On Rolling Cube Puzzles

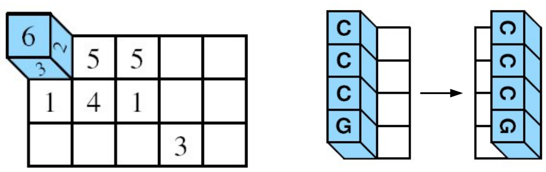
Kevin Buchin* Maike Buchin* Erik D. Demaine† Martin L. Demaine† Dania El-Khechen‡
Sándor P. Fekete§ Christian Knauer* André Schulz* Perouz Taslakian¶

Abstract

We analyze the computational complexity of various rolling cube puzzles.

1 Introduction

Consider the simple rolling cube puzzle in Fig. 1(a). The objective is to roll the die over all labeled cells of the board such that the label on the top face of the die is always the same as the label of the cell it lies on. The die may be rolled between neighboring cells by tipping it over along one edge that touches the board. The die may not be rotated within the same cell.



(a) by Joseph O'Rourke (b) by Martin Demaine

Figure 1: Rolling cube puzzles posed at CCCG 2005 [3].

Rolling cube puzzles were popularized by Martin Gardner in his *Mathematical Games* columns published in *Scientific American* [4, 5, 6]. He presented problems by Roland Sprague [9] and John Harris [7]. More recently, Robert Abbott has posed rolling cube puzzles in his books [1, 2].[1] In the open-problem session at CCCG 2005 [3], Joseph O'Rourke posed the computational complexity of rolling cube puzzles like the one in Fig. 1(a). During the discussion, Martin Demaine developed the multiple-dice puzzle in Fig. 1(b). Rolling a tetrahedron has been studied by Charles W. Trigg [10].

In general, a *rolling cube puzzle* consists of one or more dice, a board, a task, and a set of rules. A *die* is a cube with (some) labeled faces. We consider the case of a *standard die*[2], that is, a die with faces labeled 1 to 6 and with the labels on opposite sides adding up to 7. There are two standard dice. They can be distinguished by how the numbers 1, 2, and 3 are oriented with respect to each other and are called either *right-handed* or *left-handed* dice (Fig. 2). Here we use a right-handed orientation. The *board* is a grid with (some) labeled cells. Given a board, the objective is to roll the die over the cells of the board to accomplish some task, e.g., to visit all the labeled cells; sometimes we are given a starting position of the die and an ending position. We consider the case where all labeled cells must be visited.

In Section 2 we show that puzzles are easy (for a computer, not for a human) if labeled cells may be visited several times. Thus we later concentrate on puzzles where labeled cells must be visited exactly once. Cells can be of three types: *labeled*, *blocked*, or *free*. Labeled cells must be visited exactly once with the label appearing on the top face of the die being the same as the label of the cell. Blocked cells cannot be visited by the die. Free cells can be visited with any label on the top face of the die and any number of times. We restrict ourselves to puzzles with one die, but puzzles can also involve several dice as in Fig. 1(b). In Section 3 we prove that it is NP-complete to decide whether we can roll a die over the labeled cells of a board that has some free cells. This solves the open problem posed by Joseph O'Rourke. Free cells seem to be essential for the hardness of the problem; thus in Section 4 we present an algorithmic approach to puzzles with no free cells, and show that the solution to a puzzle with labeled (and possibly blocked) cells is not necessarily unique. The computational complexity of such puzzles remains open. Due to space constraints we omit the proofs of various results from this paper and refer the reader to the full version of the paper (in the electronic proceedings).

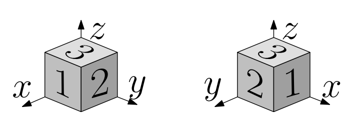


Figure 2: Right- and left-handed orientation.

*Institute of Computer Science, Free University Berlin, {buchin,mbuchin,knauer,schulza}@inf.fu-berlin.de
†Computer Science and AI Laboratory, Massachusetts Institute of Technology, {edemaine,mdemaine}@mit.edu
‡Department of Computer Science and Software Engineering, Concordia University, d_elkhec@cs.concordia.ca
§Department of Computer Science, Braunschweig University of Technology, s.fekete@tu-bs.de
¶School of Computer Science, McGill University, perouz@cs.mcgill.ca

[1] See also his website, http://www.logicmazes.com.

[2] We also consider puzzles with a two-colored die where the task is to color as many cells as possible in one color; see the full paper.

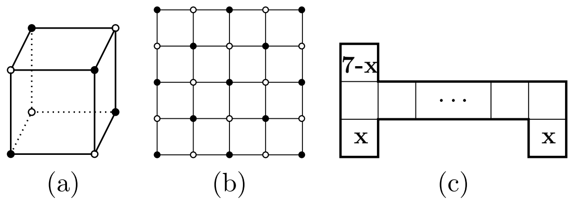


Figure 3: (a–b): Die and board with 2-colored corners. (c) U-turn and Z-turn.

2 Basic Properties

State Graph. The *state graph* has a vertex for each possible state of the die and an edge for each possible transition between two states. A *state* consists of a board position and the entire orientation of the die. In particular, the state encodes which label is on the top face, but even fixing this top label, there are four possible orientations, defined by the label facing a fixed side of the board. An edge of the state graph connects vertices corresponding to adjacent cells on the board for which it is possible to roll a die from one cell to the other, respecting the orientations of the two states. (Moves are reversible, so the graph is undirected.)

Parity Property. An important property of the state graph is that its vertex set naturally falls into two parts of equal size with no edges between the two parts. For a labeled cell, only two of the four orientations need to be considered if we restrict the die to one part of the state graph. This corresponds to the *parity property* of rolling cube puzzles [4].

In the following we look at this property in a different way. We consider how a 2-coloring of the corners of the cube induces a 2-coloring of the corners of the board. We color the corners of the cube alternatingly black and white as in Fig. 3(a). Rolling this cube over the board generates one of two 2-colorings of the corners of the board, i.e., a checkerboard pattern. One possible 2-coloring is shown in Fig. 3(b); the other is its complement with black and white exchanged. Consider a cube with a given label on top. Each of the two possible 2-colorings of the board is generated by two of the four orientations of the cube. It is not possible to return to a cell with the cube rotated by 90°.

Changing Direction Twice. If the die changes direction twice, the label on its top face is either the same as before or the opposite, depending on whether the last turn was back or not. More precisely, suppose the die starts with label x and rolls one cell in one direction, then an arbitrary number of cells in an orthogonal direction, and then one cell in the original or opposite direction. Then the die shows the label x if the turn was a *U-turn*, and $7-x$ if it was a *Z-turn*; see Fig. 3(c).

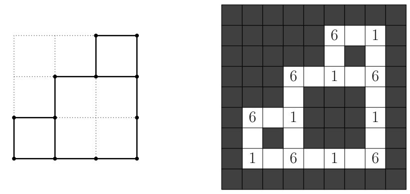


Figure 4: Reduction of a grid graph to a labeled board with blocked and free cells.

Rolling over Labeled Cells Multiple Times. Puzzles where labeled cells may be rolled on several times can be solved by checking the reachable part of the state graph, starting at some initial state (or a set of possible initial states). Thus finding a rolling path for the die corresponds to finding a sufficiently large connected component in the state graph. The reachable part can for instance be determined by a breadth first search of the state graph. This implies the following proposition:

Proposition 1 *When the labeled cells of a rolling cube puzzle may be visited arbitrarily often, the puzzle can be solved in polynomial time. The time needed is at most linear in the complexity of the state graph.*

We therefore restrict our attention to the case where every labeled face must be visited exactly once.

3 Boards with Free Cells

In this section we show that solving puzzles on boards with labeled, free, and blocked cells is NP-complete. We then refine the proof to show that the problem remains NP-complete with only labeled and free cells.

Free and Blocked Cells. We show NP-hardness by a reduction from the Hamiltonian path problem in grid graphs. An *(induced) grid graph* is an induced subgraph of the infinite grid graph that has vertices (i, j), $i, j \in \mathbb{Z}$, and edges between vertices of distance one. Grid graphs are uniquely determined by their vertex set. Detecting a Hamiltonian path or cycle in grid graphs is NP-complete [8].

We sketch the reduction using the example in Fig. 4, which shows a grid graph and the corresponding board. For each vertex of the grid graph, the board has a labeled cell. For each edge of the grid graph, the board has a free cell. The labeled cells are labeled alternatingly with the labels 1 and 6. There is a Hamiltonian path in the grid graph if and only if it is possible to roll a die over the board such that it visits each labeled cell exactly once.

Theorem 2 *Deciding whether a die can roll along a path or cycle over a board with labeled, blocked, and free cells is NP-complete.*

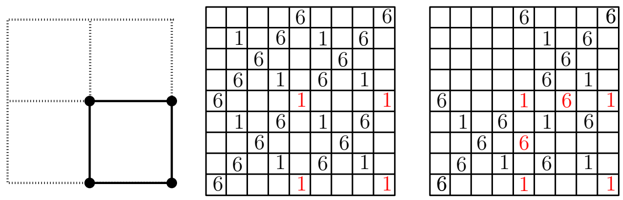


Figure 5: Reduction of a grid graph to a labeled board with free cells.

Free Cells. We now sketch a proof of how the above reduction extends to the case without blocked cells. We illustrate this by the example in Fig. 5. The leftmost figure shows a 3×3 grid with 4 vertices and 4 "non-vertices". On the corresponding board in the middle figure, vertices get the label 1 and non-vertices neighboring a vertex get the label 6. We label the cells between the ones corresponding to vertices and non-vertices such that a vertex is blocked in all four diagonal directions as shown in the figure. So far it is only possible to either roll over the 1-labels corresponding to vertices or to roll over all other labels. In the first case it is possible to roll over all 1-labels if and only if there is a Hamiltonian path in the grid graph. To allow rolling over all labeled cells (in the case of a Hamiltonian cycle) we introduce exactly one "gate" into the construction. The gate allows switching between the two sets of labels exactly once. This addition is shown in the rightmost figure: for one vertex on the boundary of the grid graph, we leave out one set of labels diagonally blocking it. Moreover, we add 6-labels halfway to both its neighbors. This guarantees that, if a die rolls over all 1-labels corresponding to vertices, then the die starts and ends at the same label, and thus we obtain a cycle. Using this construction we can prove the following theorem:

Theorem 3 *Deciding whether a die can roll along a path or cycle over a board with labeled and free cells is NP-complete.*

4 Boards without Free Cells

What happens when the board contains only labeled and possibly blocked cells? It remains open whether a polynomial-time algorithm can determine Hamiltonicity of a board, even when all the cells are labeled and when the labeling specifies the orientation of the die. In this section we provide a discussion of the problem, with the hope that our observations will lead to establishing a result. On fully labeled boards the following simple observation holds:

Observation 1 *Hamiltonian cycles cannot be rolled on fully labeled boards with an odd number of cells.*

Eliminating and Cutting. Consider the version of the problem where the labeled cells of the board also specify full die orientation, and let G be the state graph induced by such a board. First observe that, if G contains a vertex of degree at most 2, then a Hamiltonian cycle has only one way of visiting this vertex. Let a *chain* of the state graph be a path where all vertices except the first and the last have degree 2 and at least one vertex has degree 2. Such chains are effectively *forced*: they must appear in any Hamiltonian cycle. On the other hand, if a vertex u has degree more than 2 but is connected to two vertices of degree 2, then a Hamiltonian cycle will visit u using the two edges connecting u to its neighbors of degree 2. Based on these two observations, we define two operations on state graphs: (1) *Elimination*: if a vertex is incident to two forced edges, remove all other incident edges. (2) *Cutting*: if the two endpoints of a chain are also connected by an edge, remove the edge. We apply these two operations exhaustively on G to get a subgraph G' with possibly fewer edges, and such that G' has a Hamiltonian cycle if and only if G does. Observe that it is possible that G' is not a grid graph. The main question is whether G' has properties that enable us to determine its Hamiltonicity in polynomial time.

One of the properties that differentiates state graphs from grid graphs is that state graphs have forbidden configurations. One example of a forbidden configuration is a cycle of length 4 on a 2×2 grid. Other forbidden configurations are the maximum cycles on 3×2 and 4×2 grids. In fact, the shortest cycle in the state graph has length 8 and is the maximum cycle on a 3×3 grid. The cycle with the next shortest length has 10 vertices on a 5×2 grid. It turns out that these forbidden configurations disallow a large number of vertices of degree more than 2 to be packed closely together in the state graph. Let a *blob* of the state graph be a maximal set of connected vertices each having degree at least 3. The *depth* of a vertex v of a blob is the length of the shortest path from v to a vertex that is adjacent to at least one vertex of degree 2. The *depth of a blob* is the depth of the vertex with maximum depth among all vertices of the blob. We can show the following lemmata:

Lemma 4 *The depth of every blob in a state graph is at most* 2.

Lemma 5 *After applying the elimination and cutting operations to every vertex of the state graph, the depth of every blob in a state graph is at most* 1.

Whether the bound on the depth of blobs in state graphs is enough to determine its Hamiltonicity in polynomial time remains unclear. We conjecture that it is:

Conjecture 1 *On boards with labeled and possibly blocked cells, rollable Hamiltonian cycles can be determined in polynomial time.*

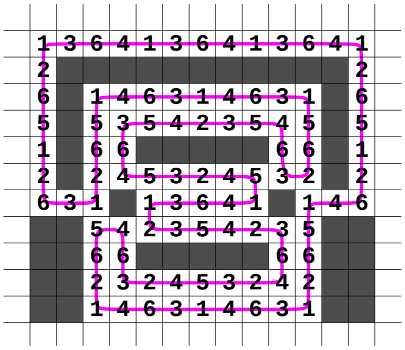


Figure 6: A labeling of the board containing more than one Hamiltonian cycle. The dark cells are blocked.

Uniqueness of Cycles? At first we conjectured that, if a state graph contains a Hamiltonian cycle, then this cycle is unique. If true, this property would possibly make it easier to determine Hamiltonicity in polynomial time, as it would increase the restrictions on the Hamiltonian cycle if one exists. The conjecture, however, is false for boards that contain blocked cells:

Observation 2 *There are boards with labeled and blocked cells in which rollable Hamiltonian cycles are not unique.*

The state graph in the example of Fig. 6 is composed of two cycles of length 8, where each cell of the first cycle is connected by a rollable path to a cell of the second. By carefully connecting copies of the state graph induced by the labeling of the board in Fig. 6, we can generate a state graph containing multiple Hamiltonian cycles. It is still unclear whether this counterexample leads to a hardness proof, or whether it can be conquered by dynamic programming. We believe the latter is the case.

It remains open whether non-uniqueness of Hamiltonian cycles holds for fully labeled boards. For such boards we pursued the following two approaches: enumeration for small boards and constructing cycles from corners. On small fully labeled boards we can test uniqueness of Hamiltonian cycles by enumerating all possible solutions by hand or by computer. To achieve this, we first enumerate all Hamiltonian cycles on the board; we then check whether the cycles are *rollable*, that is, whether a die can be rolled along the cycle, generating a consistent labeling, and starting and ending at the same state. For all rollable cycles, we check whether they are *uniquely rollable*, that is, given a labeling obtained by rolling a die along the cycle, we check whether this labeling can also be obtained by a different cycle.

Observation 3 *All rollable Hamiltonian cycles in fully labeled boards with side lengths at most* 8 *are unique.*

We explored how *border obsessiveness* (a term used to describe a strategy for solving jigsaw puzzles) can be applied to rolling cube puzzles. The following proposition shows that this strategy works at least to some extent. It can be proved by case analysis. Define the *k-neighborhood* of a cell be all cells that can be reached in k steps ignoring the labels of the cells.

Proposition 6 *If a labeling of an $n \times n$ board admits a Hamiltonian cycle, then the path of the die within the* 3*-neighborhoods of the corners is unique (up to direction). For $n = 4$, the rolling pattern at a corner is determined by the* 2*-neighborhood of this corner. For $n > 4$, the* 5*-neighborhood determines the pattern.*

Finding Long Rollable Cycles. Suppose we are given a fully labeled board and we ask for a rollable cycle that visits the maximum number of cells, without necessarily visiting them all. We show that finding a maximum cycle on a fully labeled board is NP-complete.

Theorem 7 *Deciding whether a die can roll along a cycle of length K over a board with labeled cells is NP-complete, even given starting position and orientation.*

Acknowledgment

We would like to thank Joseph O'Rourke for suggesting and discussing the problem with us. We would also like to thank Robert Abbott, Frank Hoffmann, Klaus Kriegel, Günter Rote, and the participants of the 4th GWOP (Group Emo Workshop on Open Problems) for helpful discussions.

References

[1] R. Abbott. *Mad Mazes: Intriguing Mind Twisters for Puzzle Buffs, Game Nuts and Other Smart People*. Adams, 1990.

[2] R. Abbott. *SuperMazes: Mind Twisters for Puzzle Buffs, Game Nuts, and Other Smart People*. Prima Publ., 1997.

[3] E. D. Demaine and J. O'Rourke. Open problems from CCCG 2005. In *Proc. 18th Canadian Conference on Computational Geometry*, pages 75–80, 2006.

[4] M. Gardner. Mathematical games column. *Scientific American*, 209(6):144, Dec. 1963. Solution in Jan. 1964 column.

[5] M. Gardner. Mathematical games column. *Scientific American*, 213(5):120–123, Nov. 1965. Problem 9. Solution in Dec. 1965 column.

[6] M. Gardner. Mathematical games column. *Scientific American*, 232(3):112–116, Mar. 1975. Solution in Apr. 1975 column.

[7] J. Harris. Single vacancy rolling cube problems. *J. Recreational Mathematics*, 7(3):220–223, 1974.

[8] A. Itai, C. H. Papadimitriou, and J. L. Szwarcfiter. Hamilton paths in grid graphs. *SIAM J. Comput.*, 11(4):676–686, 1982.

[9] R. Sprague. *Recreations in Mathematics*. Blackie, Glasgow, 1963. Problem 3: Heavy Boxes. Translation by T. H. O'Beirne.

[10] Charles W. Trigg. Tetrahedron rolled onto a plane. *J. Recreational Mathematics*, 3(2):82–87, 1970.