound | 1971 | 506 | 727 | 145 |
| LP-bound | 195 | 311 | 40 | 105 |
| 4-block structure | 63 | 248 | 43 | 62 |

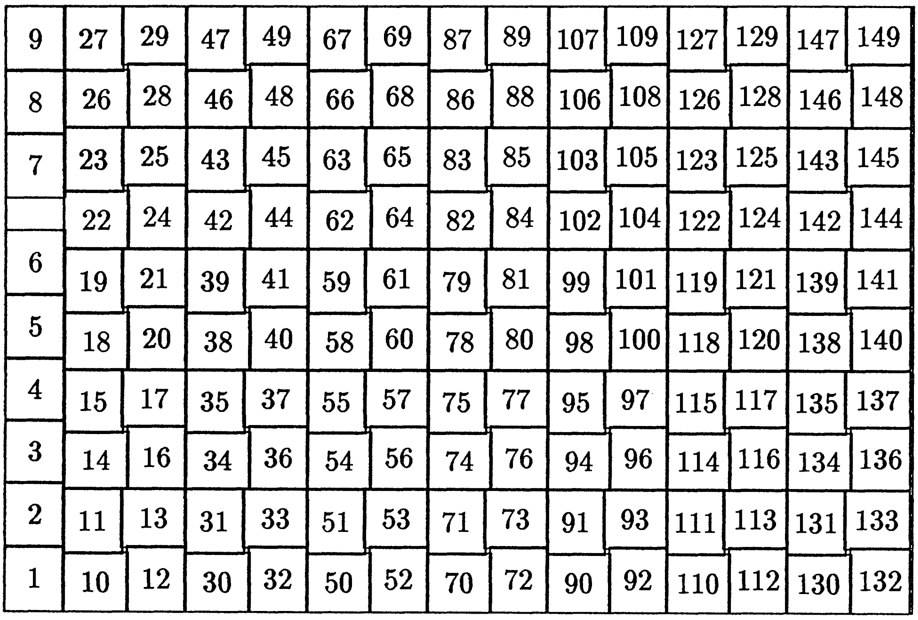


FIG. 10. *Optimal pattern for instance (300, 200, 21, 19).*

Clearly, the usage of recurrence (3) is of considerable advantage with respect to computer time needed in comparison to recurrence (2) and therefore this recurrence should be used in general to compute packing patterns. On the other hand, recurrence (2) is useful when packing patterns are needed for several $L$- and/or $W$-values.

## CONCLUDING REMARKS

In this paper we propose a new heuristic for the PLP which produces solutions at least as good as any heuristic known so far. In particular, when using this heuristic any instances of problems of type I (which are of the largest practical interest) are solved optimally. Moreover, the computational experiments reported show that the G4-heuristic can also be applied successfully to 'larger' instances (instances with up to 200 boxes per pattern).

## REFERENCES

1. K. A. DOWSLAND and W. B. DOWSLAND (1992) Packing problems. *Eur. J. Opl Res.* **56**, 2–14.
2. H. DYCKHOFF and U. FINKE (1992) *Cutting and Packing in Production and Distribution.* Physica Verlag, Heidelberg.
3. J. NELISSEN (1994) *Neue Ansätze zur Lösung des Palettenbeladungsproblems*, Dissertation RWTH Aachen.
4. H. DYCKHOFF (1990) A typology of cutting and packing problems. *Eur. J. Opl Res.* **44**, 145–160.
5. F. W. BARNES (1979) Packing the maximum number of $m \times n$ tiles in a $p \times q$ rectangle. *Disc. Math.* **26**, 93–100.
6. K. A. DOWSLAND (1984) The three-dimensional pallet chart; an analysis of the factors affecting the set of feasible layouts for a class of two-dimensional packing problems. *J. Opl Res. Soc.* **35**, 895–905.
7. H. ISERMANN (1987) Ein Planungssystem zur Optimierung der Palettenbeladung mit kongruenten rechteckigen Versandgebinden. *OR Spektrum* **9**, 235–249.
8. J. NELISSEN (1993) New approaches to the pallet loading problem, Working Paper, RWTH Aachen (available via anonymous ftp from ftp.informatik.rwth-aachen.de (137.226.225.3) as /pub/reports/pallet.ps.Z).
9. G. SCHEITHAUER (1995) Equivalence and dominance for problems of optimal packing of rectangles, Preprint MATH-NM-04-1995, TU Dresden.
10. J. HERZ (1972) Recursive computational procedure for two-dimensional cutting. *IBM J. Res. Develop.* **16**, 462–469.
11. K. A. DOWSLAND and W. B. DOWSLAND (1993) A comparative analysis of heuristics for the two-dimensional packing problem. Paper presented at EURO VI, Wien, 1993.
12. H. EXELER (1988) *Das homogene Packproblem in der betriebswirtschaftlichen Logistik.* Physica Verlag, Heidelberg.
13. G. NAUJOKS (1990) Neue Heuristiken und Strukturanalysen zum zweidimensionalen homogenen Packproblem. In: *Operations Research Proceedings 1989.* pp 257–263 Springer, Berlin/Heidelberg.
14. J. TERNO, R. LINDEMANN and G. SCHEITHAUER (1987) *Zuschnittprobleme und ihre praktische Lösung.* Verlag Harry Deutsch, Thun und Frankfurt/Main.
15. H. EXELER (1991) Upper bounds for the homogeneous case of a two-dimensional packing problem. *ZOR* **35**, 45–58.

*Received March 1995; accepted September 1995 after one revision*