Kaboozle is NP-complete, even in a Strip

Tetsuo Asano[1], Erik D. Demaine[2], Martin L. Demaine[2], and Ryuhei Uehara[1]

[1] Japan Advanced Institute of Science and Technology (JAIST)
Ishikawa 923-1292, Japan
{t-asano,uehara}@jaist.ac.jp

[2] Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology (MIT), Cambridge, MA 02139, USA
{edemaine,mdemaine}@mit.edu

Abstract. Kaboozle is a puzzle consisting of several square cards, each annotated with colored paths and dots drawn on both sides and holes drilled. The goal is to join two colored dots with paths of the same color (and fill all holes) by stacking the cards suitably. The freedoms here are to reflect, rotate, and order the cards arbitrarily, so it is not surprising that the problem is NP-complete (as we show). More surprising is that any one of these freedoms—reflection, rotation, and order—is alone enough to make the puzzle NP-complete. Furthermore, we show NP-completeness of a particularly constrained form of Kaboozle related to 1D paper folding. Specifically, we suppose that the cards are glued together into a strip, where each glued edge has a specified folding direction (mountain or valley). This variation removes the ability to rotate and reflect cards, and restricts the order to be a valid folded state of a given 1D mountain-valley pattern.

Keywords: Kaboozle, Transposer, silhouette, puzzles, origami.

1 Introduction

Kaboozle: The Labyrinth Puzzle is a puzzle created and developed in 2007 by Albatross Games Ltd., London.[3] This "multi-layer labyrinth" consists of four square cards; see Fig. 1. (In fact, each card is octagonal, but the pattern on it is a square.) Each card has holes drilled in different locations, and various colored paths and dots drawn on both sides. The goal is to arrange the cards—by rotation, reflection, and stacking in an arbitrary order—to create a continuous monochromatic path between the corner dots of the same color that is visible on one side of the stack. The goal of this paper is to understand what makes this puzzle NP-complete, when generalized to n cards instead of four.

Kaboozle is an example of a broader class of puzzles in which patterned pieces with holes must be arranged to achieve some goal, such as monochromatic sides. For example, Albatross Games Ltd. places Kaboozle in a series of puzzles called *Transposers*,[4] which all have this style. See [4] for descriptions, and [10] for the relevant patent. Our NP-hardness proofs for Kaboozle immediately imply NP-completeness for this general family of puzzles, though there are likely other special cases of interest.

[3] http://www.transposer.co.uk/KABpage1.htm
[4] http://www.transposer.co.uk/

Fig. 1. The four Kaboozle cards and one of the ten solutions.

An earlier form of this type of puzzle is a *silhouette puzzle*, where pieces are regions with holes (no pattern beyond opaque/transparent) and the goal is to make a target shape. Perhaps the first silhouette puzzle, and certainly the best known, is the "Question du Lapin" or "Rabbit Silhouette Puzzle", first produced in Paris around 1900 [7, p. 35]. Fig. 2 shows the puzzle: given the five cards on the left, stack them with the right orientations to obtain one of two different rabbit silhouettes. The puzzle can be played online.[5]

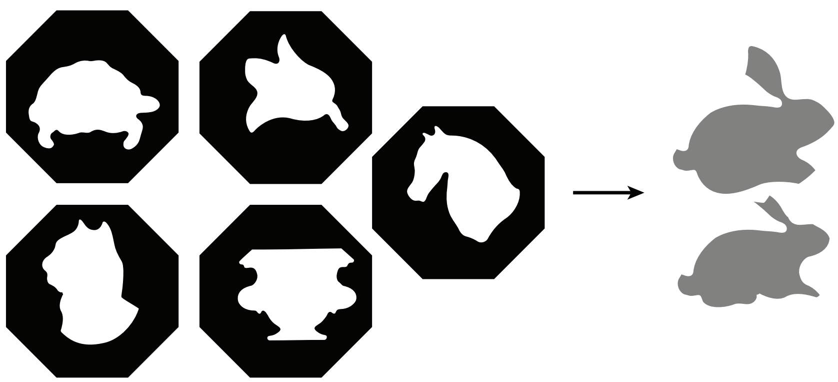

Fig. 2. The classic silhouette puzzle "Question du Lapin".

The freedoms in a silhouette puzzle are reflection and rotation of the cards; the card stacking order has no effect on the silhouette. (In fact, both rabbits can be obtained without reflecting the cards in Fig. 2, so that puzzle only needs rotation.) Are these freedoms enough for NP-completeness? We show that indeed silhouette puzzles are NP-complete, even allowing just rotation or just vertical reflection of the pieces. Furthermore, we show that Kaboozle is NP-complete under the same restriction of just rotation or just vertical reflection.

[5] http://www.puzzles.com/PuzzlePlayground/Silhouettes/Silhouettes.htm

But is reflection or rotation necessary for Kaboozle to be NP-complete? We show that Kaboozle is NP-complete even when the cards can only be stacked in a desired order, without rotation or reflection. We also show that Kaboozle is NP-complete when restricted to a restricted class of orderings that arise from paper folding, as described below.

Our folding variation of Kaboozle is inspired by a 1907 patent [5] commercialized as the (politically incorrect) "Pick the Pickaninnies" puzzle [8]. This puzzle consists of a single piece, shown on the left of Fig. 2, with holes, images (stars), and crease lines. The goal is to fold along the crease lines to make an array of stars, as shown on the right. This type of puzzle severely limits the valid stacking orders of the parts, while also effectively forbidding rotation and reflection of the parts.

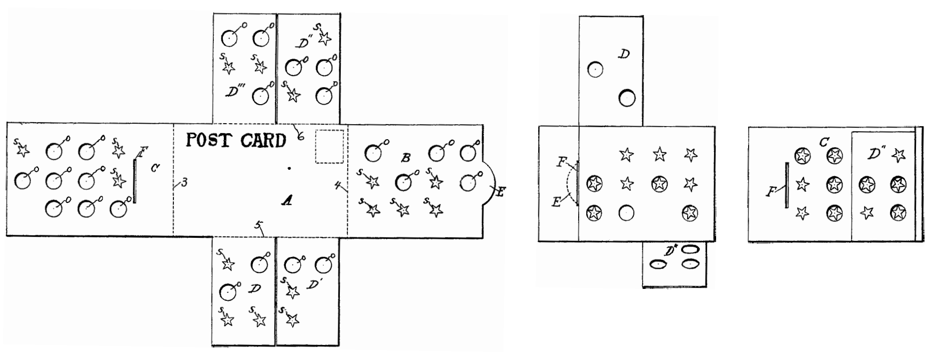

Fig. 3. Puzzle commercialized as "Pick the Pickaninnies". Figure from [5].

We consider a simple general puzzle along these lines, by restricting a generalized Kaboozle puzzle. Namely, we glue all the cards in the Kaboozle puzzle into a strip, and specify the folding direction (mountain or valley) on each glued edge (crease). Now the only freedom is folding the 1D strip of paper down to a unit size, respecting the folding directions. This freedom is a weak form of the ordering of the cards; rotation and reflection are effectively forbidden.

This idea also comes from problems in computational origami. In polynomial time, we can determine whether a mountain-valley pattern on a 1D strip of paper can be folded flat, when the distances between creases are not all the same [1]. A recent notion is *folding complexity*, the minimum number of simple folds required to construct a unit-spaced mountain-valley pattern (string) [2]. For example, n pleats alternating mountain and valley can be folded in a polylogarithmic number of simple folds and unfolds. On the contrary, the number of different ways to fold a uniform mountain-valley pattern of length n down to unit length is not well-investigated. The number of foldings of a paper strip of length n to unit length has been computed by enumeration, and it seems to be exponentially large; the curve fits to $\Theta(3.3^n)$ [6, A000136]. However, as far as the authors know, the details are not investigated, and it was not known whether this function is polynomial or exponential. Recently, the last author showed theoretical lower and upper bounds of this function: it is $\Omega(3.07^n)$ and $O(4^n)$ [9]. These results imply that a given random mountain-valley pattern of length n has $\Theta(1.65^n)$ foldings on average, which is bounded between $\Omega(1.53^n)$ and $O(2^n)$.

Intuitively, the folding version of the Kaboozle puzzle seems easy. Perhaps we could apply the standard dynamic programming technique from one side of the strip? But this intuition is not correct. Essentially, the problem requires folding a 1D strip of paper, but the strip has labels which place constraints on the folding. Despite the situation being quite restrictive, we prove the problem is still NP-complete.

Therefore we conclude that the generalized Kaboozle problem is NP-complete even if we allow only one of ordering, rotation, or reflection of the cards, and in the ordering case, even if the ordering comes from a 1D strip folding.

2 Preliminaries

We generalize the number of the Kaboozle cards to $n + 1$. Each *card* is square, with some fragments of a path drawn on both sides, and some holes drilled into it. We will use just one color of path we have to join. The (potential) endpoints of a path are distinguishable from the other fragments. To simplify, we assume that the cards are numbered $0, 1, 2, \ldots, n$.

A *strip* of the cards can be constructed as follows: for each $0 \le i \le n-1$, the right side of the card i is glued to the left side of the card $i + 1$, and that side is called the $(i + 1)$st *crease*. Each crease has a *label* "M" or "V" which means that the strip must be mountain folded or valley folded at the crease. (We define one side of the strip as the *top side*, and creases are mountain or valley folded with respect to this side.) We assume that the label of the first crease is "M" without loss of generality, or otherwise specified. For a strip of the cards, a *folded state* is a flat folding of unit length (where the unit is the width of a card) such that each crease is consistent with its label. (A folded state always exists for any string of labels [9].)

The main problem in this paper is the following:

Input: A strip of $n + 1$ Kaboozle cards, each with a label of length m.
Question: Determine whether the strip has a folded state that is consistent with the labels, and exactly one connected path is drawn on a surface of the folded state.

We begin with an observation for folding a unit pattern:

Observation 1 *A strip of $n + 1$ cards with n creases has a unique folded state if and only if the crease pattern is a pleat, i.e., "MVMV···MV" or "MVMV···MVM".*

Proof. Suppose that a mountain-valley pattern has a unique folded state. Without loss of generality, we assume that the first crease is a mountain. If the second crease is also a mountain, we have two folded states of the cards 1, 2, and 3: 2, 1, 3 and 2, 3, 1. Hence the second crease must be valley. We can repeat the argument for each crease, and obtain the pleat pattern. □

Using the pleats, we introduce a useful folding pattern for NP-completeness, namely, the *shuffle pattern* of length i: "$(\mathrm{MV})^{i-1}\mathrm{MM}(\mathrm{VM})^{i-1}$".[6] By Observation 1, the

[6] Here we use the standard notation x^k for string repetition. For example, "$(\mathrm{MV})^3\mathrm{MM}(\mathrm{VM})^3$"="MVMVMVMMVMVMVM".

left and right pleats are folded uniquely and independently. However, these pleats can be combined in any order to fold to unit length. Thus we have $\binom{2i}{i}$ distinct foldings of the shuffle pattern of length i. We note that the center card of the shuffle pattern of length i, the card $i + 1$ in our notation, always appears on one side of any folded state. We call this side the *top* of the shuffle pattern, and card $i+1$ the *top card* (although it may come to the "bottom" in a natural folding).

3 NP-completeness of generalized Kaboozle

It is easy to see that all the problems in this paper are in NP. Hence we concentrate on the proofs of NP-hardness. Our reduction is from the *1-in-3 3SAT problem*:

Input: A conjunctive normal form (CNF) Boolean formula $F(x_1, \dots, x_n) = c_1 \wedge c_2 \wedge \cdots \wedge c_m$, where each clause $c_i = \ell_1^i \vee \ell_2^i \vee \ell_3^i$ has three literals $\ell_j^i \in \{x_1, \dots, x_n, \overline{x_1}, \dots, \overline{x_n}\}$.
Question: Determine whether F has a truth assignment such that each clause contains exactly one true literal.

This problem is a well-known NP-complete variant of 3-satisfiability [3, LO4].

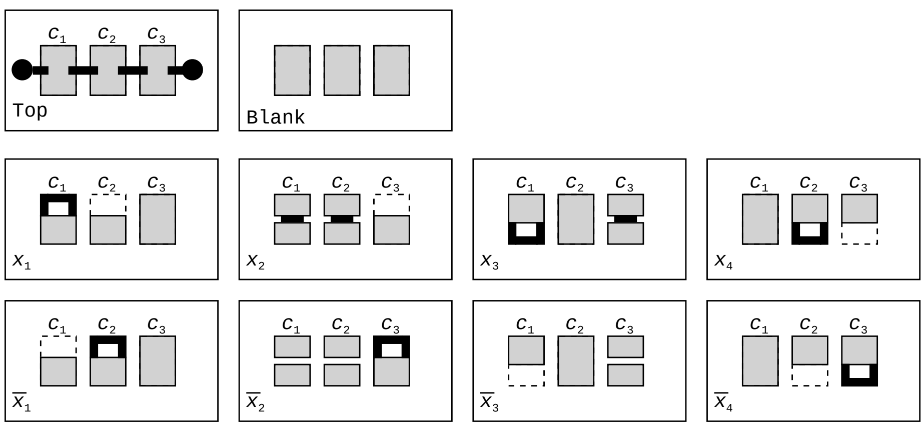

Fig. 4. Example of the reduction for $F(x_1, x_2, x_3, x_4) = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_4) \wedge (\overline{x_2} \vee x_3 \vee \overline{x_4})$.

For a given CNF formula $F(x_1, \dots, x_n)$ with n variable and m clauses, we use $4n+1$ Kaboozle cards as follows. Fig. 4 shows an example of the reduction for $F(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_4) \wedge (\overline{x_2} \vee x_3 \vee \overline{x_4})$. Each gray area is a hole in the card, each black line is a fragment of the unique path, and the black circles are the endpoints of the unique path.

Top card: One *top* card is placed at the top of the shuffle pattern, and it represents m clauses. On the top card, two endpoints of the unique path are drawn, and each clause is represented by a hole in the card. Each hole has two dimples corresponding to the borders of the path and that will be extended to one of three possible directions by the variable cards described below.

Variable card: We use $2n$ variable cards. Here, the index i with $1 \leq i \leq n$ is used to represent the ith variable, and the index j with $1 \leq j \leq m$ is used to represent the jth clause. Each card represents either x_i or $\overline{x_i}$. We make m gadgets on the card for the variable x_i as follows.

If neither x_i nor $\overline{x_i}$ appear in clause c_j, the card x_i has a hole at that place. Hence this card has no influence at that place of clause c_j.

If x_i appears in clause c_j, the card x_i has a part of the path at that place. According to the position (first, second, or third literal) in the clause, the path is depicted at top, center, or bottom, respectively, as shown in Fig. 4.

If $\overline{x_i}$ appears in clause c_j, the card x_i has a *cover area* of the path at that place. This white area covers the corresponding path drawn on the variable card corresponding to $\overline{x_i}$, as shown in Fig. 4.

Each variable card $\overline{x_i}$ is symmetric to the variable card x_i, and hence omitted.

Blank card: We use $2n$ blank cards depicted in Fig. 4. They will be used to join variable cards and the top card. They have no influence on the appearance of the variable cards.

We first show that generalized Kaboozle is NP-complete, without requiring a strip folding:

Theorem 2. *Generalized Kaboozle is* NP-*complete, even forbidding reflection and rotation.*

Proof. We use the top card and $2n$ variable cards. Make the cards asymmetric, e.g., by shifting the gadgets on each card a little, to forbid reflecting or rotating the cards (if that is allowed). Clearly, the reduction can be done in a polynomial time.

Because of the pictures of the endpoints of the unique path, the top card must be on top. It is not difficult to see that card x_i has no influence on cards x_j and $\overline{x_j}$ if $i \neq j$. Hence it is sufficient to consider the ordering between each pair x_i and $\overline{x_i}$ for $i = 1, 2, \ldots, n$.

When $F(x_1, \ldots, x_n)$ has a solution, i.e., each clause c_j contains exactly one true literal ℓ_i^j, the card corresponding to the literal activates one of three parts on the card that joins the two endpoints of the parts of path incident to the hole representing c_j in the top card. For example, consider the (wrong) assignment $x_1 = 0, x_2 = 1, x_3 = 0$, and $x_4 = 1$ for $F(x_1, x_2, x_3, x_4)$ from Fig. 4, as shown in Fig. 5. Then we put the card $\overline{x_1}$ over the card x_1, the card x_2 over the card $\overline{x_2}$, and so on. Then, the card $\overline{x_1}$ covers the parts of the path on the card x_1, the card x_2 covers the parts of the path on the card $\overline{x_2}$, and so on. Any two cards corresponding to different variables can be stacked in any order. For example, we can arrange "top", $\overline{x_1}$, x_1, x_2, $\overline{x_2}$; "top", $\overline{x_1}$, x_2, $\overline{x_2}$, x_1; or "top", $\overline{x_1}$, x_2, x_1, $\overline{x_2}$; and so on. For this assignment, the clause $c_1 = (x_1 \vee x_2 \vee x_3)$ satisfies the condition of the 1-in-3 3SAT because only x_2 is true. Hence the hole corresponding to c_1 in the top card is filled and the path is joined properly. On the other hand, all literals are true in the clause c_2, and no literal is true in the clause c_3. Hence the hole corresponding to c_2 produces loops and the path is disconnected at the hole corresponding to c_3.

Therefore, the two endpoints of the path on the top card are joined by one simple path if and only if each c_j contains exactly one true literal. □

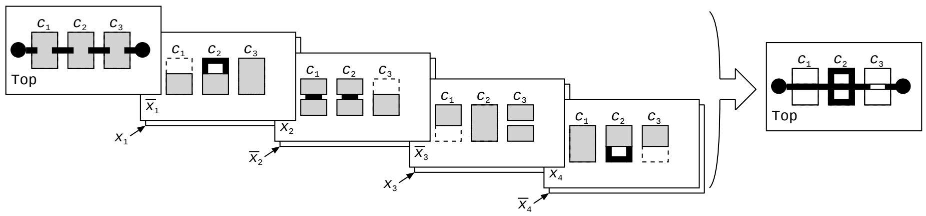

Fig. 5. For $F(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_4) \wedge (\overline{x_2} \vee x_3 \vee \overline{x_4})$, a wrong ordering of the cards that corresponds to a wrong assignment $x_1 = 0, x_2 = 1, x_3 = 0$, and $x_4 = 1$. For this assignment, the first clause c_1 contains one true literal, the second clause c_2 contains three true literals, and the third clause c_3 contains no true literal.

We now turn to the main theorem.

Theorem 3. *Generalized Kaboozle is* NP-*complete even in a strip with fixed mountain-valley pattern.*

Proof. We use the top card, $2n$ variable cards, and $2n$ blank cards. We join these cards into a strip as "x_n-b-x_{n-1}-b-$\cdots$-b-x_2-b-x_1-b-top-b-$\overline{x_1}$-b-$\overline{x_2}$-b-$\cdots$-b-$\overline{x_{n-1}}$-b-$\overline{x_n}$", where "b" means a blank card. Fig. 6 shows the example from Fig. 4). We glue the blank cards upside down, which will be reflected by folding to unit length. The mountain-valley pattern is the shuffle pattern of length n; that is, the creases on either side of the top card are mountain, and from there, the other creases are defined to form two pleats of length n.

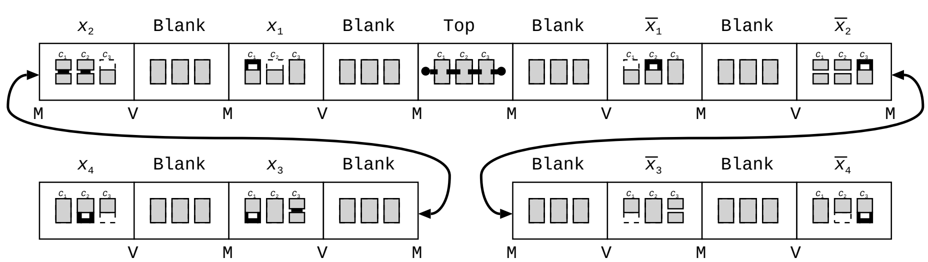

Fig. 6. The cards joined in a strip.

Now, the left pleat of the top card makes the sequence of x_is, and the right pleat makes the sequence of $\overline{x_i}$s. For each pair of x_i and $\overline{x_i}$, we can choose the ordering between the corresponding cards with an appropriate shuffling. This means that we can assign true or false to this variable. Moreover, thanks to the blank cards between the variable cards, we can arrange the ordering of the cards x_i and $\overline{x_i}$ independently for each i. Hence, by Theorem 2 and the property of the shuffle pattern, the constructed Kaboozle strip with fixed mountain-valley pattern has a solution if and only if the 1-in-3 3SAT has a solution. □

Carefully checking the proof of the main theorem, we can also let the mountain-valley pattern be free:

Corollary 1. *Generalized Kaboozle is* NP-*complete even in the strip form and allowing any mountain-valley pattern.*

Proof. We use the same strip in the proof of Theorem 3. Even if the mountain-valley pattern is not specified, the top card should be on top; otherwise, the endpoints of the path disappear. Hence both creases bordering the top card are mountains. If the 1-in-3 3SAT instance has a solution, the constructed Kaboozle puzzle has a solution by the folding in the proof of Theorem 3. On the other hand, if the Kaboozle puzzle has a solution, we can extract the ordering between x_i and $\overline{x_i}$ for each i with $1 \leq i \leq n$ from the folded state. From these orderings, we can construct the solution to the 1-in-3 3SAT instance. □

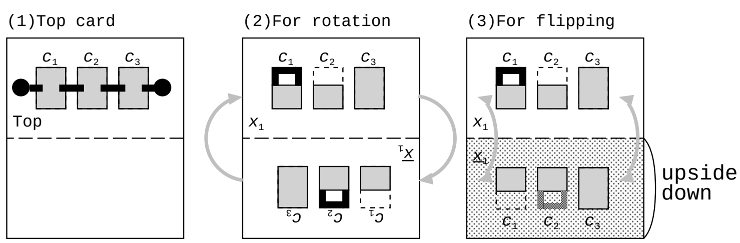

Fig. 7. Gadgets for rotation and reflection.

By combining gadgets, we can show that generalized Kaboozle is also NP-complete if we allow only either rotation or reflection. Note that we can rotate a card 180° by the combination of a horizontal reflection and a vertical reflection. To forbid this kind of cheating with cards, we restrict reflection to be vertical.

Theorem 4. *Generalized Kaboozle is* NP-*complete even if the card ordering is fixed (or free), and (1) only 180° rotation of the cards is allowed, or (2) only vertical reflection of the cards is allowed.*

Proof. As in the proof of Theorem 2, we prepare the top card and $2n$ variable cards. Now, the top card is enlarged to twice of the original cards ; see Fig. 7(1).

Rotation: For each variable x_i, two variable cards x_i and $\overline{x_i}$ are glued so that 180° rotation exchanges them; see Fig. 7(2).

Vertical reflection: For each variable x_i, two variable cards x_i and $\overline{x_i}$ are glued so that a vertical reflection exchanges them; see Fig. 7(3).

Then it is easy to see that the ordering of the cards has no influence, except the top card which should be the top, and the resultant Kaboozle has a solution if and only if the 1-in-3 3SAT instance has a satisfying truth assignment. □

Along similar lines, we can show that silhouette puzzles are NP-complete:

Theorem 5. *Silhouette puzzles are* NP-*complete even if (1) only 180° rotation of the cards is allowed, or (2) only vertical reflection of the cards is allowed.*

Proof. We reduce from regular (not 1-in-3) SAT, mimicking the gadgets in Fig. 7. The top card has one hole per clause, all in the top half of the card. Each variable card reserves the top and bottom halves for the true and false literals; each side has a solid patch for each clause the literal satisfies, and a hole for all other clauses. As in Fig. 7, the top and bottom sides are rotations or vertical reflections of each other according to the variation. A rectangular silhouette is possible if and only if the formula is satisfiable. □

Acknowledgement

The authors thank Yoshio Okamoto for helpful discussions.

References

1. Esther M. Arkin, Michael A. Bender, Erik D. Demaine, Martin L. Demaine, Joseph S. B. Mitchell, Saurabh Sethia, and Steven S. Skiena. When can you fold a map? *Computational Geometry: Theory and Applications*, 29(1):23–46, September 2004.
2. Jean Cardinal, Erik D. Demaine, Martin L. Demaine, Shinji Imahori, Stefan Langerman, and Ryuhei Uehara. Algorithmic folding complexity. In *20th International Symposium on Algorithms and Computation (ISAAC 2009)*, Lecture Notes in Computer Science, vol. 5856, pages 452–461. Springer-Verlag, 2009.
3. Michael R. Garey and David S. Johnson. *Computers and Intractability — A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
4. Jaap Scherphuis. Jaap's Puzzle Page: Transposer / Trixxy / Stained. http://www.jaapsch.net/puzzles/trixxy.htm, 2009.
5. Frank H. Lehman. Puzzle. U.S. Patent 856,196, June 1907.
6. Neil J. A. Sloane. *The On-Line Encyclopedia of Integer Sequences*. http://www.research.att.com/ njas/sequences, 2010.
7. Jerry Slocum and Jack Botermans. *Puzzles Old and New: How to Make and Solve Them*. University of Washington Press, 1988.
8. Rob Stegmann. Rob's Puzzle Page: Folding — Paper/Card. http://home.comcast.net/ stegmann/allother.htm#fold-paper, 2010.
9. Ryuhei Uehara. Stretch minimization problem of a strip paper. In *5th International Conference on Origami in Science, Mathematics and Education (5OSME)*, Singapore, 2010.
10. Chaim Raphael Weinreb. Puzzle. International Patent WO 99/15248 (and GB2345645, EP1021227, US5281986), April 1999. http://www.jaapsch.net/puzzles/patents/wo9915248.pdf