| LETTER |
| --- |

# Generalized Shisen-Sho is NP-Complete*

**Chuzo IWAMOTO**[†a)], ***Member*, Yoshihiro WADA**[†], ***Nonmember, and* Kenichi MORITA**[†], ***Member***

**SUMMARY**  Shisen-Sho is a tile-based one-player game. The instance is a set of 136 tiles embedded on $8 \times 17$ rectangular grids. Two tiles can be removed if they are labeled by the same number and if they are adjacent or can be connected with at most three orthogonal line segments. Here, line segments must not cross tiles. The aim of the game is to remove all of the 136 tiles. In this paper, we consider the generalized version of Shisen-Sho, which uses an arbitrary number of tiles embedded on rectangular grids. It is shown that deciding whether the player can remove all of the tiles is NP-complete.
*key words:*  *NP-complete, computational complexity, one-player game, Shisen-Sho*

## 1. Introduction

Shisen-Sho is a tile-based one-player game. The instance is a set of 136 tiles embedded on $8 \times 17$ rectangular grids (see Fig. 1 for the $4 \times 9$ version of the game). Tiles are numbered 1 through 34, and there are four $l$-labeled tiles for every $l \in \{1, 2, \ldots, 34\}$. The aim of the game is to remove all of the 136 tiles. Two tiles can be removed if they are labeled by the same number and if they are adjacent or can be connected with at most three orthogonal line segments. Here, line segments must not cross tiles.

In Fig. 1 (a), two pairs of tiles labeled with 1 and 9 can be removed, since they are adjacent. A pair of tiles labeled with 8 can also be removed, since they are connected with three orthogonal line segments. On the other hand, a pair of tiles labeled with 6 cannot be removed, since they are connected with four orthogonal line segments. (You can play Shisen-Sho at the web site [1].)
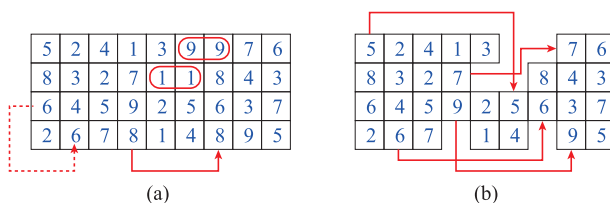


**Fig. 1**  (a) Shisen-Sho on $4 \times 9$ grids. There are three removable pairs of tiles in the initial layout of tiles. (b) After the removal of six tiles, four pairs can be removed.

In this paper, we consider the generalized version of Shisen-Sho, which uses $4p$ tiles embedded on a rectangular grids, where $p$ is an arbitrary positive integer. The tiles are numbered 1 through $p$. For each $l \in \{1, 2, \ldots, p\}$, there are four tiles which are labeled by the same number $l$. We will show that deciding whether the player can remove all of the $4p$ tiles is NP-complete. It is not difficult to show that the generalized Shisen-Sho is in NP, since the game can last at most $2p$ removals of pairs of tiles.

There has been a huge amount of literature on the computational complexity of games and puzzles. In 2009, a survey of games, puzzles, and their complexities was reported by Hearn and Demaine [3]. Very recently, the difficulty rating of Shisen-Sho was experimentally evaluated by using the game posted on a web site [4]. In the current paper, we investigate the computational complexity of the generalized Shisen-Sho from a theoretical approach.

## 2. Reduction from 3SAT to the Generalized Shisen-Sho

### 2.1 3SAT Problem

The definition of 3SAT is mostly from [2]. Let $U = \{x_1, x_2, \ldots, x_n\}$ be a set of Boolean *variables*. Boolean variables take on values 0 (false) and 1 (true). If $x$ is a variable in $U$, then $x$ and $\overline{x}$ are *literals* over $U$. The value of $\overline{x}$ is 1 (true) if and only if $x$ is 0 (false). A *clause* over $U$ is a set of literals over $U$, such as $\{\overline{x_1}, x_3, x_4\}$. It represents the disjunction of those literals and is *satisfied* by a truth assignment if and only if at least one of its members is true under that assignment.

An instance of 3SAT is a collection $f = \{c_1, c_2, \ldots, c_m\}$ of clauses over $U$ such that $|c_j| = 3$ for $1 \leq j \leq m$. The 3SAT problem asks whether there exists some truth assignment for $U$ that simultaneously satisfies all the clauses in $f$. For example, $U = \{x_1, x_2, x_3, x_4\}$, $f = \{c_1, c_2, c_3, c_4\}$, and $c_1 = \{x_1, x_2, x_3\}$, $c_2 = \{x_1, \overline{x_2}, x_4\}$, $c_3 = \{\overline{x_1}, x_3, \overline{x_4}\}$, $c_4 = \{x_2, \overline{x_3}, \overline{x_4}\}$ provide an instance of 3SAT. For this instance, the answer is "yes", since there is a truth assignment $(x_1, x_2, x_3, x_4) = (1, 0, 1, 0)$ satisfying all clauses.

### 2.2 Transformation from an Instance of 3SAT to an Initial Layout of Tiles

We present a polynomial-time transformation from an arbitrary instance $f$ of 3SAT to an initial layout of tiles such that $f$ is satisfiable if and only if all tiles can be removed.
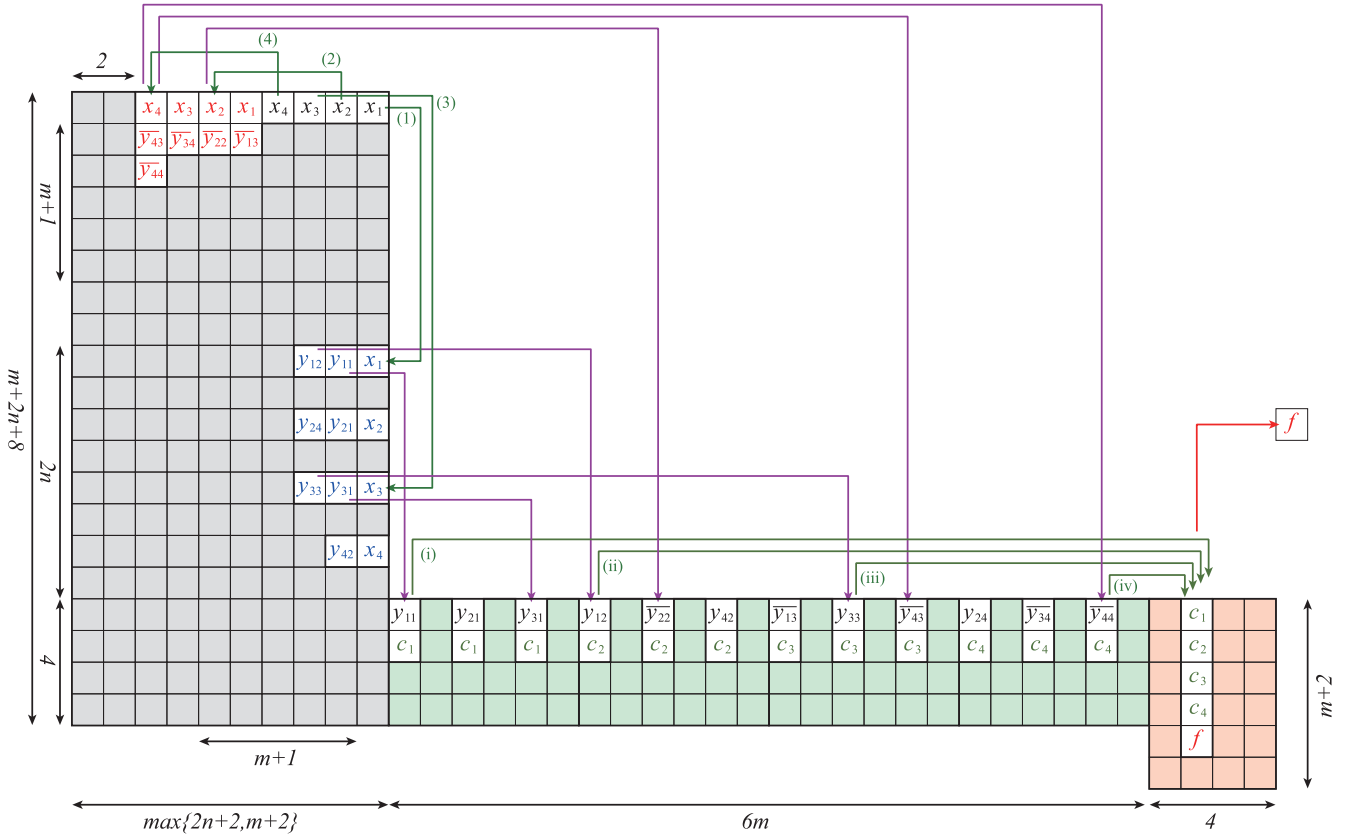
**Fig. 2** Tiling transformed from $f = \{c_1, c_2, c_3, c_4\}$, where $c_1 = \{x_1, x_2, x_3\}$, $c_2 = \{x_1, \overline{x_2}, x_4\}$, $c_3 = \{\overline{x_1}, x_3, \overline{x_4}\}$, and $c_4 = \{x_2, \overline{x_3}, \overline{x_4}\}$. $f$ is satisfiable, since there is a truth assignment $(x_1, x_2, x_3, x_4) = (1, 0, 1, 0)$ satisfying all clauses.

Let $n$ and $m$ be the numbers of variables and clauses of $f$, respectively. Without loss of generality, we assume that $n$ and $m$ are integers divisible by four.

The set of variables $x_1, x_2, \ldots, x_n$ is transformed into $((m + 2n + 8) \times \max\{2n + 2, m + 2\})$ tiles (see Fig. 2). In this tiling, the last $n$ tiles of the first row is labeled by $x_1, x_2, \ldots, x_n$ from right to left, which are followed by $n$ tiles labeled by the same labels $x_1, x_2, \ldots, x_n$. The former (resp. latter) $n$ tiles are called *black $x_i$-tiles* (*red $x_i$-tiles*). In the $(m + 5)$th through $(m + 2n + 4)$th tiles of the last column, $x_1, x_2, \ldots, x_n$ appear alternately. They are called *blue $x_i$-tiles*.

If literal $x_i$ (resp. $\overline{x_i}$) appears in clauses $c_{j_1}, c_{j_2}, \cdots$, then blue (red) $x_i$-tile is adjacent to a sequence of tiles whose labels are $y_{ij_1}, y_{ij_2}, \cdots$ ($\overline{y_{ij_1}}, \overline{y_{ij_2}}, \cdots$). (In Fig. 2, for example, literal $x_1$ appears in $c_1$ and $c_2$, so blue $x_1$ is adjacent to $y_{11}y_{12}$.) The remaining grey plain tiles have different labels. (Later, we will label all grey tiles by a, b, $\cdots$. See the upper-left tiles of Fig. 4.)

The set of clauses $c_1, c_2, \ldots, c_m$ is transformed into $4 \times 6m$ tiles. In the second row of this tiling, the 1st, 3rd, and 5th tiles are labeled with $c_1$, the 7th, 9th, and 11th tiles are labeled with $c_2$, and so on. Each of the three $c_j$-tiles is adjacent to tile $y_{ij}$ (resp. $\overline{y_{ij}}$) in the first row if clause $c_j$ contains literal $x_i$ (resp. $\overline{x_i}$).

The collection of clauses $f = \{c_1, c_2, \ldots, c_m\}$ is trans-

formed into $(m+2) \times 4$ tiles. The first $m+1$ tiles of the second column are labeled with $c_1, c_2, \ldots, c_m, f$. Finally, one more tile labeled with $f$ is added (see the isolated tile $f$ in Fig. 2). A pair of tiles $f$ are called *target tiles*.

For the tiling constructed above (see Fig. 2), the following lemma holds.

**Lemma 1:** *There is a truth assignment for variables $x_1, x_2, \ldots, x_n$ satisfying all clauses $c_1, c_2, \ldots, c_m$ if and only if the pair of target tiles $f$ can be removed.*

*Proof.* For each variable $x_i$, there are three $x_i$-tiles. For variable $x_1$, either (a) a pair of blue and black $x_1$-tiles are removed or (b) a pair of red and black $x_1$-tiles are removed. Removing a blue (resp. red) $x_1$-tile corresponds to the assignment $x_1 = 1$ (resp. $\overline{x_1} = 1$). The same observation holds for each $x_i$, where $1 \le i \le n$ (see arrows (1) through (4) in Fig. 2).

If a blue $x_i$-tile is removed, then tiles having labels $y_{ij_1}, y_{ij_2}, \cdots$ can be removed by pairing them to the corresponding tiles in the $(4 \times 6m)$-tiling. Namely, tile $y_{ij}$ (resp. tile $\overline{y_{ij}}$) can be removed if and only if variable $x_i$ (resp. $\overline{x_i}$) of clause $c_j$ has value 1.

For each $j \in \{1, 2, \ldots, m\}$, clause $c_j$ is satisfied if and only if one of the three $c_j$-tiles in the $(4 \times 6m)$-tiling can be paired to the $c_j$-tile in the $((m + 2) \times 4)$-tiling (see arrows (i) through (iv) in Fig. 2).

Therefore, there is a truth assignment for variables $x_1, x_2, \ldots, x_n$ satisfying all clauses $c_1, c_2, \ldots, c_m$ if and only if the pair of target tiles $f$ can be removed. □

### 2.3 Construction of a Rectangular Tiling in Which There Are Four Tiles Having Each Label

The tiling constructed in the previous section was not a rectangular tiling. The initial configuration of Shisen-Sho must be a rectangular tiling, in which, for every label $l$, there are exactly four $l$-labeled tiles.

We construct an O-shaped tiling (see blue tiles of Fig. 3). For each label $l$, there are four blue $l$-labeled tiles in this O-shaped tiling. (In Fig. 3, $l \in \{0, 1, \ldots, 22, A, B, \ldots, U\}$.)

One can easily verify that all the blue tiles can be removed if the upper-left tile $f$ is removed. (Please disregard "non-blue" tiles inside the O-shaped tiling. Non-blue tiles will be added to the figure in the next paragraph.)

In Fig. 2, there were only three $x_i$-tiles for every $i \in \{1, 2, \ldots, n\}$. Consequently, we add $n$ $x_i$-tiles into the inside

of the O-shaped tiling (see $x_1, x_2, x_3, x_4$ of Fig. 3). For $y_{ij}$-tiles and $\overline{y_{ij}}$-tiles, we add $6m$ tiles into the inside of the O-shaped tiling (see $y_{11}, \overline{y_{11}}, \cdots, \overline{y_{44}}$ of Fig. 3). Since $n$ and $m$ are integers divisible by four, those tiles fill up $(n + 6m)/4$ columns.

The number of grey (resp. green and orange) plain tiles is $(m + 2n + 8) \max\{2n + 2, m + 2\} - 3n - 3m$ (resp. $24m - 6m$ and $4(m + 2) - m - 1$). Those plain tiles have different labels (see labels a, b, $\cdots$, y, z in Fig. 4). Since $n$ and $m$ are integers divisible by four, the total number of plain tiles can be represented as $4k - 1$ for some integer $k$. Consequently, we add $12k - 3$ tiles into the inside of the O-shaped tiling (see tiles a, a, a, b, b, b, $\cdots$, y, y, y, z, z, z, where the last tile z is the first tile of the last column). The second and third tiles of the last column are a pair of tiles $f$. Since the last column has height 3, the tiles considered in the current paragraph fill up the inside area of the O-shaped tiling.

Finally, we fill up the remaining area by four-tile sets so that the whole tiling forms a single rectangle (see Fig. 4). Each yellow tile set consists of four tiles having the same label.



**Fig. 3** O-shaped tiling.



**Fig. 4** Initial layout of the generalized Shisen-Sho.

Assume that $f$ is satisfiable. Trivially, all yellow four-tile sets can be removed. From Lemma 1, if $f$ is satisfiable, then the target tile $f$ placed at the upper-left tile of the O-shaped tiling can be removed. Once the target tile $f$ is removed, then all of the blue tiles, $x_i$-tiles, $y_{ij}$-tiles, $\overline{y_{ij}}$-tiles, plain tiles, and $f$-tiles can be removed. Assume that $f$ is not satisfiable. In this case, the pair of target tiles $f$ cannot be removed from Lemma 1. Therefore, $f$ is satisfiable if and only if all tiles can be removed.

**References**

[1] http://g-giga.com/game-shisensho_en.html

[2] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, New York, NY, USA, 1979.

[3] R.A. Hearn and E.D. Demaine, Games, Puzzles, and Computation, A K Peters Ltd., 2009.

[4] T. Korekawa and Y. Kotani, "Difficulty rating of Shisen-Sho puzzle by a structure of state space," Trans. IPSJ, vol.53, no.6, pp.1617–1624, June 2012 (in Japanese).