The Complexity of Ladder-Lottery Realization Problem

Katsuhisa Yamanaka[1,a)] Takashi Horiyama[2,b)] Takeaki Uno[3,c)] Kunihiro Wasa[3,d)]

Abstract: A *ladder lottery* of a permutation $\pi = (p_1, p_2, \ldots, p_n)$ is a network with n vertical lines and zero or more horizontal lines each of which connects exactly two consecutive vertical lines. The top ends and the bottom ends of the vertical lines correspond to the identity permutation and π, respectively. Each horizontal line corresponds to an intersection of two vertical lines. Suppose that we are given a permutation π of $[n] = \{1, 2, \ldots, n\}$ and a multi-set S of intersections each of which is a pair of elements in $[n]$. Then Ladder-Lottery Realization problem asks whether or not there is a ladder-lottery of π in which each intersection in S appears exactly once. We show that Ladder-Lottery Realization problem is NP-complete. We also present some positive results of Ladder-Lottery Realization and its variant.

1. Introduction

A *ladder lottery*, known as the "Amidakuji" in Japan, is a very common way to obtain a "random" assignment. Japanese kids often use ladder lotteries to determine an assignment in a group. Let us show an example of how to use ladder lotteries. Suppose that, in an elementary school, we have to determine a group leader among n classmates. First, a teacher draws n vertical lines in a notebook and ticks off one of the bottom ends of the vertical lines so that any student cannot predict where the tick-mark is. See Fig. 1(a). Second, the teacher covers the bottom ends of all vertical lines, then the teacher draws some horizontal lines connecting adjacent vertical lines (Fig. 1(b)). Third, each student chooses the top end of a vertical line (Fig. 1(c)). Finally, the teacher takes off the cover. The obtained figure gives an assignment (Fig. 1(d)).

Formally, for a permutation $\pi = (p_1, p_2, \ldots, p_n)$ of $[n] = \{1, 2, \ldots, n\}$, a ladder lottery is a network with n vertical lines (*lines* for short) and zero or more horizontal lines (*bars* for short) each of which connects exactly two consecutive vertical lines. The top ends of lines correspond to the identity permutation $(1, 2, \ldots, n)$. The bottom ends of lines correspond to π. See Fig. 2(a). Each element i in $[n]$ starts from the top end of ith line from the left, and goes down along the line, then whenever i comes to an end of a bar, i goes horizontally along the bar to the other end, then goes down again. Finally, i reaches the bottom end of jth line from the left such that $i = p_j$. We can regard a bar as a modification of the current permutation, and a sequence of such modifications in a ladder lottery always results in the identity permutation.

Ladder lotteries of the reverse permutations have a one-to-one

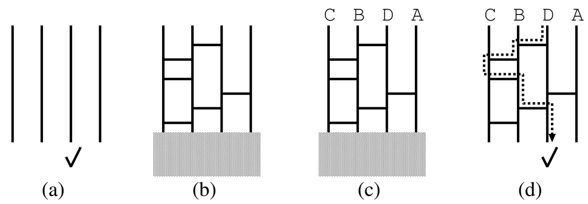

Fig. 1 An example of how to use a ladder lottery. Imagine the situation that we choose a leader among four students A, B, C, and D. (a) four vertical lines and a tick-mark. (b) The tick-mark is hided and six horizontal lines are drawn by a teacher according to his or her intuition. (c) Each student chooses a top end of a vertical line. (d) The result of the obtained assignment. In this assignment, D is a leader.

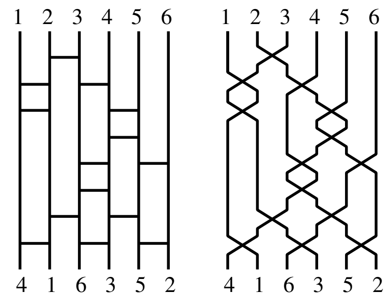

Fig. 2 (a) A ladder lottery of (4,1,6,3,5,2) and (b) its pseudoline drawing.

correspondence to pseudoline arrangements [12]. The route of an element from a top end to a bottom end corresponds to a pseudoline and a bar corresponds to an intersection of two pseudolines. To calculate the number of pseudoline arrangements, some enumeration and counting algorithms of ladder lotteries were presented in [5], [12]. The history of the counting results is shown in the Online Encyclopedia of Integer Sequences [7]. In the area of algebra, a ladder lottery is regarded as a decomposition of a permutation into adjacent transpositions. The top ends of lines correspond to the identity permutation. The bottom ends of lines correspond to a permutation. Each bar corresponds to an adjacent transposition. From these viewpoints, ladder lotteries have been

1 Iwate University, Morioka, Iwate 020–8515, Japan
2 Saitama University, Saitama, Saitama 338–8570, Japan
3 National Institute of Informatics, Chiyoda, Tokyo 101–8430, Japan
a) yamanaka@cis.iwate-u.ac.jp
b) horiyama@al.ics.saitama-u.ac.jp
c) uno@nii.ac.jp
d) wasa@nii.ac.jp

studied as mathematically attractive objects. In recent years, from the viewpoint of theoretical computer science, some problems on ladder lotteries are considered: counting [11], random generation [11], enumeration [5], [10], [11], [12], reconfiguration [3].

A few years ago, Yamanaka et al. [8] proposed the puzzle, called Token Swapping problem: We are given a permutation and a set of allowable transpositions. The Token Swapping problem asks to find a minimum-length decomposition using only transpositions in the set.*1 Recently, this puzzle and its variants have been actively studied [1], [4], [6], [9].

In this paper, we propose a new puzzle regarding ladder lotteries. The purpose of Token Swapping problem is to find a shortest decomposition of a permutation. On the other hand, we consider the problem, called Ladder-Lottery Realization, of constructing a target permutation using compositions of designated transpositions. Let us describe our problem more formally. We are given a target permutation π of and a multi-set S of transpositions. The problem asks whether one can construct the target permutation by composing each transposition in the set exactly once. In this paper, we investigate the computational complexity of Ladder-Lottery Realization problem. We show the NP-completeness of the problem and give some positive results for the problem and its variant.

Due to page limitation, all proofs are omitted.

2. Preliminaries

A *ladder lottery* of a permutation $\pi = (p_1, p_2, \ldots, p_n)$ is a network with n vertical lines (*lines* for short) and zero or more horizontal lines (*bars* for short) each of which connects two consecutive vertical lines. The top ends of the n lines correspond to the identity permutation. The bottom ends of the n lines correspond to π. See Fig. 2(a). Each element i in the identity permutation starts the top end of ith line from the left, and goes down along the line, then whenever i comes to an end of a bar, i goes to the other end and goes down again, then finally i reaches the bottom end of jth line such that $i = p_j$. By representing the route for each element i as a pseudoline and each bar as an intersection of two pseudolines, one can represent a ladder lottery as a drawing of pseudolines. In this paper, for convenience of descriptions, we use the pseudoline drawing to represent a ladder lottery. For example, Fig. 2(b) is the pseudoline drawing of the ladder lottery in Fig. 2(a). From now on, if it is clear from the context, we call the route of an element as a pseudoline. Clearly, we can regard that a pseudoline in the pseudoline drawing of a ladder lottery forms a y-monotone curve. Hence, in the following, we assume that any pseudoline is y-monotone.

Now, let us define Ladder-Lottery Realization problem. Suppose that we are given a permutation $\pi = (p_1, p_2, \ldots, p_n)$ of $[n]$ and a multi-set S of intersections each of which is a pair of elements in $[n]$. Then Ladder-Lottery Realization asks whether or not there is a ladder-lottery of π in which each intersection in S appears exactly once. For example, suppose that we are given the

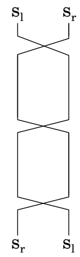

Fig. 3 Room gadget with 4 rooms.

permutation (4,1,6,3,5,2) and the multi-set

$$\{\{1,3\}^2, \{1,4\}, \{2,3\}, \{2,4\}^3, \{2,5\}^3, \{2,6\}, \{3,4\}, \{3,6\}, \{5,6\}^3\}$$

of intersections, where $\{i, j\}^k$ means k $\{i, j\}$s. Then, the answer is yes, since the ladder lottery in Fig. 2(a) is a solution.

3. Hardness of ladder-lottery realization

We give a reduction from a well-known NP-complete problem One-in-Three 3SAT:

Problem: One-in-Three 3SAT [2]
Instance: Set X of variables, collection C of clauses over X such that each clause in C contains exactly three literals.
Question: Is there a truth assignment for X such that each clause in C has exactly one true literal?

Let $I_S = (X, C)$ be an instance of One-in-Three 3SAT, where $X = \{x_1, x_2, \ldots, x_n\}$ is a set of variables and $C = \{C_1, C_2, \ldots, C_m\}$ is a collection of clauses. We may assume without loss of generality that any clause $C_i \in C$ does not contain both the positive and the negative literals of any variable in X. We denote by n and m the numbers of variables and clauses, respectively. We are going to construct an instance $I_R = (\pi, S)$ of Ladder-Lottery Realization from I_S, where π is a permutation and S is a multi-set of intersections.

To reduce I_S to I_R, we prepare the gadgets: a room gadget, a drawer gadget, a variable gadget, a clause gadget, and an assignment gadget. Let us explain these gadgets one by one.

Room gadget

First, we define a room gadget. The *room gadget* consists of two pseudolines s_ℓ, s_r and a multi-set $S_R(I_S)$ of intersections. The top ends of the two pseudolines appear in the order s_ℓ, s_r and their bottom ends appear in the reverse order. We define the multi-set of intersections so that the two pseudolines form $4n$ regions:

$$S_R(I_S) := \{s_l, s_r\}^{4n-1}.$$

Then the two pseudolines intersect $4n-2$ closed regions and the top and bottom regions enclosed by s_ℓ and s_r. See Fig. 3. We call the ith region from the top the *ith room*.

Later, we use two rooms to represent an assignment of each variable. More precisely, we use the $(4i-3)$rd and $(4i-1)$th rooms to represent the assignment of the variable x_i for $i = 1, 2, \ldots, n$.

Drawer gadget

We next define a *drawer gadget*, which consists of $4n$ pseudo-

*1 Actually, the Token Swapping problem is defined as a puzzle consisting of n tokens on n-vertex graph where each token has a distinct starting vertex and a distinct target vertex it wants to reach, and the only allowed transformation is to swap the tokens on adjacent vertices [8].

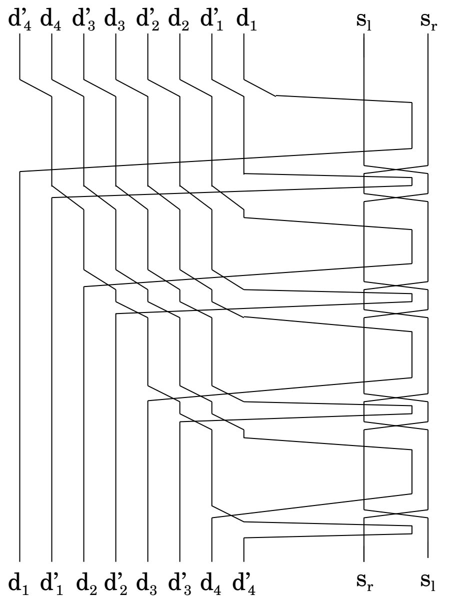

Fig. 4 Drawer gadget.

lines $d_1, d'_1, d_2, d'_2, \dots, d_{2n}, d'_{2n}$ and a multi-set $S_D(I_S)$ of intersections. The top ends of the pseudolines are arranged in the order $d'_{2n}, d_{2n}, d'_{2n-1}, d_{2n-1}, \dots, d'_1, d_1$ in the left region of the pseudolines of the room gadget and their bottom ends are arranged in the reverse order, namely $d_1, d'_1, d_2, d'_2, \dots, d_{2n}, d'_{2n}$ (see Fig. 4).

We define $S_D(I_S)$ such that d_i and d'_i for each $i = 1, 2, \dots, 2n$ come to the $(2i-1)$th and $(2i)$th rooms and leave the rooms, respectively. Besides, every pseudoline in the drawer gadget crosses with all other pseudolines except itself in the gadget exactly once. The formal definition of $S_D(I_S)$ is as follows:

$$\begin{aligned}
&S_D(I_S) := \\
&\quad \{\{d_i, d_{i'}\}, \{d_i, d'_{i'}\} \mid i, i' = 1, 2, \dots, 2n \text{ and } i < i'\} \\
&\quad \cup \{\{d_i, d'_i\} \mid i = 1, 2, \dots, 2n\} \\
&\quad \cup \{\{d'_i, d_{i'}\}, \{d'_i, d'_{i'}\} \mid i, i' = 1, 2, \dots, 2n \text{ and } i < i'\} \\
&\quad \cup \{\{d_i, s_\ell\}^2 \mid i = 1, 2, \dots, 2n\}. \\
&\quad \cup \{\{d'_i, s_r\}^2 \mid i = 1, 2, \dots, 2n\}.
\end{aligned}$$

Fig. 4 shows an example of pseudolines of a drawer gadget and a room gadget. From the definition of $S_D(I_S)$, one can observe the form of a pseudoline in the drawer gadget, as follows. First, d_i for each $i = 1, 2, \dots, 2n$ crosses with every $d_{i'}$ and $d'_{i'}$ with $i' < i$. Then d_i crosses with s_ℓ two times. That is, d_i comes to $(2i-1)$th room and leaves it. Then d_i crosses with every $d_{i'}$ with $i' > i$ and every $d'_{i''}$ with $i'' \geq i$. As a result, the bottom end of d_i is $(2i-1)$th one from the left among the pseudolines of the drawer gadget. The shape of d'_i for each $i = 1, 2, \dots, 2n$ is similar.

Now, we explain why d_i and d'_i for $i = 1, 2, \dots, 2n$ form the above shape more formally. For any y-coordinate, a pseudoline d_i (and d'_i) is *rightmost* if, in the y-coordinate, the x-coordinate of d_i (and d'_i) is the largest among all the pseudolines in a drawer gadget. The *rightmost y-coordinate set* of d_i (and d'_i) is the set of the y-coordinates in which d_i (and d'_i) is rightmost. From the definition of a drawer gadget, the pseudolines in the drawer gadget cross each other exactly once and the order of the bottom ends of the pseudolines is the reverse order of their top ends. Hence, it can be observed that a rightmost y-coordinate set of a pseudoline always forms an open interval. Since s_ℓ crosses with $d_1, d_2, \dots, d_{2n}$ and does not cross with $d'_1, d'_2, \dots, d'_{2n}$, s_ℓ crosses with d_i in a y-coordinate in the rightmost y-coordinate set of d_i. Similarly, s_r crosses with d'_i in a y-coordinate in the rightmost y-coordinate set of d'_i. Therefore, the drawing of the pseudolines of a drawer gadget and a room gadget is unique, as shown in Fig. 4.

Variable gadget

Here, let us define a *variable gadget* consisting of n pseudolines and a multi-set $S_X(I_S)$ of intersections. We create a pseudoline $p(x_i)$ for each variable x_i, and arrange their top ends in the order $p(x_1), p(x_2), \dots, p(x_n)$, and all the top ends appear in the right of s_r. We also define the order of their bottom ends as the same one.

Let us explain the outline of the form of $p(x_i)$ (Fig. 5). $p(x_i)$ crosses with d_{2i-1} and d_{2i} but does not cross with s_ℓ. Hence, $p(x_i)$ crosses the two pseudolines in only the coresponding rooms. First, the pseudoline $p(x_i)$ crosses with other pseudolines to approach the room gadget. Then, $p(x_i)$ comes to and leaves two rooms one by one. In the rooms, $p(x_i)$ crosses with d_{2i-1} and d_{2i}. Finally, $p(x_i)$ crosses with other pseudolines to go back to its the original position. Now, we define the multi-set $S(p(x_i))$ of intersections for $p(x_i)$ as follows:

$$\begin{aligned}
S(p(x_i)) :=& \{p(x_i), s_r\}^4 \cup \{p(x_i), d_{2i-1}\}^2 \cup \{p(x_i), d_{2i}\}^2 \\
&\cup \bigcup_{i'=1}^{i-1} \{p(x_i), p(x_{i'})\}^2.
\end{aligned}$$

Let us explain the shape of $p(x_i)$ more carefully. The multi-set $S(p(x_i))$ does not include $\{p(x_i), s_\ell\}$, and hence $p(x_i)$ cannot enter the left region of s_ℓ. However, $S(p(x_i))$ includes both $\{p(x_i), d_{2i-1}\}^2$ and $\{p(x_i), d_{2i}\}^2$. Hence, $p(x_i)$ comes to the $(4i-3)$rd and $(4i-1)$th rooms to cross with d_{2i-1} and d_{2i}, respectively. To approach the rooms, $p(x_i)$ crosses with $p(x_{i-1})$, $p(x_{i-2})$, $\dots$, $p(x_1)$. Then, $p(x_i)$ arrives at the region next to the target rooms. First, $p(x_i)$ comes to the $(4i-3)$rd room, crosses with d_{2i-1} two times in the room, and leaves the room. Next, $p(x_i)$ comes to the $(4i-1)$th room, crosses with d_{2i} two times in the room, and leaves the room. Then, to go back to the original position, $p(x_i)$ crosses with $p(x_1)$, $p(x_2)$, $\dots$, $p(x_{i-1})$ again.

We show an example in Fig. 5. Note that, since $p(x_i)$ does not cross with s_ℓ, it has to cross with pseudolines of a drawer gadget only in the rooms to which the pseudolines come.

Now, let us define the multi-set of intersections of a variable gadget:

$$S_X(I_S) := \bigcup_{i=1}^{n} S(p(x_i)).$$

Clause gadget

A *clause gadget* consists of m pseudolines corresponding to the clauses in C and a multi-set $S_C(I_S)$ of intersections. We create a pseudoline $p(C_j)$ for each clause $C_j \in C$. The order of the top ends of the pseudolines is $p(C_1), p(C_2), \dots, p(C_m)$ between the top ends of s_r and $p(x_1)$ (See Fig. 6). The order of the bottom

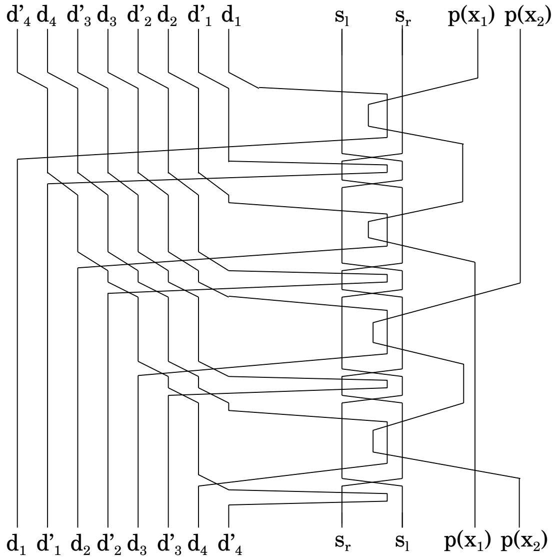

Fig. 5 An example of a variable gadget for $n = 2$.

ends of the pseudolines is the same as the top ends. The bottom ends are arranged between the bottom ends of s_r and $p(x_1)$ (See Fig. 6).

We design a multi-set of intersections for $p(C_j)$ for $j = 1, 2, \ldots, m$ so that $p(C_j)$ forms the shape below. If C_j includes a positive literal of x_i, then $p(C_j)$ comes to and leaves the $(4i-3)$rd room. If C_j includes a negative literal of x_i, $p(C_j)$ comes to and leaves the $(4i-1)$th room. Otherwise, C_j includes neither the positive nor negative literals of x_i, $p(C_j)$ comes to neither the $(4i-3)$rd nor $(4i-1)$th rooms. To force $p(C_j)$ to be such a shape, we define a multi-set of intersections, as follows. We denote by $L(C_j)$ the set of literals in C_j. Let $L(C_j) = \{\ell_{j,1}, \ell_{j,2}, \ell_{j,3}\}$. For each literal $\ell_{j,p}$, $p \in \{1, 2, 3\}$, we define the following multi-set of intersections.

$$\begin{aligned} S(\ell_{j,p}, C_j) :=& \{\{p(C_j), p(C_{j'})\}^2 \mid j' < j \wedge \ell_{j,p} \notin L(C_{j'})\} \\ &\cup \{\{p(C_j), d_{2i-1}\}^2 \mid \ell_{j,p} = x_i\} \\ &\cup \{\{p(C_j), d_{2i}\}^2 \mid \ell_{j,p} = \overline{x_i}\} \\ &\cup \{\{p(C_j), s_r\}^2\} \end{aligned}$$

The intersections in the first set of $S(\ell_{j,p}, C_j)$ are used to approach the room gadget corresponding to $\ell_{j,p}$. If $\ell_{j,p} \in L(C_{j'})$ holds, $p(C_j)$ and $p(C_{j'})$ has no intersection. The intersections in the second and third sets are used to force $p(C_j)$ to come to the rooms corresponding to the literals of x_i.

Besides, we define the following multi-set of intersections for $p(C_j)$ and $p(x_i)$:

$$\begin{aligned} &S(\ell_{j,p}, C_j, x_i) := \\ &\quad \{\{p(C_j), p(x_i)\}^4 \mid \ell_{j,p} \neq x_i \wedge \ell_{j,p} \neq \overline{x_i}\} \\ &\quad \cup \{\{p(C_j), p(x_i)\}^2 \mid \ell_{j,p} = x_i \vee \ell_{j,p} = \overline{x_i}\} \end{aligned}$$

The intersections above are used so that $p(x_i)$ comes to the corresponding the room gadget.

Now we define the set of intersections for clauses, as follows:

$$S_C(I_S) := \left(\bigcup_{j=1}^{m}\bigcup_{p=1}^{3} S(\ell_{j,p}, C_j)\right) \cup \left(\bigcup_{i=1}^{n}\bigcup_{j=1}^{m}\bigcup_{p=1}^{3} S(\ell_{j,p}, C_j, x_i)\right).$$

We give an example shown in Fig. 6. The example shows an reduced instance from the One-in-Three 3SAT instance (X, C), where $X = \{x_1, x_2, x_3, x_4\}$, $C = \{C_1, C_2\}$, $C_1 = (\overline{x_1} \vee x_2 \vee \overline{x_3})$, and $C_2 = (x_2 \vee x_3 \vee x_4)$.

Assignment gadget

The last gadget is the one for representing a truth-false assignment of variables. We define an *assignment gadget* consisting of a pseudoline a and a set of intersections for a. The top and bottom ends of a are respectively located in the left of d'_{2n} and d_1 (see Fig. 6). We define that a crosses with each $p(x_i)$ twice for $i = 1, 2, \ldots, n$ and a crosses with s_ℓ $2n$ times but does not cross with s_r to make a cross with each $p(x_i)$ in either $(4i-3)$th or $(4i-1)$th room. If a crosses with $p(x_i)$ in $(4i-3)$rd room, then it means that x_i is assigned true. Otherwise, if a crosses with $p(x_i)$ in $(4i-1)$th room, then it means that x_i is assigned false. Besides, we force that a crosses with each $p(C_j)$ two times. This corresponds to make the clause C_j true. The pseudoline a touches each C_j exactly once, and hence this assignment corresponds to a solution of an instance of One-in-Three 3SAT. We can define the multi-set of intersections which implements such shape of a:

$$\begin{aligned} S_A(I_S) :=& \left(\bigcup_{i=1}^{n}\{a, p(x_i)\}^2\right) \cup \left(\bigcup_{i=1}^{2n}(\{a, d_i\}^{2n}, \{a, d'_i\}^{2n})\right) \\ &\cup \left(\bigcup_{j=1}^{m}\{a, C_j\}^2\right) \cup \{a, s_\ell\}^{2n}. \end{aligned}$$

The first term is for the intersections with $p(x_i)$ for each $i = 1, 2, \ldots, n$. The second term is the intersections with the pseudolines in the drawer gadget to approach the rooms and to go back to the original position. Note that a does not have to go back to the leftmost region for each entrance to a room. In Fig. 6, a goes back to the leftmost region immediately after each entrance to a room. This is just an example of the form of a. The third term is for the intersections with the pseudolines in the clause gadget. The last term is for the intersections with s_ℓ to come to rooms. The pseudoline a cannot go inside the right region of s_ℓ since there is no intersection $\{a, s_\ell\}$. Hence, a has to cross with the pseudolines of the variables and the clauses in the rooms.

Now, we are ready to describe a reduced instance of Ladder-Lottery Realization. Given an instance $I_S = (X, C)$ of One-in-Three 3SAT, we construct an instance $I_R = (\pi(I_S), S(I_S))$, where

$$\begin{aligned} \pi(I_S) = (&a, d_1, d'_1, d_2, d'_2, \ldots, d'_{2n}, d_{2n}, s_r, s_\ell, \\ &p(C_1), p(C_2), \ldots, p(C_m), \\ &p(x_1), p(x_2), \ldots, p(x_n)) \end{aligned}$$

and

$$S(I_S) = S_R(I_S) \cup S_D(I_S) \cup S_X(I_S) \cup S_C(I_S) \cup S_A(I_S).$$

Using the reduction above, one can show NP-completeness of

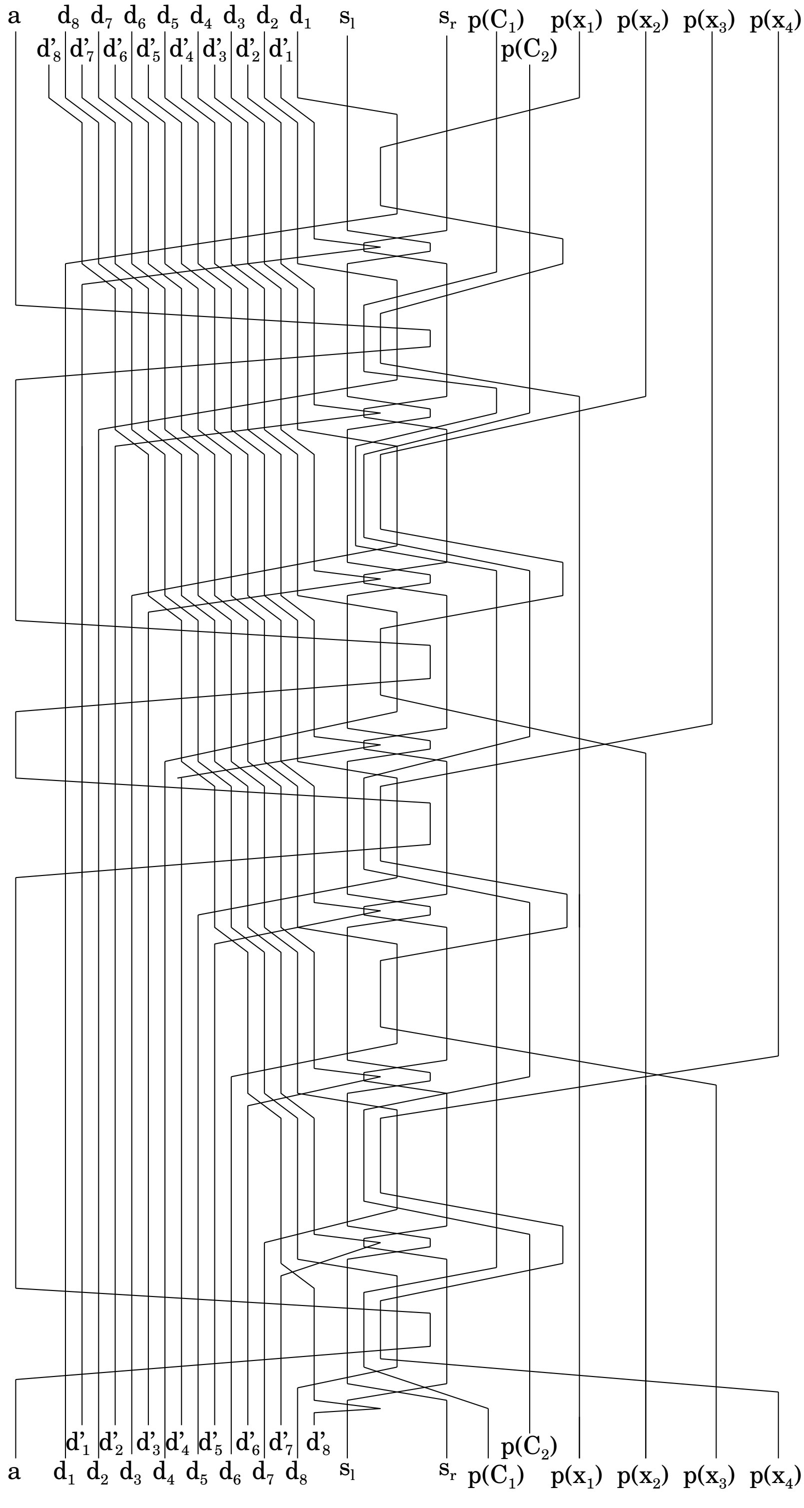

Fig. 6 Reduced instance from (X, C) of a One-in-Three 3SAT instance, where $X = \{x_1, x_2, x_3, x_4\}$, $C = \{C_1, C_2\}$, $C_1 = (\overline{x_1} \vee x_2 \vee \overline{x_3})$, and $C_2 = (x_2 \vee x_3 \vee x_4)$. The assignment gadget represents $(x_1, x_2, x_3, x_4) = (0, 0, 1, 0)$.

Ladder-Lottery Realization.

Theorem 1 Ladder-Lottery Realization is NP-complete.

Proof. Clearly, Ladder-Lottery Realization is in NP. Now, let us show NP-hardness of Ladder-Lottery Realization below.

Let $I_S = (X, C)$ be an instance of One-in-Three 3SAT, where $X = \{x_1, x_2, \ldots, x_n\}$ is a set of variables and $C = \{C_1, C_2, \ldots, C_m\}$ is a collection of clauses. Let $I_R = (\pi(I_S), S(I_S))$ be the instance of Ladder-Lottery Realization obtained from I_S using the reduction in Section 3. Clearly, I_R can be constructed in polynomial time.

Suppose that I_S is a yes-instance. Then I_S has a truth assignment such that each clause has exactly one true literal. If x_i is true, then we make the assignment pseudoline a cross with $p(x_i)$ and the pseudolines of the clauses including the positive literal of x_i in the $(4i-3)$rd room. Otherwise, x_i is false, then we make a cross with $p(x_i)$ and the pseudolines of the clauses including the negative literal of x_i in the $(4i-1)$th room. Since the truth assignment is a solution of One-in-Three 3SAT, the pseudoline of each clause crosses with a exactly two times in a room. Hence, we have a ladder lottery which is a solution of I_R.

Now, let us assume that I_R is a yes-instance. Let L be a solution, namely a ladder lottery, of I_R. The assignment pseudoline a has to cross with the pseudolines of the variables and the clauses in rooms, because a does not intersect with s_r. The pseudolines in a drawer gadget makes $p(x_i)$ for each x_i appear only in the $(4i-3)$rd and $(4i-1)$th rooms. Each clause gadget $p(C_j)$ appears in only exactly three rooms in which literals in C_j appear. Here, the assignment pseudoline must cross with $p(x_i)$ twice in either the $(4i-3)$rd or the $(4i-1)$th room. This gives a truth-false assignment of variables. If the intersection $\{a, p(x_i)\}$ appears in the $(4i-3)$rd room, x_i is set to true. Otherwise, x_i is set to false. Besides, a crosses with $p(C_j)$ exactly two times. Therefore the truth assignment obtained from L is a solution of I_S. □

4. Positive result

In this section, we give positive results. Let $I_R = (\pi, S)$ be an instance of Ladder-Lottery Realization, where π is a permutation of $[n]$ and S is a multi-set of intersections. If $\{i, j\}^k \in S$, we say that the *multiplicity* of $\{i, j\}$ in S is k.

Theorem 2 Let $I_R = (\pi, S)$ be an instance of Ladder-Lottery Realization. If the multiplicity of every intersection in S is 1, one can determine whether or not I_R is a yes-instance in polynomial time.

Proof. It can be observed that, any permutation has a unique multi-set of intersections such that the multiplicity of every intersection is 1. Actually, such a multi-set is equivalent to the set of inversions of a permutation. Hence, to solve I_R, it is sufficient to check whether the set of inversions of π is equivalent to S. □

References

[1] É. Bonnet, T. Miltzow, and P. Rzążewski. Complexity of token swapping and its variants. *Algorithmica*, pages 1–27, Oct 2017.

[2] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, 1979.

[3] T. Horiyama, K. Wasa, and K. Yamanaka. Reconfiguring optimal ladder lotteries. In *Proceedings of the 10th Japanese-Hungarian Symposium on Discrete Mathematics and Its Applications*, pages 217–224, May 2017.

[4] J. Kawahara, T. Saitoh, and R. Yoshinaka. The time complexity of the token swapping problem and its parallel variants. In *Proceeding of The 11th International Conference and Workshops on Algorithms and Computation*, volume 10167 of *Lecture Notes in Computer Science*, pages 448–459, 2017.

[5] J. Kawahara, T. Saitoh, R. Yoshinaka, and S. Minato. Counting primitive sorting networks by πdds. *Hokkaido University, Division of Computer Science, TCS Technical Reports*, TCS-TR-A-11-54, 2011.

[6] T. Miltzow, L. Narins, Y. Okamoto, G. Rote, A. Thomas, and T. Uno. Approximation and hardness of token swapping. In *Proceeding of 24th Annual European Symposium on Algorithms, ESA 2016*, pages 66:1–66:15, 2016.

[7] N. Sloane. The on-line encyclopedia of integer sequences. Published electronically at `https://oeis.org/A006245`. Accessed: 2018-05-02.

[8] K. Yamanaka, E. D. Demaine, T. Ito, J. Kawahara, M. Kiyomi, Y. Okamoto, T. Saitoh, A. Suzuki, K. Uchizawa, and T. Uno. Swapping labeled tokens on graphs. *Theoretical Computer Science*, 586:81–94, 2015.

[9] K. Yamanaka, T. Horiyama, D. Kirkpatrick, Y. Otachi, T. Saitoh, R. Uehara, and Y. Uno. Swapping colored tokens on graphs. *Theoretical Computer Science*, 729:1–10, 2018.

[10] K. Yamanaka and S. Nakano. Efficient enumeration of all ladder lotteries with k bars. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 97-A(6):1163–1170, 2014.

[11] K. Yamanaka and S. Nakano. Enumeration, counting, and random generation of ladder lotteries. *IEICE Transactions Information and Systems*, 100-D(3):444–451, 2017.

[12] K. Yamanaka, S. Nakano, Y. Matsui, R. Uehara, and K. Nakada. Efficient enumeration of all ladder lotteries and its application. *Theoretical Computer Science*, 411:1714–1722, 2010.