Regular Paper

Anti-Slide

Kazuyuki Amano[1,a)] Shin-ichi Nakano[1,b)] Koichi Yamazaki[1,c)]

Received: July 31, 2014, Accepted: January 7, 2015

Abstract: The *anti-slide packing* is a packing of a number of three dimensional pieces of same size into a larger box such that none of them can slide in any direction. In this paper, we consider the problem of how to find a sparsest anti-slide packing. We give an IP formulation of this problem, and obtain the solutions for some small cases by using an IP solver. In addition, we give the upper and lower bounds on the ratio of the volume occupied by the pieces when the size of a box approaches infinity. For the case of piece size $2 \times 2 \times 1$, we show that a sparsest anti-slide packing occupies at least 28.8% and at most 66.7% of the total volume.

Keywords: anti-slide packing, integer programming, IP solver, puzzle

1. Introduction

Given a $4 \times 4 \times 4$ box and a number of $2 \times 2 \times 1$ pieces, consider a problem of how to pack the pieces into the box in such a way that none of them can slide in any direction. We assume that there is no friction between the pieces and, of course, a piece can never go out of the box. The object is to use the minimum number of pieces starting from 16 pieces. This puzzle, named *anti-slide*, was created by William Strijbos and won second place at the 1994 Hikimi Puzzle Competition in Japan [1].

An optimal solution uses 12 pieces. There are three such solutions [1]. One of them is shown in **Figs. 1** and **2**. The ratio of the volume occupied by the pieces is $48/64 = 75\%$.

It is quite natural to ask whether this ratio, 75%, is the best possible if we consider such an anti-slide packing in a larger box. This problem, to find a sparsest anti-slide packing, is a main focus of this paper.

One of the applications for a practical purpose is the following: Imagine that you are a manufacturer of caramel. The size of a caramel is 2 cm $\times$ 2 cm $\times$ 1 cm and you would like to sell it packed in a box of a certain size, e.g., a box with each side of length 4 cm. One cannot see the inside of the box. The problem is how to pack the caramels into the box. Assume that as the manufacturer, you want to pack them as sparse as possible so the contents do not rattle when the package is shaken!

Throughout the paper, we consider the *integral* and *orthgonal* version of the problem. The integral means that we assume all the coordinates of corner points of pieces are integers. The orthgonal means that we assume that each piece is axis-aligned and that we avoid sliding to any orthogonal direction, i.e., parallel to x, y or z-axes. In what follows, we call such an anti-slide packing *stable*. The *density* of a packing is defined to be the ratio of the volume

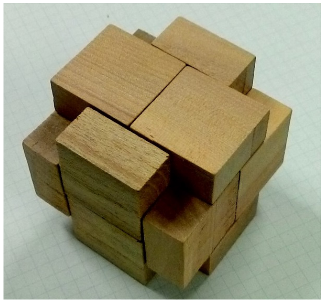


Fig. 1 A sparsest anti-slide packing of $2 \times 2 \times 1$ pieces for $4 \times 4 \times 4$ box.

	2	2	
3	2	2	4
3	1	1	4
	1	1	

5	5	6	6
3			4
3			4
7	7	8	8

5	5	6	6
9			10
9			10
7	7	8	8

	11	11	
9	11	11	10
9	12	12	10
	12	12	

Fig. 2 A packing of pieces in Fig. 1. Each number represents a piece of size $2 \times 2 \times 1$.

occupied by the pieces in the packing. Under these restrictions, the problem is to find a stable packing of pieces consisting of $2 \times 2 \times 1$ unit cells into a large box with the lowest density. Honestly speaking, at this moment, we do not know whether these two restrictions affect the solution of the original problem.

In this paper, we first give an integer linear programming (IP, in short) formulation of this problem for the case that the size of a box is finite. Then by using an IP solver we obtain an optimal solution for small boxes. Representing various puzzles by an IP model is nowadays considered as one of the standard techniques for analyzing and solving puzzles (see e.g., Refs. [3], [4], [5]). These will be described in Section 2.

Then, in Section 3, we give the upper and lower bounds on the density of a sparsest stable packing when a box is sufficiently large. The upper bound is obtained by observing that a packing for a small box can be extended by repeating its pattern in a certain way. It is notable that we obtain the lower bound also by solving an IP problem. We construct a non-trivial IP problem whose objective value gives a lower bound on the density of a

1 Department of Computer Science, Gunma University, Kiryu, Gunma 376–8515, Japan
a) amano@gunma-u.ac.jp
b) nakano@cs.gunma-u.ac.jp
c) koichi@cs.gunma-u.ac.jp

sparsest stable packing. It is shown that the density of a sparsest stable packing of $2 \times 2 \times 1$ pieces for a sufficiently large box is between 28.8% and 66.7%. The results of some experiments for another size of pieces are also presented.

The conclusion to the paper is given in Section 4.

All experiments in this paper are done on a standard PC (Intel Xeon E3-1240v3@3.40 GHz with 16 GB RAM on Windows 7) using the IP solver Gurobi 5.6.2 [2].

2. IP Formulation

For positive integers p and q with $p < q$, $[p]$ denotes $\{1, \ldots, p\}$ and $[p, q]$ denotes $\{p, p+1, \ldots, q\}$.

In this section, we give an IP model to obtain a sparsest stable packing in a box of size $\ell \times m \times n$. By the integral restriction, we view the box as the three dimensional array of cells of unit size. Each cell in the box is identified by $(i, j, k) \in [\ell] \times [m] \times [n]$ in the natural way. Throughout the paper, we assume without loss of generality that $l \geq m \geq n$.

We say a piece is placed at (i, j, k) if the closest cell of the piece to the origin of the box is (i, j, k). Also we say a piece is placed with direction $d \in \{0, 1, 2\}$ each of which is depicted in **Fig. 3**. For example, if a piece of size $2 \times 2 \times 1$ is placed at $(1, 3, 2)$ with direction $d = 1$, then the cells $(1, 3, 2), (1, 3, 3), (2, 3, 2)$ and $(2, 3, 3)$ are occupied by this piece.

We introduce two types of 0-1 variables

$$\{p[i, j, k, d] \mid (i, j, k) \in [\ell] \times [m] \times [n], d \in \{0, 1, 2\}\}$$

and

$$\{o[i, j, k] \mid (i, j, k) \in [\ell] \times [m] \times [n]\}.$$

The variable $p[i, j, k, d]$ takes value 1 when there is a piece placed at (i, j, k) with direction d. The variable $o[i, j, k]$ takes value 1 when the cell (i, j, k) is occupied. If we cannot place a piece at (i, j, k) with direction d inside the box, then the variable $p[i, j, k, d]$ is fixed to 0.

We below give a set of linear constraints such that the solution of the model gives a stable packing.

We should place at least one piece in the box. This is expressed as

$$\sum_{(i,j,k),d} p[i, j, k, d] \geq 1, \tag{1}$$

where the summation ranges over all $(i, j, k) \in [\ell] \times [m] \times [n]$ and all directions $d \in \{0, 1, 2\}$. We refer to this as the *non-emptiness condition*.

The variable $o[i, j, k]$ is expressed as

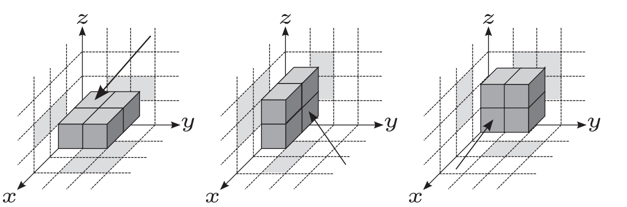


Fig. 3 The placement of a piece with direction $d = 0$ (left), $d = 1$ (middle) and $d = 2$ (right). The arrow indicates the cell closest to the origin of the box.

$$o[i, j, k] = \sum_{(i',j',k'),d} p[i', j', k', d] \tag{2}$$

where the summation ranges over all (i', j', k') and d such that the cell (i, j, k) is occupied when a piece is placed at (i', j', k') with direction d. We refer to Eq. (2) as the *non-intersecting condition*.

If we place a piece at (i, j, k) with direction d, then this piece occupies a number of cells. This can be written by a set of constraints

$$o[i', j', k'] \geq p[i, j, k, d] \tag{3}$$

for each cell (i', j', k') occupied by this piece. We refer to this as the *occupancy condition*. Note that the occupancy condition is actually implied by the non-intersecting condition and hence we can omit this from our IP model without changing the solution. We leave this here since we observed from experiments that adding this to the model helps to reduce the computation time in some (but less than a half) cases.

Finally, we describe the condition that every piece does not slide. Consider a piece P placed at (i, j, k) with direction d. In order to avoid sliding P, for each face F of P, at least one cell touching F should be occupied by another piece unless F meets the boundary of the box. See **Fig. 4**.

This can be described as follows: For each face F of a piece placed at (i, j, k) with direction d, except for F meeting the boundary of the box, we add the constraint

$$\sum_{(i',j',k') \in F'} o[i', j', k'] \geq p[i, j, k, d], \tag{4}$$

where F' is the set of cells touching F. We refer to this as the *anti-slide condition*.

It is easy to see that a feasible solution satisfying all the conditions described in Eqs. (1) to (4) is corresponding to a stable packing. The objective function is to minimize the number of pieces in the box. This is expressed by

$$\sum_{(i,j,k),d} p[i, j, k, d], \tag{5}$$

where the summation ranges over all $(i, j, k) \in [\ell] \times [m] \times [n]$ and all directions $d \in \{0, 1, 2\}$.

The smallest number of pieces used in a stable packing for small boxes obtained by an IP solver is shown in **Table 1**. The density of packings varies from 2/3 to 1. The minimum is attained when $(\ell, m, n) = (6, 6, 2)$ and $(6, 6, 4)$. A sparsest packing for the $6\times6\times2$ box is shown in **Fig. 5**, and the one for the $6\times6\times4$ box is consisting of two layers of this packing.

The computation time is dramatically increased as the volume of a box is increased. See **Fig. 6**. It seems that the logarithm of

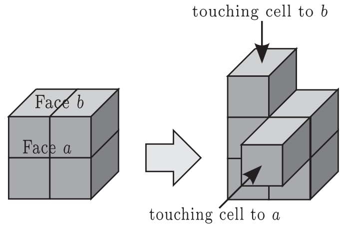


Fig. 4 A cell touching a face of the piece.

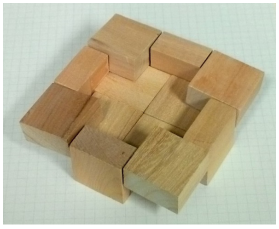
Table 1 Optimal solutions for small boxes. The symbol "∗" indicates the computation exceeds the time limit, which is set to 6 hours (= 21,600 seconds).

$n = 2$								
$m\backslash\ell$	2	3	4	5	6	7	8	9
2	2	3	4	5	6	7	8	9
3		4	6	6	7	8	9	10
4			8	8	9	11	13	14
5				10	11	14	15	17
6					12	15	17	19
7						18	21	22
8							24	∗
9								∗

$n = 3$							
$m\backslash\ell$	3	4	5	6	7	8	9
3	6	8	10	12	14	15	17
4		10	12	14	16	18	20
5			15	18	21	24	27
6				21	24	27	∗
7					28	∗	∗
8						∗	∗
9							∗

$n = 4$						
$m\backslash\ell$	4	5	6	7	8	9
4	12	15	18	20	24	∗
5		19	22	26	∗	∗
6			24	∗	∗	∗
7				∗	∗	∗
8					∗	∗
9						∗

$n = 5$		
$m\backslash\ell$	5	6
5	24	∗
6		∗



		5	5		
	3	3	7	7	
1	3	3	7	7	8
1	2	2	6	6	8
	2	2	6	6	
		4	4		

10	10	5	5	12	12
10	10			12	12
1					8
1					8
9	9			11	11
9	9	4	4	11	11

Fig. 5 A sparsest packing of $2 \times 2 \times 1$ pieces for $6 \times 6 \times 2$ box with density $2/3$.

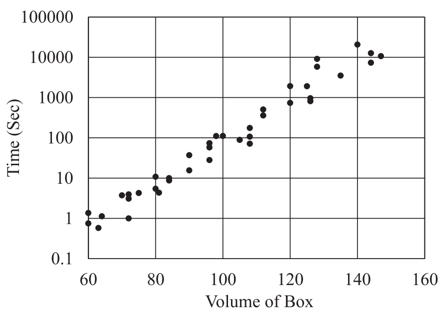


Fig. 6 The computation time. The data for boxes of volume < 60 is omitted.

the time tends to be proportional to the volume of a box. Note that we did not use the occupancy condition (i.e., Eq. (3)) to obtain the data for Fig. 6.

For example, the computation for $4 \times 4 \times 4$ box (which gives a solution in Fig. 1) finishes in less than two seconds, whereas it takes around one hour to finish the computation for $5 \times 5 \times 5$ box; which gives a packing with 24 pieces shown in **Fig. 7**. The density of this packing is 76.8%, which is denser than the one for

Fig. 7 A sparsest stable packing of $2 \times 2 \times 1$ pieces for $5 \times 5 \times 5$ box, which uses 24 pieces. Getting a development plan of this packing is left to the readers.

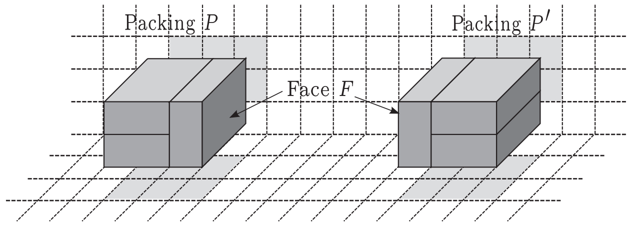


Fig. 8 Constructing a stable packing by joining two reflected copies of packing for a smaller box.

$4 \times 4 \times 4$ box, and in addition this is harder to pack.

3. Limit of Volume

It is interesting to see the density of a sparsest stable packing when the size of a box approaches infinity. Consider a packing using $a \times b \times c$ pieces with $a \geq b \geq c$. Let $C(a, b, c; n)$ denote the number of occupied cells of a sparsest stable packing for $n \times n \times n$ box. We denote by $\overline{V}(a, b, c)$ and $\underline{V}(a, b, c)$ the upper and lower limits of the density of a sparsest stable packing where n approaches infinity, i.e.,

$$\overline{V}(a, b, c) := \limsup_{n\to\infty} \frac{C(a, b, c; n)}{n^3},$$

$$\underline{V}(a, b, c) := \liminf_{n\to\infty} \frac{C(a, b, c; n)}{n^3}.$$

We believe that the limit $\lim_{n\to\infty} C(a, b, c; n)/n^3$ exists, although we could not give a proof of the convergence.

3.1 Upper Bounds

Given any stable packing P of a box B, we can construct a stable packing of a larger box L which is twice as large as B, as follows. Assume L consists of two boxes B_1 and B_2, and each of which is identical to B. First pack B_1 by P, then pack B_2 by P' so that P' is the mirror image of P with respect to the shared face of B_1 and B_2. See **Fig. 8**.

This procedure can be repeated arbitrary times in any direction. Hence we can get a stable packing of a large box whose density is same as the one for the "base" packing. This says that any packing obtained in the previous section gives an upper bound on $\overline{V}(2, 2, 1)$. The sparsest packing obtained is for $6 \times 6 \times 2$ box (see Fig. 5). It uses 12 pieces and so we have $\overline{V}(2, 2, 1) \leq (12 \cdot 4)/72 = 2/3 \sim 66.7\%$.

If one would like to seek more regular patterns, the following definition is useful.

Definition 1 A stable packing of pieces in a finite box is called *extendable*, if we repeat its placement any number of times in any direction then we can obtain a stable packing for a box of

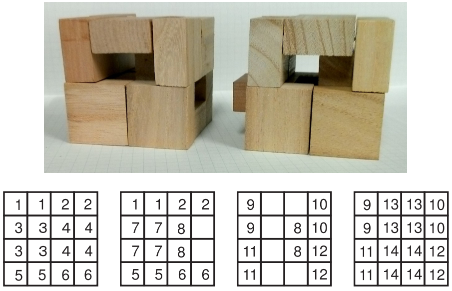


Fig. 9 A stable but not extendable packing of $2 \times 2 \times 1$ pieces. If we place two copies of this packing side by side, then the piece No. 7 in the right box will slide to the left.

Table 2 Optimal solutions for small toruses. The symbol "$*$" indicates the computation has timed out (= 6 hours). The values are identical to those in Table 1 except for $(\ell, m, n) = (7, 4, 4)$, which is underlined.

$n = 2$

$m \backslash \ell$	2	3	4	5	6	7	8	9
2	2	3	4	5	6	7	8	9
3		4	6	6	7	8	9	10
4			8	8	9	11	13	14
5				10	11	14	15	17
6					12	15	17	19
7						18	21	22
8							24	$*$
9								$*$

$n = 3$

$m \backslash \ell$	3	4	5	6	7	8	9
3	6	8	10	12	14	15	17
4		10	12	14	16	18	20
5			15	18	21	24	27
6				21	24	27	30
7					28	$*$	$*$
8						$*$	$*$
9							$*$

$n = 4$

$m \backslash \ell$	4	5	6	7	8	9
4	12	15	18	$\underline{22}$	24	$*$
5		19	22	26	$*$	$*$
6			24	$*$	$*$	$*$
7				$*$	$*$	$*$
8					$*$	$*$
9						$*$

$n = 5$

$m \backslash \ell$	5	6
5	24	$*$
6		$*$

a certain size.

It is an easy observation that an extendable stable packing can be obtained by a similar IP model as described in Section 2 in which we treat the box in a torus-like fashion. More precisely, when we build the constraints we consider that the top face of the box meets the bottom face of (an another copy of) the same packing. Similarly, the left and front faces are considered to meet the right and back faces, respectively. Note that there is a packing that is stable but not extendable. See **Fig. 9** for example.

The number of pieces used in a sparsest extendable packing for small boxes given by an IP solver is shown in **Table 2**. Interestingly, the values are identical to those without extendable condition (shown in Table 1) except for $(\ell, m, n) = (7, 4, 4)$. The density of packings varies from 2/3 to 1. As to the case without extendable condition, the minimum attains when $(\ell, m, n) = (6, 6, 2)$ and $(6, 6, 4)$. The packing obtained for $(\ell, m, n) = (6, 6, 2)$ is, again, the one shown in Fig. 5. Note that the packing for $4 \times 4 \times 4$ box shown in Fig. 1 is also extendable.

We also conduct some experiments for another size of pieces such as $4 \times 2 \times 1$. The sparsest stable packing of $4 \times 2 \times 1$ pieces obtained so far has the density of 50%. See **Fig. 10**.

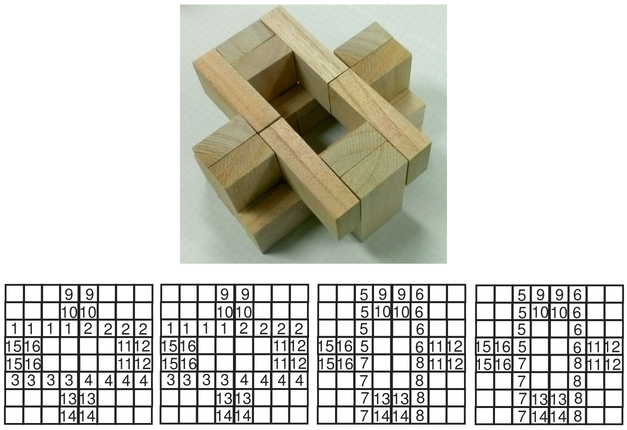


Fig. 10 An extendable packing of $4 \times 2 \times 1$ pieces for $8 \times 8 \times 4$ box with density 1/2.

In summary, we have

Theorem 2 $\overline{V}(2, 2, 1) \leq 2/3$ and $\overline{V}(4, 2, 1) \leq 1/2$.

3.2 Lower Bounds

In this subsection, we consider the lower bound on the density of a sparsest stable packing. For a while, we restrict our attention to the case that the size of a piece is $2 \times 2 \times 1$.

In order to obtain the lower bound, one possible approach is to consider the linear programming (LP) relaxation of the IP model shown in Section 2. We can obtain the LP relaxation of our IP model by removing the boolean constraints for the variables $p[\cdot, \cdot, \cdot, \cdot]$'s and $o[\cdot, \cdot, \cdot]$'s. Clearly, the optimum value of the objective function (i.e., Eq. (5)) of the LP relaxation problem gives a lower bound on the number of pieces needed to construct a stable packing. Hence the multiplication of 4 and this bound must be a lower bound on the number of occupied cells in any stable packing.

Unfortunately, this approach does not seem to work. In fact, we verify using Gurobi that the optimum value of an LP relaxation problem is 1 for every reasonable size of box (e.g., $(\ell, m, n) = (20, 20, 20)$). It seems that the non-emptiness condition in Eq. (1) can be satisfied with equality. This means that we should take an another approach.

A *hole* is a sub-box H consisting only of empty cells such that each of the six faces of H does not meet the boundary of the box.

We first show that a large hole cannot exist.

Lemma 3 For every stable packing of $2 \times 2 \times 1$ pieces for a sufficiently large box, a hole of size $5 \times 5 \times 2$ or larger cannot exist.

Proof. We verify the lemma by giving an IP model such that its feasibility corresponds to the existence of such a hole and then verifying the infeasibility of the obtained model. We give no objective function in the IP model, because we are interested in only the feasibility for the proof.

Given a stable packing for a sufficiently large box, suppose that there is a hole of size $5 \times 5 \times 2$ or larger. We can assume without loss of generality that there is a hole H of size $5 \times 5 \times 2$ such that the upper 5×5 face of H has a touching cell occupied by a piece. We denote the set of all touching cells to the upper face of H, which contains this occupied cell, by S. Note that $H \cup S$ forms a sub-box of size $5 \times 5 \times 3$ such that H is empty and S is *not* empty (*).

We show below that in any packing with the hole H some piece near $H \cup S$ cannot avoid sliding. To this end, we consider a sub-box Z of size $9 \times 9 \times 7$ consisting of $H \cup S$ together with all surrounded cells of distance at most two to $H \cup S$. We seek a *pseudo-stable* packing in Z, which is intuitively a packing with the assumption that every piece containing a cell outside of Z is not sliding. More precisely, in a pseudo-stable packing, a piece P will not slide to a certain direction even if there is no occupied cells touching the face of P when one can place another piece Q touching P so that Q contains a cell outside of Z. See **Fig. 11**.

It is easy to observe that, for any stable packing that contains a sub-box $H \cup S$ satisfying (*), there is a pseudo-stable packing for Z. Hence, in order to show the lemma, it is sufficient to verify that there is no pseudo-stable packing for Z.

It is not hard to construct an IP model for checking the existence of such a packing by slightly modifying the model given in the previous section. Let $H = [3, 7] \times [3, 7] \times [3, 4]$, $S = [3, 7] \times [3, 7] \times \{5\}$ and $Z = [9] \times [9] \times [7]$ as shown in **Fig. 12**.

We can use constraints describing the non-intersecting and the occupancy conditions (in Eqs. (2) and (3)) without modification. Then add the constraints

$$\sum_{(i,j,k) \in H} o[i, j, k] = 0,$$

which says H is empty, and

$$\sum_{(i,j,k) \in S} o[i, j, k] \geq 1,$$

which says S is not empty. Note that without the constraint on S a packing with no pieces will be a feasible solution to the model. This is the reason why we introduce S.

We also introduce a new set of 0-1 variables $r[i, j, k]$ for $(i, j, k) \in Z$. The variable $r[i, j, k]$ takes value 1 iff the cell (i, j, k) is occupied by a piece, i.e., $o[i, j, k] = 1$, or there would be occupied by a piece cut by the boundary of Z. Actually, in the case of piece size $2 \times 2 \times 1$, we add the following constraints: If at least

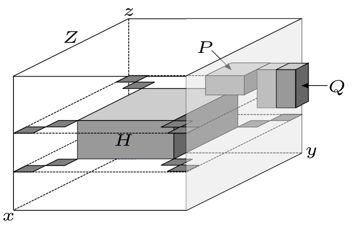


Fig. 11 A pseudo-stable packing. The piece Q touching the right face of the piece P exceeds the boundary of Z. In such a case, we assume that P does not slide to right even when Q is removed.

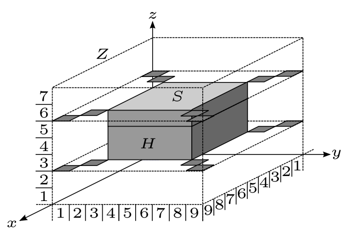


Fig. 12 The sub-boxes H, S and Z.

one of i, j and k is 1, or $i = 9$, $j = 9$ or $k = 7$, then

$$r[i, j, k] = 1,$$

and if otherwise

$$r[i, j, k] = o[i, j, k].$$

Finally, we place a similar constraint to the anti-slide condition described in Eq. (4), in which $o[i, j, k]$ is replaced by $r[i, j, k]$.

By using an IP solver, it is easy to verify that the above model is infeasible, which implies the conclusion of the lemma. □

Remark that the size $5 \times 5 \times 2$ in Lemma 3 is optimal in a sense that we cannot refute the existence of a smaller hole by an argument used in the proof of Lemma 3. This does not mean that a smaller hole such as $5 \times 4 \times 2$ *can* exist in a stable packing. In fact, we do not know the size of a largest hole of a stable packing. It is known that we can create a $4 \times 2 \times 2$ hole by stacking two copies of placements in Fig. 5 with upside-down which yields a stable packing for $6 \times 6 \times 4$ box.

We now turn to the lower bound on $\underline{V}(2, 2, 1)$. Lemma 3 immediately implies $\underline{V}(2, 2, 1) \geq 1/(5 \times 5 \times 2) = 0.02$ since every $5 \times 5 \times 2$ sub-box contains at least one occupied cell.

As expected, we can obtain a better lower bound by considering a larger sparse but non-hole sub-box.

We below give an IP model L such that the optimal value of the objective function, denoted by $\mathrm{Obj}(L)$, gives a lower bound on the number of occupied cells in any $\ell \times m \times n$ sub-box of a stable packing. Once this is shown, the lower bound

$$\underline{V}(2, 2, 1) \geq \frac{\mathrm{Obj}(L)}{\ell m n}$$

is obvious.

The model L is almost same to that in the proof of Lemma 3.

Let H be a sub-box of size $\ell \times m \times n$ not smaller than $5 \times 5 \times 2$. We put $H = [3, \ell + 2] \times [3, m + 2] \times [3, n + 2]$ and $Z = [\ell + 4] \times [m + 4] \times [n + 4]$, and seek a pseudo-stable packing for Z.

As before, we use three types of variables : (i) $p[i, j, k, d]$ which indicates if a piece is placed at (i, j, k) with direction d, (ii) $o[i, j, k]$ which indicates if the cell (i, j, k) is occupied, and (iii) $r[i, j, k]$ which indicates if the cell (i, j, k) is occupied or it can be occupied by a piece cut by the boundary of Z.

The objective function is to minimize

$$\sum_{(i,j,k) \in H} o[i, j, k],$$

which is the number of occupied cells in H. Note that in the objective function, the variables $o[\cdot, \cdot, \cdot]$ are considered rather than the variables $p[\cdot, \cdot, \cdot, \cdot]$ (cf. IP formulation given in Section 2).

By Lemma 3, we can assume that H is not empty, i.e.,

$$\sum_{(i,j,k) \in H} o[i, j, k] \geq 1.$$

We use constraints describing the non-intersecting and the occupancy conditions (i.e., Eqs. (2) and (3)) without modification. We put

$$o[i, j, k] = \sum_{(i',j',k'),d} p[i', j', k', d] \qquad (\forall (i, j, k) \in Z),$$

where the summation ranges over all (i', j', k') and d such that the cell (i, j, k) is occupied when a piece is placed at (i', j', k') with direction d. In addition, for every possible (i, j, k) and d, we add a set of constraints

$$o[i', j', k'] \geq p[i, j, k, d],$$

for each cell (i', j', k') occupied by the piece placed at (i, j, k) with direction d.

As to the proof of Lemma 3, we seek a pseudo-stable packing and introduce r-variables satisfying

$$r[i, j, k] = \begin{cases} o[i, j, k] & (i, j, k) \in [2, \ell+3] \times [2, m+3] \times [2, n+3], \\ 1 & \text{otherwise.} \end{cases}$$

The final constraints express the anti-slide condition, which is similar to Eq. (4). For each possible (i, j, k) and d, and for each face F of the piece placed at (i, j, k) with direction d, we add a set of constraints

$$\sum_{(i', j', k') \in F'} r[i', j', k'] \geq p[i, j, k, d],$$

where F' is the set of cells touching F. This finishes the description of L whose optimal value gives a minimum number of occupied cells in H by a pseudo-stable packing for Z.

It would be natural to expect that a larger lower bound can be obtained by considering a larger sub-box. In fact, the optimal values of the model L for $(\ell, m, n) = (5, 5, 2), (5, 5, 3)$ and $(5, 5, 4)$ are 1, 7 and 24, respectively. The lower bounds on $\underline{V}(2, 2, 1)$ obtained by them are $1/(5 \times 5 \times 2) = 0.02$, $7/(5 \times 5 \times 3) > 0.093$ and $24/(5 \times 5 \times 4) = 0.24$.

However, solving L is quite time consuming. So far, we could solve L only up to $(\ell, m, n) = (5, 5, 5)$. The optimal value for this case is 36, which implies the following theorem.

Theorem 4 For every stable packing of $2 \times 2 \times 1$ pieces for every sufficiently large box, every $5 \times 5 \times 5$ sub-box contains at least 36 occupied cells. Hence $\underline{V}(2, 2, 1) \geq 36/5^3 = 0.288$. □

There still is a large gap between the upper and lower bounds on the density of a sparsest packing that we have obtained.

4. Concluding Remarks

In this paper, we give an IP formulation of the problem to find an anti-slide (stable) packing and give the upper and lower bounds on the density of a sparsest stable packing.

There are many interesting problems to be pursued in future studies. Apparently, the most interested one is to determine the sparsest packing of various piece sizes. We conjecture that the current best solution (given by the repeated use of the pattern shown in Fig. 5) is in fact the sparsest packing of $2 \times 2 \times 1$ pieces. If this is true, how can it be proven? We list below some of other problems.

- Does the limit $\lim_{n \to \infty} C(a, b, c; n)/n^3$ exist?
- What is the size of a largest possible hole in a stable packing?
- Repeated use of a small pattern appears to always give the optimal packing. Can we show this formally?
- Is it true that the integral and orthogonal restrictions do not affect the density of the sparsest stable packing?

Acknowledgments The authors would like to thank anonymous referees for their helpful comments. This research was partially supported by KAKENHI Grant Number 24500006, 24500007, 24106006 and 26330004.

References

[1] Anti-Slide, available from ⟨http://www.johnrausch.com/Puzzleworld/puz/anti_slide.htm⟩ (retrieved July, 2014).

[2] Gurobi Optimization Inc. Gurobi Optimizer, available from ⟨http://www.gurobi.com⟩.

[3] Barlett, A., Chartier, T.P., Langville, A.N. and Rankin, T.D.: Integer Programming Model for the Sudoku Problem, *J. Online Mathematics and its Applications*, Vol.8, Article ID 1798 (2008).

[4] Demaine, E.D., Okamoto, Y., Uehara, R. and Uno, Y.: Computational Complexity and an Integer Programming Model of Shakashaka, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Science*, Vol.E97-A, pp.1213–1219 (2014).

[5] Meuffels, W.J.M. and Hertog, D. den: Puzzle–Solving the Battleship Puzzle as an Integer Programming Problem, *INFORMS Trans. Education*, Vol.10, No.3, pp.156–162 (2010).

Kazuyuki Amano received his M.E. and D.E. degrees in information sciences from Tohoku University in 1993 and 1996, respectively. Presently, he is a professor in Department of Computer Science, Gunma University, Kiryu, Japan. His research interests include the theory of computation and combinatorics. He enjoys writing computer programs to solve puzzles and problems of discrete mathematics.

Shin-ichi Nakano received his M.E. and D.E. degrees in engineering from Tohoku University in 1987 and 1992, respectively. Presently, he is a professor in Department of Computer Science, Gunma University, Kiryu, Japan. His research interests include graph algorithms and combinatorial algorithms.

Koichi Yamazaki received his M.E. and Ph.D. degrees in 1989 and 1995, respectively from Tokyo Denki University. He is a professor in Department of Electronics and Informatics, Gunma University. His research focuses on discrete mathematics (graph theory, graph/approximation algorithms, combinatorial optimization, etc.).