2,664 Coin-Sliding Font Puzzles

Erik D. Demaine* Martin L. Demaine*

Abstract

We present two font designs, each with 37 symbols (letters, digits, and slash), as grid configurations of the same number of coins. Each pair of symbols (say, A and B) forms a puzzle: re-arrange the first symbol (A) into the second (B) by a sequence of moves. Each move picks up one coin and places it in an empty grid cell that is adjacent to at least two other coins (the "2-adjacency" rule). We also present an online puzzle video game to play all 2,664 of these puzzles, where you can try to set the record on the minimum number of moves.

1 Coin-Sliding Puzzles

At our first G4G (G4G5 in 2002), we presented several new coin-sliding puzzles [DD04] based on our research with Helena Verrill [DDV02]. Figure 1 shows one example. In this type of puzzle, the goal is to transform the start configuration (drawn on the left) into the target configuration (drawn on the right) via a sequence of "moves". Each ***move*** picks up one coin and places it in an empty grid cell that is ***adjacent to at least two other coins*** (the ***2-adjacency*** rule).[1] A second goal is to minimize the number of moves that achieve the desired transformation.

Martin Gardner wrote about puzzles like this [Gar75], but on the triangular grid. Indeed, staying on the triangular grid is probably the original motivation for the 2-adjacency rule, as these

*MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA, {edemaine,mdemaine}@mit.edu

[1] We do not consider a "sliding" constraint (continuous planar motion of the coin without collision), which is present in only one puzzle in [Gar75]. The more precise name for these puzzles is "coin-moving puzzles", as in [DDV02], but we use the less formal term "coin-sliding" here.

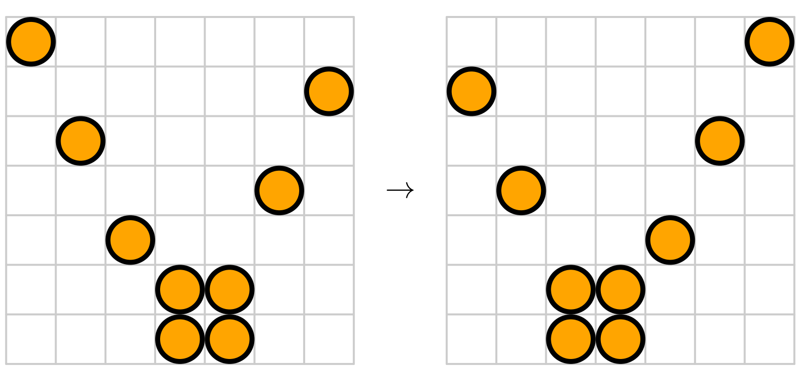

Figure 1: Puzzle 9 from [DD04]. The n-coin version of this puzzle is the asymptotically hardest puzzle: it requires $\Omega(n^3)$ moves, and all n-coin coin-sliding puzzles on the square grid can be solved in $O(n^3)$ moves [DDV02]. The exact constant factor is unknown, however.

moves force the coins to remain on a triangular grid. But triangular-grid coin-sliding puzzles turn out to be much simpler, both from a puzzle perspective and in terms of mathematics and algorithms [DDV02]. Thus we focus here on the square-grid coin-sliding puzzles, which originate with Harry Langman [Lan51].

Our main result with Verrill [DDV02] is a sufficient condition for a coin-sliding puzzle on the square grid to have a solution, and a corresponding algorithm to solve these puzzles. To state the result, we need to define the notion of "span" of a configuration of coins. Imagine you have a bag full of extra coins, and you place as many as you can onto the board while still respecting the 2-adjacency rule for each placement. The ***span*** is the resulting configuration, which is a rectangle or disjoint union of rectangles (with at least two blank rows in between the rectangles). The span represents the maximum set of reachable positions that the coins could reach (even without the bag of extra coins). Making moves can therefore never increase the span, only decrease it accidentally.

Our sufficient condition is that "two extra coins suffice" in the following technical sense:

Theorem 1 [DDV02, Theorem 2] *Configuration A of coins can be re-arranged into configuration B via 2-adjacency moves on the square grid if there are two "extra" coins e_1 and e_2, each adjacent to two other coins (not each other), such that the span of $A - e_1 - e_2$ contains the span of $B - e_1 - e_2$. The number of moves is $O(n^3)$ where n is the number of coins, and the moves can be found algorithmically in $O(n^3)$ time.*

This theorem tells us an easy way to design puzzles that are guaranteed solvable: just make sure the spans of the two configurations match (or configuration A's span is more than configuration B's span), and make sure there are two extra coins. However, it remains an open problem how to find the fewest moves to solve such a puzzle.

2 Coin-Sliding Fonts

Over the past dozen years, we have developed several different typefaces/fonts that express text through mathematical theorems or open problems in broadly accessible forms, often through the use of puzzles. The fonts are all free to play with on the web.[2]

In this paper, we revisit sliding-coin puzzles from the perspective of mathematical/puzzle fonts. Figures 2 and 3 show our two font designs, one with 12 coins on a 5×7 rectangle and one with 13 coins on a 5×9 rectangle. Each font consists of 37 symbols (26 letters, 10 digits, and slash[3]), where each symbol is made from the same number of coins arranged on the square grid within the same size of rectangle (which is also the span of the configuration). You can write messages in these fonts using our online web application.[4]

Every pair of symbols within the same font defines a coin-sliding puzzle. Thus we obtain $37 \cdot 36 = 1{,}332$ puzzles within each font, for a total of 2,664 puzzles.

2.1 Puzzle Video Game

We implemented a puzzle video game for playing all of these puzzles. You can play on any device with a web browser[5] or using an Android app[6]. Figure 4 shows what the user interface looks like.

[2] http://erikdemaine.org/fonts/

[3] We included slash because it plays a significant role in many of our early coin-sliding puzzles [DDV02, DD04].

[4] http://erikdemaine.org/fonts/coinsliding/

[5] https://coinsliding.erikdemaine.org/

[6] https://play.google.com/store/apps/details?id=org.erikdemaine.coinsliding

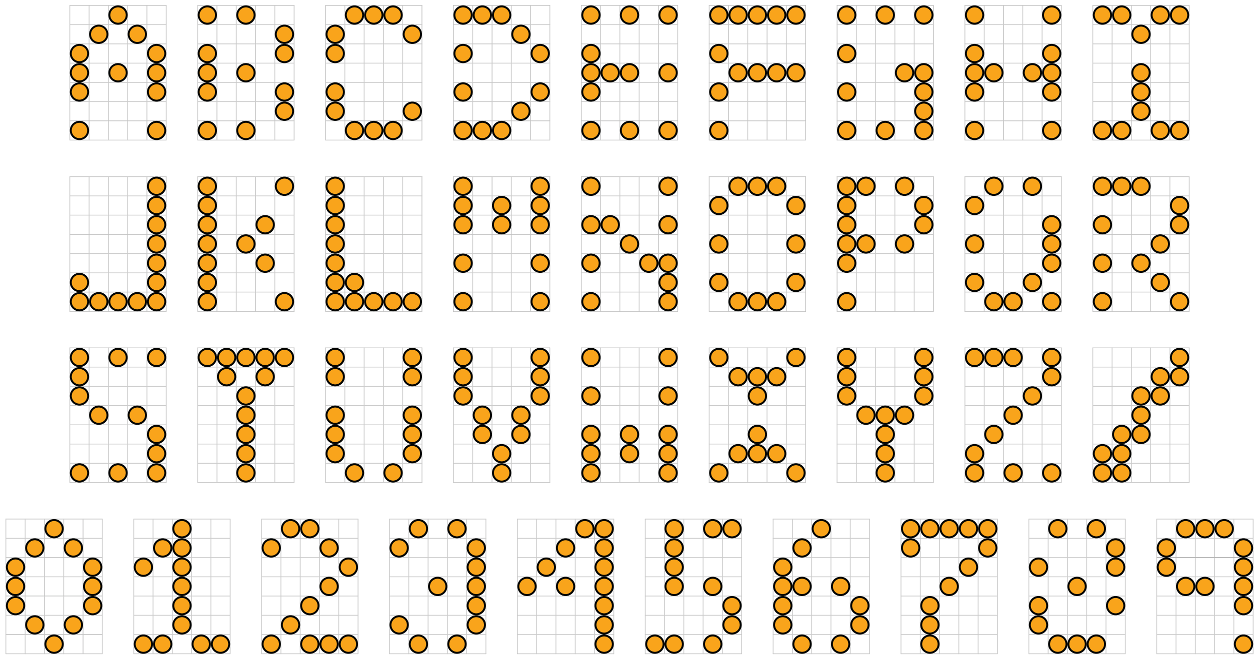

Figure 2: 5×7 coin-sliding font. Each symbol consists of 12 coins.

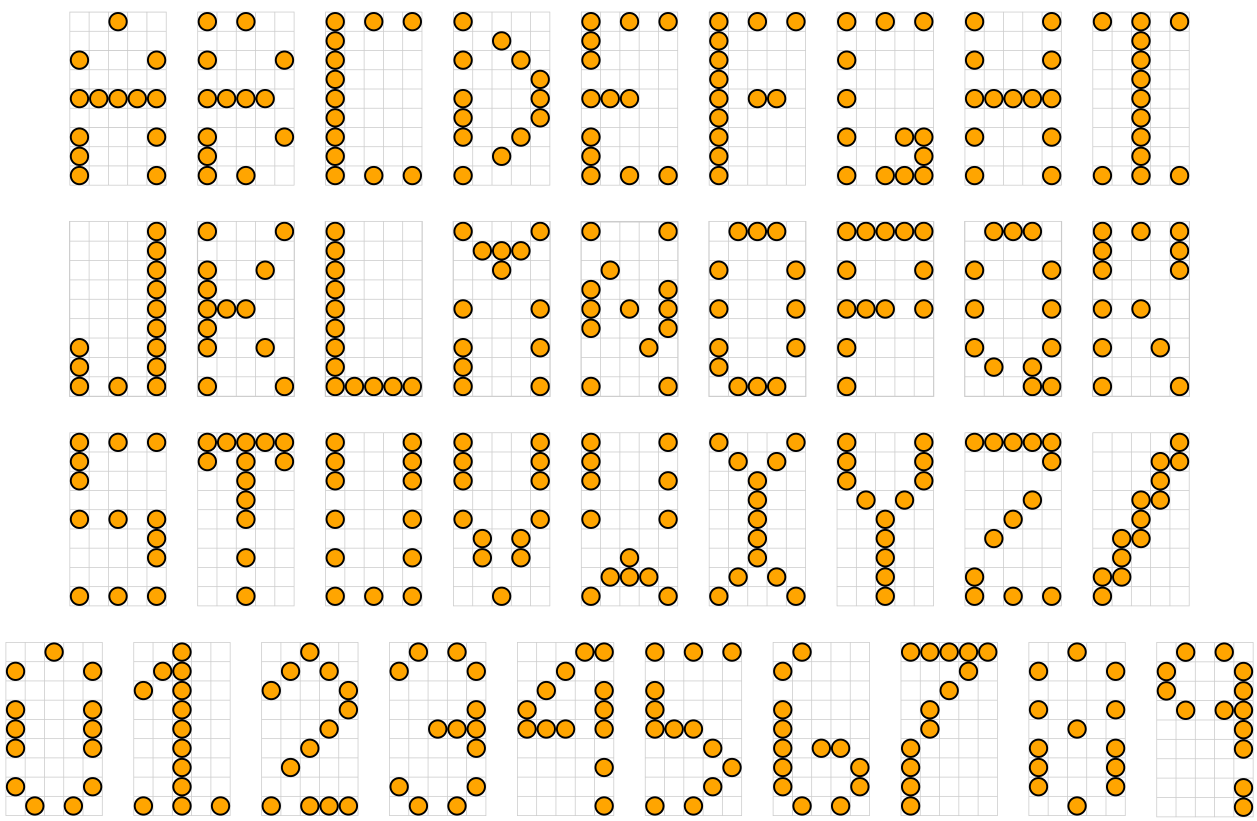

Figure 3: 5×9 coin-sliding font. Each symbol consists of 13 coins.

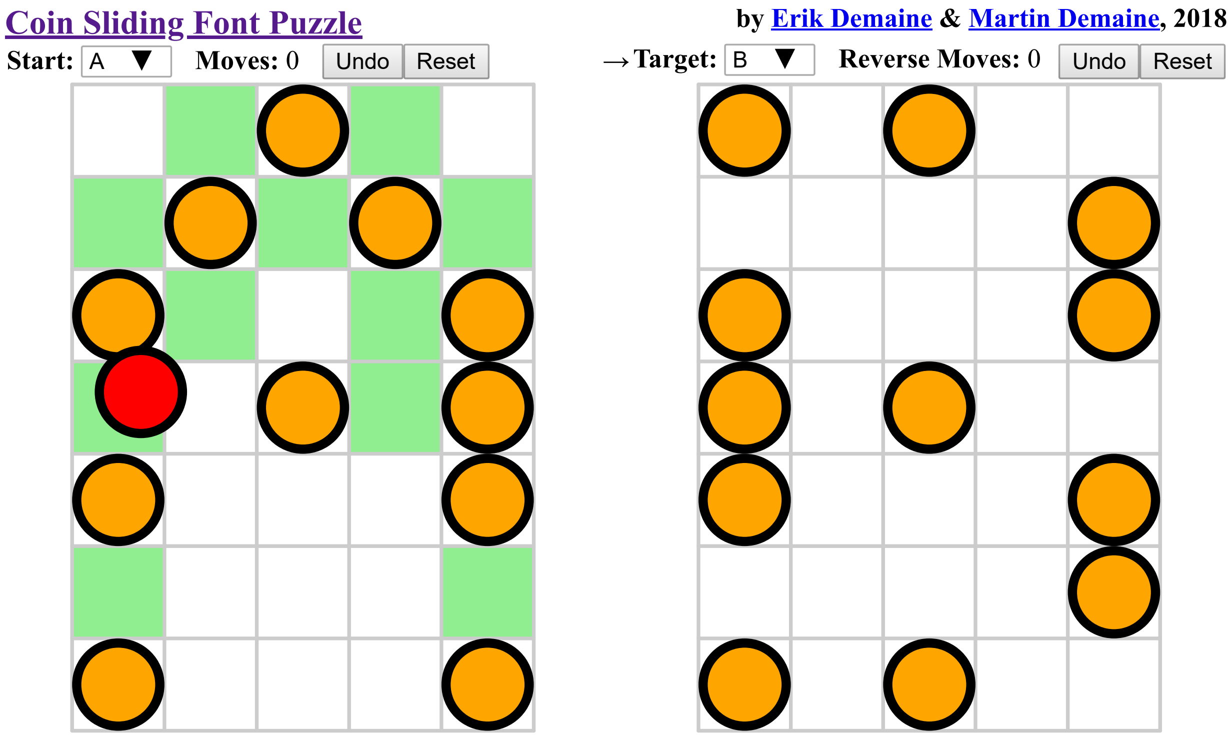

Figure 4: Coin-sliding puzzle video game. Play online at https://coinsliding.erikdemaine.org/

To play a puzzle, you select a font (5×7 or 5×9), then choose a puzzle from "All puzzles in family" or using the "Start" and "Target" selections. Dragging coins makes moves. If you get stuck, you can "Undo" move by move, or "Reset" to the beginning.

When you solve a puzzle, you can post your score (number of moves) along with your name. Help us find good solutions to all the puzzles! This will give us a better understanding of the number of moves required to solve coin sliding puzzles, which remains a mathematical mystery.

An example solution animation can be found on a special website.[7]

The source code is also available.[8]

2.2 Proof of Solvability

We prove that all of the puzzles are solvable. Theorem 1 covers most of the puzzles, as they all have span equal to the full rectangle (either 5×7 or 5×9), even after removing two well-chosen coins. However, not all of the configurations have extra coins neighboring two other coins, so they are not valid choices for the target configuration B in Theorem 1. Nonetheless, we can show that all symbol configurations are ***reachable*** from valid B configurations in Theorem 1 (and thus from all valid A configurations, including all other symbols). Figures 5 and 6 prove each case, either highlighting two suitable extra coins, or showing a sequence of ***reverse moves*** (with arrows) that free up two suitable extra coins. A reverse move is the exact opposite of a 2-adjacency move, i.e., it moves a coin from a position adjacent to at least two other coins to any other position. Each sequence of reverse moves can be verified by dragging coins on the right side of the puzzle video game's user interface.[4]

[7]http://erikdemaine.org/fonts/coinsliding/g4g.html
[8]https://github.com/edemaine/coinsliding

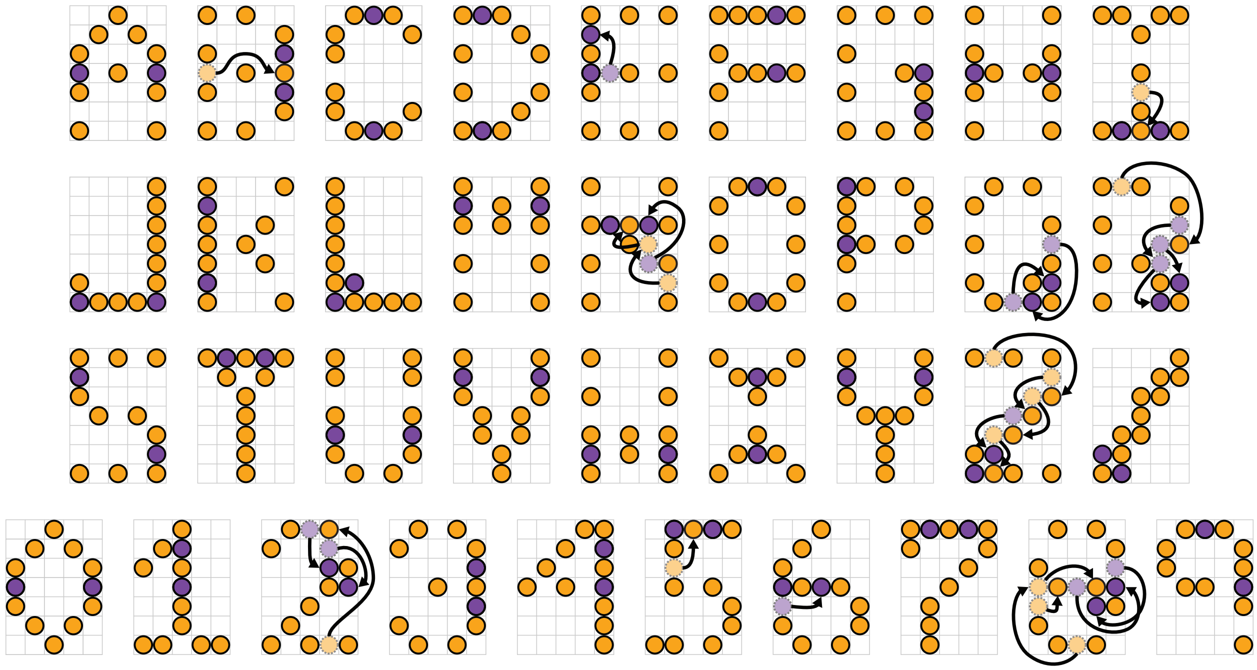

Figure 5: Reachability proof for 5 × 7 coin-sliding font.

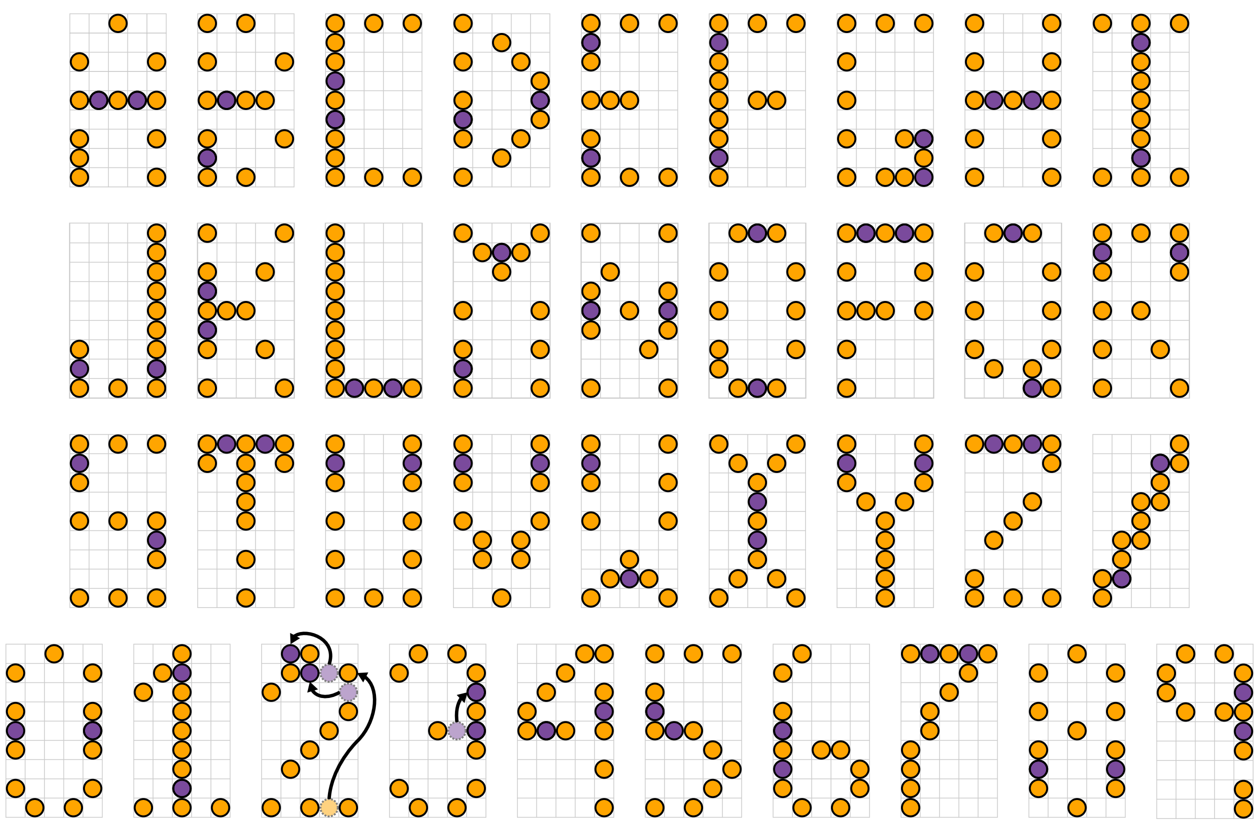

Figure 6: Reachability proof for 5 × 9 coin-sliding font.

References

[DD04] Erik D. Demaine and Martin L. Demaine. Sliding-coin puzzles. In *Tribute to a Mathemagician*, pages 63–72. A K Peters, 2004.

[DDV02] Erik D. Demaine, Martin L. Demaine, and Helena Verrill. Coin-moving puzzles. In R. J. Nowakowski, editor, *More Games of No Chance*, pages 405–431. Cambridge University Press, 2002. Collection of papers from the MSRI Combinatorial Game Theory Research Workshop, Berkeley, California, July 2000. http://www.arXiv.org/abs/cs.DM/0204002.

[Gar75] Martin Gardner. Penny puzzles. In *Mathematical Carnival*, chapter 2, pages 12–26. Alfred A. Knopf, New York, 1975. Appeared in *Scientific American*, 214(2):112–118, February 1966, with solutions in 214(3):116–117, March 1966.

[Lan51] Harry Langman. Curiosa 261: A disc puzzle. *Scripta Mathematica*, 17(1–2):144, March–June 1951.