Reconfiguration of Labeled Matchings in Triangular Grid Graphs

Naonori Kakimura
Department of Mathematics, Keio University, Yokohama, Japan

Yuta Mishima
The Japan Research Institute, Limited, Tokyo, Japan

Abstract

This paper introduces a new reconfiguration problem of matchings in a triangular grid graph. In this problem, we are given a nearly perfect matching in which each matching edge is labeled, and aim to transform it to a target matching by sliding edges one by one. This problem is motivated to investigate the solvability of a sliding-block puzzle called "Gourds" on a hexagonal grid board, introduced by Hamersma et al. [ISAAC 2020]. The main contribution of this paper is to prove that, if a triangular grid graph is factor-critical and has a vertex of degree 6, then any two matchings can be reconfigured to each other. Moreover, for a triangular grid graph (which may not have a degree-6 vertex), we present another sufficient condition using the local connectivity. Both of our results provide broad sufficient conditions for the solvability of the Gourds puzzle on a hexagonal grid board with holes, where Hamersma et al. left it as an open question.

2012 ACM Subject Classification Theory of computation → Graph algorithms analysis

Keywords and phrases combinatorial reconfiguration, matching, factor-critical graphs, sliding-block puzzles

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2024.43

Related Version *arXiv Version*: `https://arxiv.org/abs/2409.11723` [19]

Funding *Naonori Kakimura*: Supported by JSPS KAKENHI Grant Numbers JP22H05001, JP20H05795, and 23K21646, Japan and JST ERATO Grant Number JPMJER2301, Japan.

Acknowledgements The authors are grateful to Yushi Uno for valuable discussions. We are also grateful for the anonymous referees of ISAAC 2024 for their helpful comments.

1 Introduction

Combinatorial reconfiguration is a fundamental research subject that studies the solution space of combinatorial problems. A typical example is solving sliding-block puzzles such as the 15-puzzle. The 15-puzzle can be viewed as the transformation between the arrangement of puzzle pieces, and the goal is to transform an initial arrangement of pieces to a given target arrangement. Combinatorial reconfiguration has applications in a variety of fields such as mathematical puzzles, operations research, and computational geometry. See the surveys by Nishimura [23] and van den Heuvel [28].

Hamersma et al. [13] introduced a new sliding-block puzzle on a hexagonal grid, which they call *Gourds*. The name "gourd" refers to the shape of the puzzle pieces, which are essentially 1×2 pieces on a board. Like in the 15-puzzle, only one grid cell is empty. The goal is to obtain a target configuration of pieces on the board by moving pieces one-by-one, similar to the 15-puzzle. Here we allow a piece to make three different kinds of moves: slide, turn, and pivot (see [13] for the details). They characterized hexagonal grid boards without holes such that the Gourds puzzle[1] is always solvable, and left it as a main open question to characterize boards *with holes*.

[1] In this paper, the Gourds puzzle refers to the numbered type in [13] where each piece has numbers.

35th International Symposium on Algorithms and Computation (ISAAC 2024).
Editors: Julián Mestre and Anthony Wirth; Article No. 43; pp. 43:1–43:16
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Motivated by the study of the Gourds puzzle, we introduce a reconfiguration problem of matchings in a triangular grid graph. In the problem, we are given a matching that exposes only one vertex, which is called a *nearly perfect matching*. Each matching edge, which corresponds to a puzzle piece, is labeled. We are allowed to slide a matching edge toward the exposed vertex. The goal is to move matching edges one-by-one to obtain a target labeled matching. It should be emphasized that each edge in the given matching has to be moved to the edge with the same label in the target matching. See Section 2 for the formal definition. We remark that our problem can be defined on a general graph, which may be of independent interest. The problem setting is different from the matching reconfiguration problems studied in the literature. See Section 1.1.

In this paper, we investigate the reconfigurability of the above reconfiguration problem of labeled matchings on a triangular grid graph. In particular, we aim to characterize a triangular grid graph such that any two labeled matchings can be reconfigured to each other. We call such a graph *reconfigurable*.

As mentioned in Section 3, it is not difficult to observe that, if a graph is reconfigurable, then it is 2-connected and factor-critical. A graph is *factor-critical* if it has a nearly perfect matching that exposes any vertex. These two conditions, however, are not sufficient, as there exists a 2-connected factor-critical graph that is not reconfigurable.

The main contribution of this paper is to prove that, if a 2-connected factor-critical triangular grid graph has at least one vertex of degree 6, then it is reconfigurable. Our results can be adapted to the Gourds puzzle by taking the dual of a triangular grid graph, which implies that the Gourds puzzle can always be solved when at least one hexagonal cell on the board does not touch the holes or the outer face.

The key idea to prove the main result is to exploit the ear decomposition in matching theory. A factor-critical graph is known to have a constructive characterization using ear decomposition with odd paths and cycles. Using the ear structure, we show that, if an ear decomposition starts from a reconfigurable subgraph, then we can recursively find reconfiguration steps between any two labeled matchings. However, every ear decomposition does not necessarily satisfy the above assumption. We then investigate the matching structure of a triangular grid graph to identify simple reconfigurable subgraphs such that every triangular grid graph with a vertex of degree 6 admits an ear decomposition starting from one of them.

In addition, for a triangular grid graph (which may not have a vertex of degree 6), we present another sufficient condition for the reconfigurability using the local connectivity. A graph is said to be *locally-connected* if the neighbor vertices of any vertex induce a connected graph. We prove that, if a triangular grid graph is locally-connected, but not isomorphic to the Star of David graph (Figure 1), then it is reconfigurable. Moreover, we show that, for a graph with $2n+1$ vertices, we can find in polynomial time reconfiguration steps with length $O(n^3)$.

The characterization for the Gourds puzzle by Hamersma et al. [13] implies that a triangular grid graph is reconfigurable if it is 2-connected, but not isomorphic to the Star of David graph, and has no *holes*, where a hole is a face with boundary length at least 6. Our conditions, which allow to have a hole, are much broader than theirs, as the local connectivity and the 2-connectivity are equivalent for a graph with no holes.

Due to the space limitation, we omit the proofs of statements with the symbol $\star$ marks, which may be found in the full version of this paper [19].

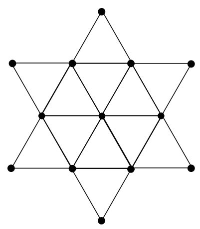

Figure 1 The Star of David graph.

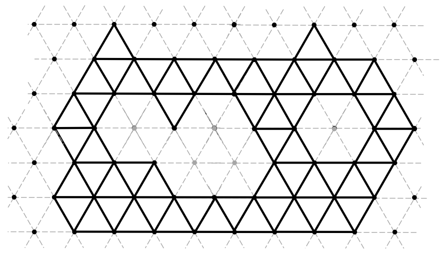

Figure 2 A triangular grid graph with 2 holes.

1.1 Related Work

A factor-critical graph plays an important role in matching theory. It is known that any non-bipartite graph can be decomposed in terms of maximum matching, called the *Gallai-Edmonds* decomposition. It essentially decomposes a given graph into factor-critical graphs, graphs with perfect matchings, and bipartite graphs. Also, factor-critical graphs are used to describe facets of the matching polytope of a given graph. See [21, 26] for the details.

Sliding-block puzzles have been investigated in both recreational mathematics and algorithms research. The 15-puzzle was introduced as a prize problem by Sam Loyd in 1878 [27]. In the 15-puzzle, it is characterized by odd/even permutations whether any configuration can be realized or not [18]. However, it is NP-complete for $n \times n$ boards to compute the smallest number of steps to reach a given configuration [10, 24]. There are many variants of the 15-puzzle such as Rush Hour [6, 11] and rolling-block puzzles [7]. Many puzzles have been shown NP-hard or PSPACE-hard (see e.g., [15]).

In the literature of combinatorial reconfiguration, the reconfiguration of matchings has been studied extensively. Ito et al. [16] initiated to study a reconfiguration problem of matchings. The aim is to decide whether a given matching can be transformed to a target matching by adding/removing a matching edge in each step. They showed that the problem can be solved in polynomial time with the aid of the Gallai-Edmonds decomposition. On reconfigurating perfect matchings, Ito et al. [17] studied the shortest transformation of perfect matchings by taking the symmetric difference along an alternating cycle, motivated by the study of a diameter of the matching polytope (see also [8, 25]). Bonamy et al. [3] restricts the length of alternating cycles used in the transformations to be 4. We remark that all the above mentioned problems aim to transform an initial (perfect) matching to a target one in which their matching edges are *not* labeled.

By taking the line graph of a given graph, reconfiguration problems of matchings can be viewed as reconfiguration problems of independents sets. Our problem setting is related to its variant known as the token sliding problem. The token sliding problem is PSPACE-complete, even on restricted graph classes such as planar graphs [14]. On the other hand, the problem can be solved in linear time on trees [9], and it is fixed-parameter tractable on bounded degree graphs [2]. See also [5] for the survey on the independent set and dominating set reconfiguration problems. Another related problem is the token swapping problem. In the problem, we are given tokens on each vertex of a graph, and we want to move every token to its target position by swapping two adjacent tokens. See e.g., [1, 4, 20, 22] and references therein.

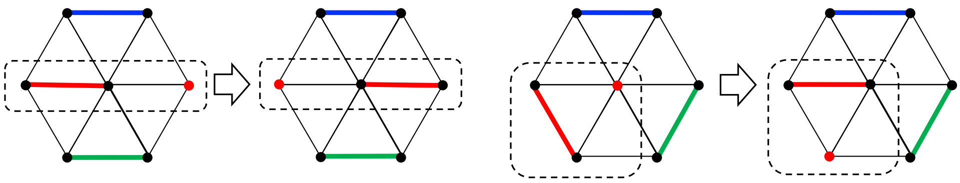

Figure 3 Slide operations. The colored, thick edges correspond to pieces.

2 Preliminaries

Let $G = (V, E)$ be an undirected graph with $2n + 1$ vertices. For a vertex u, we denote by $N(u)$ the set of vertices adjacent to u. For a vertex subset X, the subgraph induced by X in G is denoted by $G[X]$. A path or a cycle is *odd* if it has an odd number of edges.

A *matching* is a subset of edges that have no common end vertices. A matching is *nearly perfect* if its size is n. A vertex is *covered by* a matching M if it is the end vertex of some edge in M, and *exposed by* M if it is not covered by M. A cycle is M*-alternating* if edges in M and $E \setminus M$ appear alternatively along C, except for one vertex (when the cycle is odd).

Reconfiguration of Labeled Matching in Triangular Grid Graphs

Consider the 2-dimensional triangular lattice of infinite size. A *triangular grid* graph is a subgraph induced by a finite number of points in the triangular lattice. See Figure 2 for example. In this paper, we also assume that a triangular grid graph is always connected. A *hole* of a triangular grid graph is a face of the graph whose boundary is a cycle of length at least 6.

We here formally define our reconfiguration problem. Let $G = (V, E)$ be a triangular grid graph with $2n + 1$ vertices. We denote $V = [2n + 1]$, where, for a positive integer x, we write $[x] = \{1, 2, \dots, x\}$.

A *placement* is a mapping $p : [n] \to E$ such that $p(i)$ and $p(j)$ have no common end vertices for every distinct i, j. We call each $p(i)$ a *piece*. Then $\{p(i) \mid i \in [n]\} \subseteq E$ forms a nearly perfect matching in G, which is denoted by M_p. Let v_p be the unique vertex exposed by M_p. We also say that a mapping p *exposes* v_p.

We define the following operations on a placement, which we call slide (see Figure 3). Suppose that there exists an integer $j \in [n]$ such that $p(j) = (u, v)$ and $(v, v_p) \in E$. Then we transform p to a placement p_s defined as

$$p_s(i) = \begin{cases} p(i) & (i \neq j) \\ (v, v_p) & (i = j). \end{cases}$$

The obtained placement p_s exposes the vertex u. In this case, we write $p \rightsquigarrow p_s$.

Let p, q be two distinct placements. If there exists a sequence of placements $\mu_0, \mu_1, \dots, \mu_\ell$ such that (1) $\mu_0 = p$, $\mu_\ell = q$, (2) $\mu_t \rightsquigarrow \mu_{t+1}$ for every integer $t \in \{0, 1, \dots, \ell - 1\}$, then we say that p is *reconfigured* to q. A graph is *reconfigurable* if any two placements can be reconfigured to each other.

We remark that, in the Gourds puzzle, a piece has a pair of labels (numbers), meaning that each piece has an orientation. That is, a mapping is defined from $[n]$ to $\{(u, v), (v, u) \mid (u, v) \in E\}$. This requires us to define another operation to change the orientation of pieces. Specifically, when a piece with the exposed vertex induces a triangle, we are allowed to change the orientation of the piece. Our problem does not distinguish (u, v) and (v, u), and

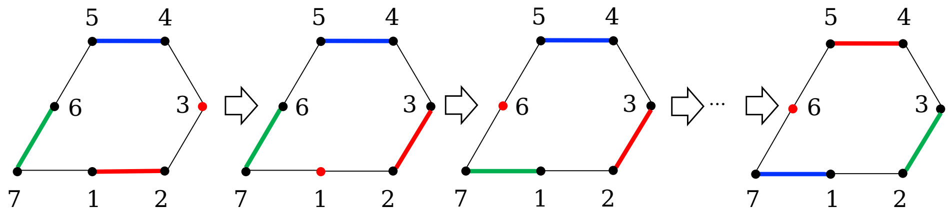

Figure 4 Rotation operations for a placement aligned with an odd cycle when $k = 3$.

a placement is defined on a mapping from $[n]$ to E. It should be noted, however, that our results can be adapted to the Gourds puzzle case with orientation. See Sections 3.1 and 4 for the details.

Rotation along a Cycle

We define a sequence of slide operations, called *rotation*, which will be used in the subsequent sections. Let C be an odd cycle. We say that a placement p is *aligned with* C if C is an odd M_p-alternating cycle and C has the exposed vertex v_p.

Let p be a placement aligned with C. In what follows, we assume for simplicity that $V(C) = [2k+1]$ for some integer $k \in [n]$, and that the vertices of C are aligned in the anti-clockwise order along C. We also assume that the first k pieces $p(1), p(2), \ldots, p(k)$ of p are located on the cycle C.

For an odd integer $j \in [2k+1]$, we define a placement p_j as, for $i \in [k]$,

$$p_j(i) = \begin{cases} (2i-1, 2i) & (2i < j) \\ (2i, 2i+1) & (j < 2i), \end{cases}$$

and $p_j(i) = p(i)$ for $i \geq k+1$. Thus p_j exposes the vertex j. Moreover, for two integers $h, j \in [2k+1]$ such that $h \equiv j \pmod 2$, define $p_{j,h}$ as, for $i \in [k]$,

$$p_{j,h}(i) = \begin{cases} (h+2i-2, h+2i-1) & (h+2i-1 < j) \\ (h+2i-1, h+2i) & (j < h+2i-1), \end{cases}$$

where these vertex labels are defined on $\mathbb{Z}_{2k+1}$ (i.e., modulo $2k+1$), and $p_{j,h}(i) = p(i)$ for $i \geq k+1$.

Figure 4 is an example when $k = 3$. The left-most figure depicts a placement $p_{3,1} = p_3$ where $(p_{3,1}(1), p_{3,1}(2), p_{3,1}(3)) = ((1,2),(4,5),(6,7))$. By applying slide to $p_{3,1}$ once, we obtain $p_{1,1} = p_1$, that is, $(p_{1,1}(1), p_{1,1}(2), p_{1,1}(3)) = ((2,3),(4,5),(6,7))$. The right-most figure depicts a placement $p_{6,4}$, which is $(p_{6,4}(1), p_{6,4}(2), p_{6,4}(3)) = ((4,5),(7,1),(2,3))$.

The following observation asserts that $p_{j,h}$'s can be reconfigured to each other in $O(k^2)$ slide operations along C. Such a sequence of slide operations is called *rotation along* C, or we say that we *rotate* p *along* C.

▶ **Observation 2.1.** *For any two odd integers $j, j' \in [2k+1]$, we can reconfigure p_j to $p_{j'}$ using at most k slide operations. Moreover, for any four integers $j, j' \in [2k+1]$ and $h, h' \in [2k+1]$ such that $j \equiv h$ and $j' \equiv h' \pmod 2$, we can reconfigure $p_{j,h}$ to $p_{j',h'}$ using at most $k^2 + k$ slide operations.*

Proof. Observe that, applying slide to p_j along C, the exposed vertex j is moved to $j-2$ or $j+2 \pmod{2k+1}$. This means that $p_j \rightsquigarrow p_{j-2}$ and $p_j \rightsquigarrow p_{j+2}$ hold. Hence, by repeating slide operations at most k times, p_j can be reconfigured to $p_{j'}$. We next show the second

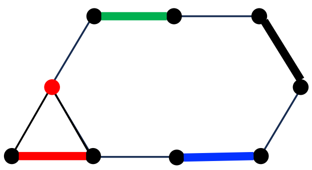

Figure 5 A factor-critical graph that is not reconfigurable. We cannot change the ordering of the pieces by slide.

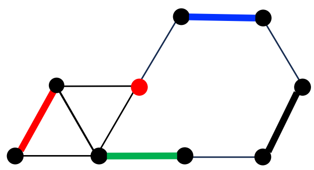

Figure 6 A factor-critical graph that is reconfigurable.

statement. Similarly to the first statement, we can reconfigure $p_{j,h}$ to $p_{h,h}$ in at most k slide operations. Applying slide to $p_{h,h}$ along C, we obtain a placement $p_{h-2,h+1}$. Repeating this procedure at most k times, we can reconfigure $p_{j,h}$ to $p_{h'-3,h'}$. Since we can reconfigure $p_{h'-3,h'}$ to $p_{j',h'}$ in at most k slide operations, the total number of slide operations is at most $k^2 + k$. ◀

3 Reconfiguration on Factor-Critical Graphs

In this section, we discuss the reconfigurability of a factor-critical graph. Recall that a graph is *factor-critical* if, for any vertex v, G has a nearly perfect matching that does not cover v.

As mentioned in Introduction, being a factor-critical graph is a necessary condition for reconfigurability.

▶ **Observation 3.1.** *If a triangular grid graph G is reconfigurable, then it is factor-critical.*

Proof. If G is not factor-critical, then G has some vertex u such that every nearly perfect matching covers u. Then we cannot move the piece covering u in an initial placement so that u becomes exposed. Hence there exist two placements such that one cannot be reconfigured to the other. Thus the observation holds. ◀

Moreover, as observed in Hamersma et al. [13], the 2-connectivity is necessary for a graph to be reconfigurable. We remark that, even if a graph is 2-connected and factor-critical, it may not be reconfigurable. See Figure 5.

The main theorem of this section is the following. We show that a graph is reconfigurable if it has a vertex of degree 6, which corresponds to a vertex not on the boundary cycles of the holes or the outer face.

▶ **Theorem 3.2.** *Let $G = (V, E)$ be a 2-connected factor-critical triangular grid graph. If G has a vertex of degree 6, then G is reconfigurable.*

We remark that our condition is not necessary, as there exists a 2-connected factor-critical graph such that it does not have a vertex of degree 6, but it is reconfigurable. See Figure 6 (see also Lemma 3.14 and Section 5).

3.1 Proof Overview

In this section, we present the proof overview of Theorem 3.2. The proof makes use of the ear decomposition of a factor-critical graph to design a reconfiguration sequence.

An *ear decomposition* of a graph G is a sequence of subgraphs $G_1, G_2, \dots, G_k = G$ starting from a subgraph G_1 such that G_{i+1} is obtained from G_i by adding an ear P_i for each $i \geq 1$, where an *ear* P of a subgraph G' is a path of G with end vertices in G' such that P is

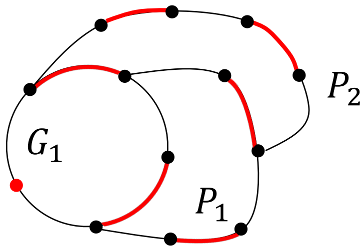

Figure 7 An ear decomposition and a matching aligned with the ear decomposition.

internally disjoint from G'. We denote by $G' + P$ the subgraph obtained from G' by adding the ear P. Thus, in the ear decomposition, it holds that $G_{i+1} = G_i + P_i$ for each $i \in [k-1]$. See Figure 7 for an example.

An ear decomposition is *proper* if the end vertices of each ear are distinct, and *odd* if each ear is of odd length. It is known that a 2-connected factor-critical graph is characterized by odd and proper ear decomposition.

▶ **Proposition 3.3** (Theorem 5.5.2 in Lovász–Plummer [21])**.** *A graph G is 2-connected and factor-critical if and only if G has an odd and proper ear decomposition starting from an odd cycle.*

Let p be a placement of G. Recall that M_p denotes a nearly perfect matching $\{p(i) \in E \mid i \in [n]\}$, and v_p is the vertex exposed by M_p. We say that a placement p is *aligned with an ear decomposition* $G_1, \ldots, G_k$ if it satisfies the following two conditions (Figure 7).

(a) G_1 is an odd M_p-alternating cycle with the exposed vertex v_p.

(b) For each $i \in [k-1]$, the ear P_i is M_p-alternating and its end vertices are not covered by $M_p \cap E(P_i)$.

We show that any placement p can be reconfigured so that the obtained placement p' is aligned with a given ear decomposition $G_1, \ldots, G_k$. See Section 3.2 for the proof.

▶ **Lemma 3.4.** *Let G be a 2-connected factor-critical triangular grid graph with $2n+1$ vertices. Let $G_1, \ldots, G_k$ be an odd and proper ear decomposition of G. Then we can reconfigure any placement p to a placement aligned with $G_1, \ldots, G_k$ in $O(kn)$ slide operations.*

Let q be a target placement of G. Applying Lemma 3.4 to q as well, we can reconfigure q so that the obtained placement q' is aligned with $G_1, \ldots, G_k$. By taking the inverse of the reconfiguration steps, we see that q' can be reconfigured to q in $O(kn)$ slide operations. Therefore, in order to reconfigure p to q, it suffices to reconfigure p' to q'.

We now present how to find a reconfiguration sequence between two placements aligned with $G_1, \ldots, G_k$. Since G_i is factor-critical for any $i \in [k]$, the ear structure suggests to design a reconfiguration sequence recursively. In fact, we will show in Lemma 3.9 (Section 3.3) that, if G_j is a reconfigurable graph with at least 5 vertices for some $j < k$, then so is $G_k = G$. Note that the lemma holds even if a graph is not a triangular grid graph. However, G_j's may not necessarily be reconfigurable, as there exists a factor-critical graph which is not reconfigurable. To overcome the difficulty, we introduce a special kind of ear decomposition starting from simple reconfigurable subgraphs.

We say that an odd and proper ear decomposition $G_1, G_2, \ldots, G_k$ is *admissible* if it satisfies either

(i) G_1 is a cycle of length 5 (Figure 8), or

(ii) P_1 is of length 3 and has the end vertices u, v which are adjacent in G_1 (Figure 9).

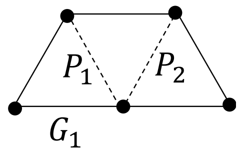

Figure 8 A pentagon.

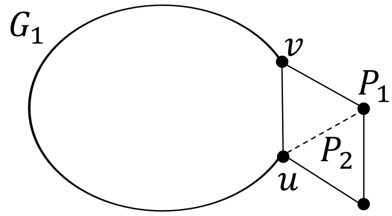

Figure 9 An odd cycle with a diamond.

Consider the case when (i) is satisfied. Since the ordering of ears with length 1 may be changed in the ear decomposition, we may assume that the first 2 ears P_1 and P_2 are the inner edges of G_1. Then $G_3 = G_1 + P_1 + P_2$ induces a pentagon in G, where a *pentagon* is a subgraph induced by three adjacent triangles. It is not difficult to see that the pentagon is reconfigurable in a constant number of slide operations. On the other hand, consider the case when (ii) is satisfied. Similarly to the case (i), we may assume that the second ear P_2 is the inner edge of P_1. The subgraph $G_3 = G_1 + P_1 + P_2$ induces an odd cycle attached to a diamond, where a *diamond* is a subgraph induced by two adjacent triangles. The subgraph is shown to be reconfigurable in Lemma 3.14.

Therefore, G_3 in an admissible ear decomposition is reconfigurable. Hence, if there exists an admissible ear decomposition, we can find a reconfiguration sequence as below. See Section 3.3 for the details.

▶ **Lemma 3.5.** *Let $G_1, \ldots, G_k = G$ be an admissible ear decomposition of G. Then any two placements aligned with the ear decomposition can be reconfigured to each other.*

We remark that the length of a reconfiguration sequence obtained in the above lemma is at most n^{2n} for a graph with $2n + 1$ vertices, which is not bounded by a polynomial in n. It may be interesting to find the optimal bound on the length of reconfiguration sequences. It is known in [13] that the length is $\Omega(n^2)$.

Finally, we show that there always exists an admissible ear decomposition in a triangular grid graph with a vertex of degree 6.

▶ **Theorem 3.6** (⋆). *Let G be a 2-connected factor-critical triangular grid graph such that it has a vertex of degree 6. Then G has an admissible ear decomposition.*

Theorem 3.6 is a graph-theoretical result independent of designing a reconfiguration sequence. Theorem 3.6 can be proved by investigating the matching structure of factor-critical triangular grid graphs.

In summary, a reconfiguration sequence from an initial placement p to a target placement q can be realized as below, which completes the proof of Theorem 3.2.

1. Reconfigure p to a placement aligned with an admissible ear decomposition $G_1, \ldots, G_k$, denoted by p', by Lemma 3.4.
2. Using Lemma 3.5, reconfigure p' to another placement q' aligned with $G_1, \ldots, G_k$, where q' is a placement obtained from q by Lemma 3.4.
3. Reconfigure q' to the target placement q.

We remark that the proof of Theorem 3.2 above can be adapted to the Gourds puzzle in which a piece has an orientation. This is because the structures (i) and (ii) in an admissible ear decomposition can also be used to change the orientation of pieces in an arbitrary way. Thus we have the following corollary.

▶ **Corollary 3.7.** *Let B be a hexagonal grid such that the dual triangular grid graph is a 2-connected factor-critical graph with a vertex of degree 6. Then any two configurations of the same set of pieces on B can be reconfigured to each other.*

3.2 Reconfiguration to a Placement Aligned with Ear Decomposition

In this subsection, we will show Lemma 3.4, that is, we will show that we can reconfigure an initial placement p to a placement aligned with a given odd and proper ear decomposition $G_1, \dots, G_k$.

We first prove that we can reconfigure so that any vertex is exposed.

▶ **Lemma 3.8.** *Let G be a 2-connected factor-critical triangular grid graph with $2n+1$ vertices. For any vertex v, we can reconfigure a placement p so that v is the exposed vertex, in $O(n)$ slide operations.*

Proof. Since G is factor-critical, G has a nearly perfect matching M_v that exposes v. The symmetric difference $M_p \Delta M_v$ contains an M_p-alternating path P from v_p to v, which is of even length. We reconfigure p by sliding the pieces on the path P one-by-one. The resulting placement exposes v. The number of slide operations is $|P|/2$, which is $O(n)$. ◀

To obtain a placement aligned with the ear decomposition, we first reconfigure so that an end vertex of the last ear P_{k-1} is exposed using Lemma 3.8. Then, since each inner vertex of the last ear P_{k-1} is of degree 2 in G, the obtained placement is aligned with P_{k-1}. By applying this procedure repeatedly for each ear, we can obtain a placement aligned with $G_1, \dots, G_k$. This implies Lemma 3.4 as below.

Proof of Lemma 3.4. Let v_i be an end vertex of ear P_i for $i \in [k-1]$. The basic observation is that, each inner vertex of the last ear P_{k-1} is of degree 2, and hence, if a nearly perfect matching M exposes v_{k-1}, the last ear P_{k-1} is an M-alternating path such that the end vertices are not covered by edges of $M \cap E(P_{k-1})$.

We perform the following procedure for each $i = k-1, k-2, \dots, 1$. Note that G_i is factor-critical for any $i \in [k]$.

1. Applying Lemma 3.8 to G_{i+1}, we reconfigure the current placement of G_{i+1} so that v_i is exposed. Then the resulting placement is aligned with P_i by the above observation.

In the end of the above procedure, the obtained placement of the original graph G is aligned with $P_{k-1}, \dots, P_1$. Moreover, the exposed vertex is on G_1. Thus this is a desired placement. The necessary number of slide operations is $O(kn)$, since we repeat the procedure of Lemma 3.8 $k-1$ times. ◀

3.3 Reconfiguration using Ear Decomposition

We next present how to reconfigure a placement aligned with an ear decomposition. Using the ear structure, we can find a reconfiguration sequence if the subgraph G_{k-1} is reconfigurable.

▶ **Lemma 3.9.** *Let $G_1, \dots, G_k$ be an odd and proper ear decomposition of a graph G with $2n+1$ vertices. Suppose that G_{k-1} has at least 5 vertices, and that, in G_{k-1}, any placement aligned with the ear decomposition $G_1, \dots, G_{k-1}$ can be reconfigured to another placement aligned with $G_1, \dots, G_{k-1}$, using t slide operations. Then there exists a reconfiguration sequence between any two placements along with $G_1, \dots, G_k$ in G, which requires $O(n^2(t+n))$ slide operations.*

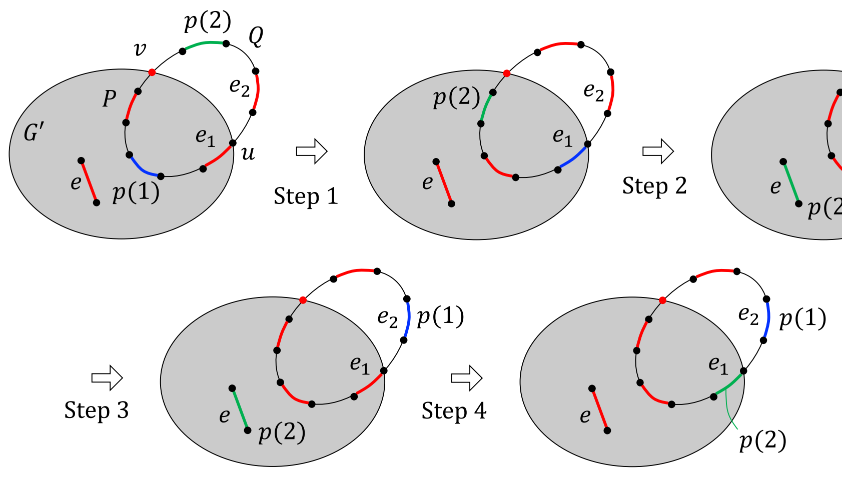

Figure 10 Proof of Claim 3.11: An example when $j = 2$.

Proof. For simplicity, we denote $G_{k-1} = G'$ and $P_{k-1} = Q$ in the proof. Let u, v be the end vertices of Q. Let p be an initial placement of G and q be a target placement of G, both of which are aligned with the ear decomposition. It follows from Lemma 3.8 that we can reconfigure p (and q, resp.) so that v is exposed. Hence we may assume that both p and q expose v. Moreover, by changing the indices of the pieces if necessary, we may assume that the pieces $q(1), \dots, g(\ell)$ are placed on the ear Q in the order from v to u.

▷ Claim 3.10. There exists an M_p-alternating path P of even length from v to u in G'.

Proof. Since G' is factor-critical, there exists a nearly perfect matching M_u that exposes u. Taking $M_p \triangle M_u$, we see that there exists an even M_p-alternating path P from v to u in G'. ◁

Let C be the cycle consisting of P and Q. Then C is an odd M-alternating cycle in G. We will show that we can reconfigure p so that the first ℓ pieces are on the ear Q, using the cycle C. We consider the following two cases, depending on whether C is a Hamilton cycle or not.

▷ Claim 3.11. Suppose that C is not a Hamilton cycle of G. Then we can reconfigure p to a placement p' so that $p'(1), \dots, p'(\ell)$ are placed on Q in this order from v to u, using $O(\ell(t + n^2))$ slide operations.

Proof. In this case, the graph G' has an edge $e \notin E(P)$ such that $e \in M_p$. Let e_1 be the edge of M_p covering u, and e_2 be the edge of of M_p covering the vertex adjacent to u on Q. See Figure 10.

We may assume that $p(1)$ is on the cycle C, as otherwise $p(1)$ is contained in G', and hence we can reconfigure the current placement in G' so that $p(1)$ is on P, keeping v exposed, using t slide operations by the assumption.

We reconfigure p by the following 4 steps for $j = 2, \dots, \ell$. Initially, we set $\tilde{p} = p$.

1. We move the piece $\tilde{p}(j)$ so that $\tilde{p}(j)$ is on P as follows.
   a. If $\tilde{p}(j)$ is on Q, we rotate the current placement $\tilde{p}$ along C so that $\tilde{p}(j)$ is located on P.
   b. If $\tilde{p}(j)$ is contained in G' but not on P, then we reconfigure the current placement $\tilde{p}$ on G' so that $\tilde{p}(j)$ is located on P, keeping that $\tilde{p}(1), \dots, \tilde{p}(j-1)$ are on C.

2. We reconfigure the current placement $\tilde{p}$ on G' to swap $\tilde{p}(j)$ and the piece on e. Thus $\tilde{p}(j) = e$.
3. We rotate the current placement $\tilde{p}$ along C so that $\tilde{p}(j-1)$ is e_2.
4. We reconfigure the current placement $\tilde{p}$ on G' to swap $\tilde{p}(j)$ and the piece on e_1. Thus $\tilde{p}(j)$ has been changed to e_1.

In the end of the j-th iteration, $\tilde{p}(1), \ldots, \tilde{p}(j)$ are located on C in this order from v to u. Therefore, in the end of the above procedure, the pieces $\tilde{p}(1), \ldots, \tilde{p}(\ell)$ are located on C in this order from v to u. Thus we can rotate $\tilde{p}$ along C so that they are on Q.

In the above procedure, for each j, we reconfigure the placement restricted on G' in a constant number of times, and we rotate the placement along C at most twice. Therefore, the total number of slide operations is $O(\ell(t + n^2))$ by Observation 2.1. ◁

▷ Claim 3.12. Suppose that C is a Hamilton cycle of G. Then we can reconfigure p to a placement p' so that $p'(1), \ldots, p'(\ell)$ are placed on Q in this order from v to u, using $O(\ell n(t + n))$ slide operations.

Proof. Since G' has at least 5 vertices, P has at least 2 edges of M_p. Let e_1, e_2 be two edges in $M_p \cap E(P)$ such that e_1, e_2 appear consecutively along P. We can swap the 2 pieces on e_1 and e_2 by reconfiguring on G', using t slide operations. By using the strategy similar to the bubble sort algorithm, we can obtain a placement p' such that $p'(1), \ldots, p'(\ell)$ are on C in this order from v to u. This requires $O(\ell n)$ swaps. Since each swap takes $O(t + n)$ slide operations, it takes $O(\ell n(t + n))$ slide operations in total. ◁

In each case, we can reconfigure p to a placement p' so that the pieces $p'(1), \ldots, p'(\ell)$ are located on Q in this order from v to u. Since we can reconfigure the placement on G' to any placement, we can reconfigure p' to q. The total number of slide operations is $O(\ell n(t + n)) = O(n^2(t + n))$. ◀

By applying Lemma 3.9 recursively, we see that, if G_j is a reconfigurable subgraph with at least 5 vertices for some $j < k$, then $G_k = G$ is reconfigurable. In particular, if a given ear decomposition is admissible, then G is shown to be reconfigurable.

Below we upper-bound the number of operations to reconfigure two placements aligned with an admissible ear decomposition. We will show each case of the definition of an admissible ear decomposition separately. We assume that a graph has $2n + 1$ vertices for $n \geq 2$.

▶ **Lemma 3.13** (⋆). *Let $G_1, \ldots, G_k$ be an admissible ear decomposition such that G_1 is a cycle of length 5 (Figure 8). Then we can reconfigure an arbitrary placement p aligned with $G_1, \ldots, G_k$ to another placement aligned with $G_1, \ldots, G_k$ in at most n^{2n} slide operations.*

We next discuss the second case of an admissible ear decomposition. The following lemma says that the base case is reconfigurable.

Let $\tilde{G} = (V, E)$ be a triangular grid graph with $2n + 1$ vertices for $n \geq 3$ as in Figure 11. More specifically, $V = [2n + 1]$, and it consists of an odd cycle C of length $2n - 1$ with vertex set $[2n - 1]$, attached to a diamond D with vertex set $\{2n - 2, 2n - 1, 2n, 2n + 1\}$.

▶ **Lemma 3.14.** *The graph $\tilde{G}$ defined above with $2n + 1$ vertices ($n \geq 3$) is reconfigurable in at most $n^3 + n^2$ operations.*

Proof. Let p and q be an initial and target placements of $\tilde{G}$, respectively. We may assume that the target pieces $q(1), \ldots, q(n - 1)$ are located in the anti-clockwise order along C, and that D has pieces $q(n - 1)$ and $q(n)$. Let C' be the Hamilton cycle of length $2n + 1$ in $\tilde{G}$.

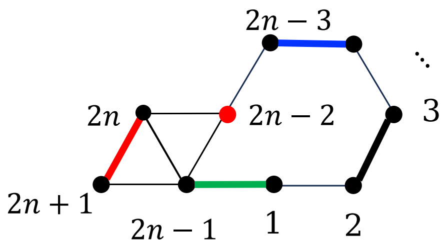

Figure 11 A factor-critical graph that is reconfigurable.

We present a reconfiguration sequence as follows: Initially, we set $\tilde{p} = p$. For $j = 1, 2, \ldots, n$, we do the following 2 steps.

1. We rotate the current placement $\tilde{p}$ along C' so that $\tilde{p}(j)$ is equal to the edge $(2n, 2n+1)$.
2. We rotate the current placement $\tilde{p}$ along C so that $\tilde{p}(j-1)$ is equal to the edge $(2n-1, 1)$. Then $\tilde{p}(1), \ldots, \tilde{p}(j)$ are located on C' in the anti-clockwise order.

In the end of the procedure, $\tilde{p}(1), \ldots, \tilde{p}(n)$ are located on C' in the anti-clockwise order, which is the desired placement q. In each iteration, we rotate the current placement along C or C'. By choosing the shorter one between the clockwise rotation and the anti-clockwise rotation, it requires at most $n^2 + n$ slide operations by Observation 2.1. Hence the total number of slide operations is at most $n^3 + n^2$. ◀

We next show the case when an admissible ear decomposition satisfies the second case. This, together with Lemma 3.13, proves Lemma 3.5.

▶ **Lemma 3.15** (⋆). *Let $G_1, \ldots, G_k$ be an admissible ear decomposition such that P_1 is of length 3 and has the end vertices u, v which are adjacent in G_1 (Figures 9 and 11). Then we can reconfigure an arbitrary placement p aligned with $G_1, \ldots, G_k$ to another placement in at most n^{2n} slide operations.*

4 Reconfiguration on Locally-Connected Graphs

In this section, we consider a triangular grid graph which is locally-connected. A vertex u in a graph G is said to be *locally-connected* if the subgraph $G[N(u)]$ is connected. A graph G is called *locally-connected* if every vertex is locally-connected. It is observed that, if G is locally-connected, then it is 2-connected, since, for a cut vertex u, the subgraph $G[N(u)]$ is disconnected.

The following theorem says that a locally-connected triangular grid graph, except for the Star of David graph (Figure 1), is Hamiltonian. We note that, since their proof is constructive, a Hamilton cycle can be found in polynomial time.

▶ **Theorem 4.1** (Gordon, Orlovich, and Werner [12]). *Let G be a triangular grid graph. If G is locally-connected, but not isomorphic to the Star of David graph, then it has a Hamilton cycle.*

It follows from the above theorem that a locally-connected triangular grid graph, except for the Star of David graph, is factor-critical, as it has an odd and proper ear decomposition starting from a Hamilton cycle such that all the ears are single edges. On the other hand, the Star of David graph is not factor-critical, and hence it is not reconfigurable (see also [13]).

The main theorem of this section is the following.

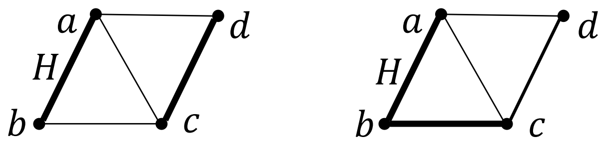

Figure 12 Diamonds and Hamilton cycles satisfying the condition in Lemma 4.5.

▶ **Theorem 4.2.** *Let $G = (V, E)$ be a triangular grid graph with $2n + 1$ vertices. If G is locally-connected, but not isomorphic to the Star of David graph, then G is reconfigurable. Moreover, a reconfiguration sequence using $O(n^3)$ slide operations can be found in polynomial time.*

The proof for Theorem 4.2 exploits a Hamilton cycle in G to design a reconfiguration sequence. We note that the proof for 2-connected graphs with no holes by Hamersma et al. [13] also uses a Hamilton cycle. Our proof refines their proof so that we can deal with holes.

Suppose that we are given two placements p and q. The proposed algorithm to reconfigure p to q consists of the following three phases.

1. Reconfigure p to a placement aligned with a Hamilton cycle H, denoted by p'.
2. Reconfigure p' to another placement q' aligned with H.
3. Reconfigure q' to the target placement q.

In Phase 1, we first reconfigure the initial placement p to a placement aligned with the Hamilton cycle H, which is denoted by p'. Applying the same procedure to the target placement q, we obtain a placement aligned with H, denoted by q'. We then reconfigure p' to q' in Phase 2. In Phase 3, the placement q' can be reconfigured to the target placement q by taking the inverse of Phase 1 operations for q.

It was shown in Hamersma et al. [13] that we can reconfigure a placement to a placement aligned with a Hamilton cycle. We recall that a locally-connected graph is 2-connected.

▶ **Theorem 4.3** (Hamersma et al. [13])**.** *Let G be a 2-connected triangular grid graph with $2n + 1$ vertices. Then we can reconfigure any placement to a placement aligned with a Hamilton cycle H, using $O(n^2)$ slide operations.*

Therefore, Phases 1 and 3 can be implemented in $O(n^2)$ slide operations. Thus it suffices to implement Phase 2 to reconfigure any placement aligned with H to another placement aligned with H. This step is realized by the following theorem.

▶ **Theorem 4.4.** *Let G be a triangular grid graph with $2n + 1$ vertices, which is locally-connected, but not isomorphic to the Star of David graph. Let H be a Hamilton cycle. For a pair of placements p, q aligned with H, we can reconfigure p to q in $O(n^3)$ slide operations.*

The proof of Theorem 4.4 adopts a similar strategy to that of Theorem 3.2, where we employs a Hamilton cycle instead of an ear decomposition. We identify a small subgraph that can be used to reconfigure placements aligned with H.

▶ **Lemma 4.5** ($\star$)**.** *Let G be a triangular grid graph with $2n + 1$ vertices. Let H be a Hamilton cycle of G. Suppose that G has a diamond, whose vertices are a, b, c, d aligned in the anti-clockwise order (Figure 12), such that either*

(i) *H contains the edges (a, b) and (c, d), but does not contain (a, c), or*
(ii) *H contains the edges (a, b) and (b, c).*

Then we can reconfigure any placement aligned with H to another placement aligned with H in $O(n^3)$ slide operations.

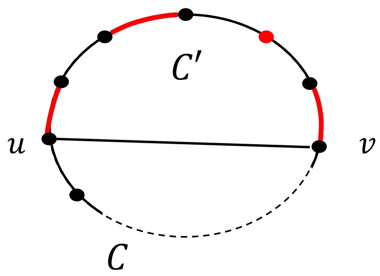

Figure 13 An odd cycle with one chord.

We then show that such a diamond with a Hamilton cycle always exists if G is a locally-connected triangular grid graph, which is not isomorphic to the Star of David graph. This shows Theorem 4.4.

▶ **Lemma 4.6** (⋆). *Let G be a triangular grid graph with $2n+1$ vertices. If G is a locally-connected triangular grid graph, which is not isomorphic to the Star of David graph, then there exist a Hamilton cycle H and a diamond, whose vertices are denoted by a, b, c, d (Figure 12), that satisfy either* (i) *or* (ii) *in Lemma* 4.5*.*

This section is concluded with stating our results on the Gourds puzzle.

▶ **Corollary 4.7.** *Let B be a hexagonal grid such that the dual graph is locally-connected, but not isomorphic to the Star of David graph. Then any two configurations of the same set of n pieces on B can be reconfigured to each other, using $O(n^3)$ moves.*

5 Concluding Remarks

In this paper, we introduced a new reconfiguration problem of labeled matchings in a triangular grid graph. We provided sufficient conditions for a graph to be reconfigurable using a factor-critical graphs and a locally-connected graphs. It remains open to characterize a reconfigurable triangular grid graph, when it is a factor-critical graph with no vertex of degree 6, but not locally-connected. Let us here discuss the difficulty to obtain the characterization. For example, consider a graph G consisting of an odd cycle C of length $2n+1$ with one chord (u, v) (Figure 13). Let C' be the odd cycle of G with the edge (u, v). The length of C' is denoted by $2m+1$. Then we can observe that the reconfigurability of G depends on m and n. Specifically, G is reconfigurable if and only if $n-1$ and $m-1$ are mutually prime. Indeed, since we can only rotate a placement along C' and C, which correspond to cyclic permutations on $[m]$ and $[n]$, respectively, any permutation can be realized if and only if $n-1$ and $m-1$ are mutually prime. This observation would imply that it requires algebraic conditions to characterize a reconfigurable graph, like the 15-puzzle.

References

1 Oswin Aichholzer, Erik D. Demaine, Matias Korman, Anna Lubiw, Jayson Lynch, Zuzana Masárová, Mikhail Rudoy, Virginia Vassilevska Williams, and Nicole Wein. Hardness of token swapping on trees. In Shiri Chechik, Gonzalo Navarro, Eva Rotenberg, and Grzegorz Herman, editors, *30th Annual European Symposium on Algorithms, ESA 2022, September 5-9, 2022, Berlin/Potsdam, Germany*, volume 244 of *LIPIcs*, pages 3:1–3:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.ESA.2022.3.

2 Valentin Bartier, Nicolas Bousquet, and Amer E. Mouawad. Galactic token sliding. *J. Comput. Syst. Sci.*, 136:220–248, 2023. doi:10.1016/J.JCSS.2023.03.008.

3 Marthe Bonamy, Nicolas Bousquet, Marc Heinrich, Takehiro Ito, Yusuke Kobayashi, Arnaud Mary, Moritz Mühlenthaler, and Kunihiro Wasa. The perfect matching reconfiguration problem. In Peter Rossmanith, Pinar Heggernes, and Joost-Pieter Katoen, editors, *44th International Symposium on Mathematical Foundations of Computer Science, MFCS 2019, August 26-30, 2019, Aachen, Germany*, volume 138 of *LIPIcs*, pages 80:1–80:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPICS.MFCS.2019.80.

4 Édouard Bonnet, Tillmann Miltzow, and Pawel Rzazewski. Complexity of token swapping and its variants. *Algorithmica*, 80(9):2656–2682, 2018. doi:10.1007/S00453-017-0387-0.

5 Nicolas Bousquet, Amer E. Mouawad, Naomi Nishimura, and Sebastian Siebertz. A survey on the parameterized complexity of the independent set and (connected) dominating set reconfiguration problems. *CoRR*, abs/2204.10526, 2022. doi:10.48550/arXiv.2204.10526.

6 Josh Brunner, Lily Chung, Erik D. Demaine, Dylan H. Hendrickson, Adam Hesterberg, Adam Suhl, and Avi Zeff. 1 X 1 rush hour with fixed blocks is pspace-complete. In Martin Farach-Colton, Giuseppe Prencipe, and Ryuhei Uehara, editors, *10th International Conference on Fun with Algorithms, FUN 2021, May 30 to June 1, 2021, Favignana Island, Sicily, Italy*, volume 157 of *LIPIcs*, pages 7:1–7:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.FUN.2021.7.

7 Kevin Buchin and Maike Buchin. Rolling block mazes are pspace-complete. *J. Inf. Process.*, 20(3):719–722, 2012. doi:10.2197/IPSJJIP.20.719.

8 Jean Cardinal and Raphael Steiner. Inapproximability of shortest paths on perfect matching polytopes. In Alberto Del Pia and Volker Kaibel, editors, *Integer Programming and Combinatorial Optimization - 24th International Conference, IPCO 2023, Madison, WI, USA, June 21-23, 2023, Proceedings*, volume 13904 of *Lecture Notes in Computer Science*, pages 72–86. Springer, 2023. doi:10.1007/978-3-031-32726-1_6.

9 Erik D. Demaine, Martin L. Demaine, Eli Fox-Epstein, Duc A. Hoang, Takehiro Ito, Hirotaka Ono, Yota Otachi, Ryuhei Uehara, and Takeshi Yamada. Linear-time algorithm for sliding tokens on trees. *Theor. Comput. Sci.*, 600:132–142, 2015. doi:10.1016/J.TCS.2015.07.037.

10 Erik D. Demaine and Mikhail Rudoy. A simple proof that the $(n^2 - 1)$-puzzle is hard. *Theor. Comput. Sci.*, 732:80–84, 2018. doi:10.1016/J.TCS.2018.04.031.

11 Gary William Flake and Eric B. Baum. Rush hour is pspace-complete, or "why you should generously tip parking lot attendants". *Theor. Comput. Sci.*, 270(1-2):895–911, 2002. doi:10.1016/S0304-3975(01)00173-6.

12 Valery S. Gordon, Yury L. Orlovich, and Frank Werner. Hamiltonian properties of triangular grid graphs. *Discret. Math.*, 308(24):6166–6188, 2008. doi:10.1016/J.DISC.2007.11.040.

13 Joep Hamersma, Marc J. van Kreveld, Yushi Uno, and Tom C. van der Zanden. Gourds: A sliding-block puzzle with turning. In Yixin Cao, Siu-Wing Cheng, and Minming Li, editors, *31st International Symposium on Algorithms and Computation, ISAAC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, volume 181 of *LIPIcs*, pages 33:1–33:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.ISAAC.2020.33.

14 Robert A. Hearn and Erik D. Demaine. Pspace-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation. *Theor. Comput. Sci.*, 343(1-2):72–96, 2005. doi:10.1016/J.TCS.2005.05.008.

15 Robert A. Hearn and Erik D. Demaine. *Games, puzzles and computation.* A K Peters, 2009.

16 Takehiro Ito, Erik D. Demaine, Nicholas J. A. Harvey, Christos H. Papadimitriou, Martha Sideri, Ryuhei Uehara, and Yushi Uno. On the complexity of reconfiguration problems. *Theor. Comput. Sci.*, 412(12-14):1054–1065, 2011. doi:10.1016/J.TCS.2010.12.005.

17 Takehiro Ito, Naonori Kakimura, Naoyuki Kamiyama, Yusuke Kobayashi, and Yoshio Okamoto. Shortest reconfiguration of perfect matchings via alternating cycles. *SIAM J. Discret. Math.*, 36(2):1102–1123, 2022. doi:10.1137/20M1364370.

18 Wm. Woolsey Johnson and William E. Story. Notes on the "15" puzzle. *American Journal of Mathematics*, 2(4):397–404, 1879. URL: http://www.jstor.org/stable/2369492.

19 Naonori Kakimura and Yuta Mishima. Reconfiguration of labeled matchings in triangular grid graphs, 2024. arXiv:2409.11723.

20 Dohan Kim. Sorting on graphs by adjacent swaps using permutation groups. *Comput. Sci. Rev.*, 22:89–105, 2016. doi:10.1016/J.COSREV.2016.09.003.

21 L. Lovász and M.D. Plummer. *Matching Theory*. AMS Chelsea Publishing Series. AMS Chelsea Pub., 2009.

22 Tillmann Miltzow, Lothar Narins, Yoshio Okamoto, Günter Rote, Antonis Thomas, and Takeaki Uno. Approximation and hardness of token swapping. In Piotr Sankowski and Christos D. Zaroliagis, editors, *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, volume 57 of *LIPIcs*, pages 66:1–66:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPICS.ESA.2016.66.

23 Naomi Nishimura. Introduction to reconfiguration. *Algorithms*, 11(4):52, 2018. doi:10.3390/a11040052.

24 Daniel Ratner and Manfred K. Warmuth. Finding a shortest solution for the N × N extension of the 15-puzzle is intractable. In Tom Kehler, editor, *Proceedings of the 5th National Conference on Artificial Intelligence. Philadelphia, PA, USA, August 11-15, 1986. Volume 1: Science*, pages 168–172. Morgan Kaufmann, 1986. URL: http://www.aaai.org/Library/AAAI/1986/aaai86-027.php.

25 Laura Sanità. The diameter of the fractional matching polytope and its hardness implications. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 910–921, 2018. doi:10.1109/FOCS.2018.00090.

26 Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, Berlin, 2003.

27 J. Slocum and D. Sonneveld. *The 15 Puzzle Book: How It Drove the World Crazy*. Slocum Puzzle Foundation, 2006.

28 Jan van den Heuvel. The complexity of change. In Simon R. Blackburn, Stefanie Gerke, and Mark Wildon, editors, *Surveys in Combinatorics 2013*, volume 409 of *London Mathematical Society Lecture Note Series*, pages 127–160. Cambridge University Press, 2013. doi:10.1017/CBO9781139506748.005.