Computational Geometry: Theory and Applications 110 (2023) 101949

Contents lists available at ScienceDirect

Computational Geometry: Theory and Applications

journal homepage: www.elsevier.com/locate/comgeo

Rectangular Spiral Galaxies are still hard [☆], [☆☆]

Erik D. Demaine [a], Maarten Löffler [b], Christiane Schmidt [c],[*]

[a] Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, United States of America
[b] Department of Information and Computing Sciences, Universiteit Utrecht, Netherlands
[c] Department of Science and Technology, Linköping University, Sweden

ARTICLE INFO

Article history:
Received 30 September 2021
Received in revised form 5 October 2022
Accepted 6 October 2022
Available online 10 October 2022

Keywords:
Pencil-and-paper puzzle
Spiral Galaxies
NP-completeness
Non-crossing matching

ABSTRACT

Spiral Galaxies is a pencil-and-paper puzzle played on a grid of unit squares: given a set of points called *centers*, the goal is to partition the grid into polyominoes such that each polyomino contains exactly one center and is 180° rotationally symmetric about its center. We show that this puzzle is NP-complete, ASP-complete, and #P-complete even if (a) all solutions to the puzzle have rectangles for polyominoes; or (b) the polyominoes are required to be rectangles and all solutions to the puzzle have just 1×1, 1×3, and 3×1 rectangles. The proof for the latter variant also implies NP/ASP/#P-completeness of finding a noncrossing perfect matching in distance-2 grid graphs where edges connect vertices of Euclidean distance 2. Moreover, we prove NP-completeness of the design problem of minimizing the number of centers such that there exists a set of galaxies that exactly cover a given shape.

1. Introduction

Spiral Galaxies is a pencil-and-paper puzzle published by Nikoli since 2001 [2] under the name “Tentai Show” [3]. It is played on a grid of unit squares with given “centers” — points located at grid vertices, cell centers, or edge midpoints. The goal is to decompose the grid into polyominoes called “galaxies” such that each galaxy contains exactly one center and is 180° rotationally symmetric about its center; see Fig. 1(a). The solution for a Spiral Galaxies puzzle may not be unique, but typically puzzles are designed to have a unique solution.

Friedman [4] proved Spiral Galaxies NP-complete for general polyomino galaxies. His proof uses galaxies of area up to 22. Fertin et al. [5] proved NP-completeness for galaxies of area at most 7, while the puzzle becomes solvable in polynomial time if galaxies are restricted to have area at most 2 or to be squares. How much simpler can we make the galaxy shapes and still have Spiral Galaxies puzzles be NP-complete?

In this paper, we consider Spiral Galaxies with rectangular galaxies, in two possible senses:

1. In ***Rectangular Galaxies***, the polyomino galaxies are required to be rectangles. This is an additional restriction on the puzzle beyond 180° rotational symmetry around the unique center contained in each rectangle.

[☆] This research was performed in part at the 30th and the 33rd Bellairs Winter Workshop on Computational Geometry. We thank all other participants for a fruitful atmosphere.

[☆☆] A preliminary extended abstract appeared as a preprint in the Proceedings of the 37th European Workshop on Computational Geometry [1].

[*] Corresponding author.
E-mail addresses: edemaine@mit.edu (E.D. Demaine), m.loffler@uu.nl (M. Löffler), christiane.schmidt@liu.se (C. Schmidt).

https://doi.org/10.1016/j.comgeo.2022.101949

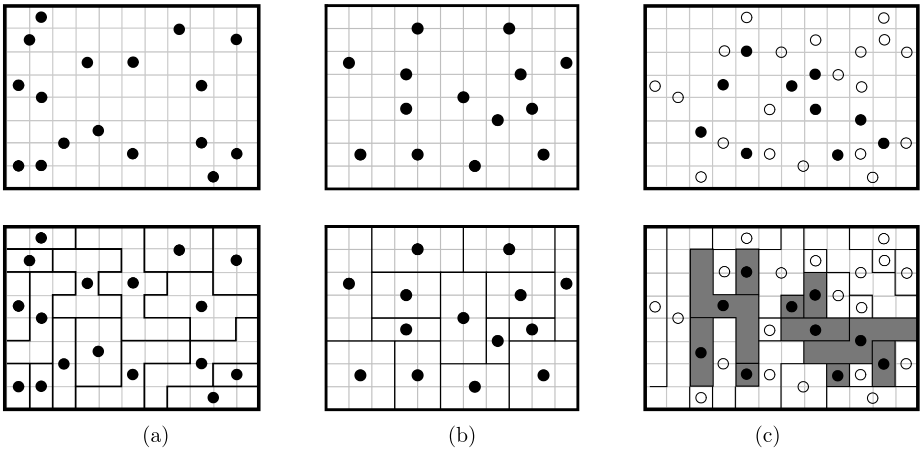


Fig. 1. Several styles of Spiral Galaxies puzzles (top) and their solutions (bottom). (a) Classic Spiral Galaxies puzzle. (b) Rectangular Galaxies puzzle. (c) Spiral Galaxies puzzle with black and white centers such that the polyominoes containing the black centers in the solution yield a picture.

2. Spiral Galaxies puzzles designed so that all their solutions (if any) have rectangular galaxies. In other words, we have a *promise* that all solutions are rectangular.

Fig. 1(b) shows an example of a Spiral Galaxies puzzle where the unique solution is rectangular, so it is a valid input to either Rectangular Galaxies or rectangular-promise Spiral Galaxies.

We prove that both of these puzzle variants are NP-complete and ASP-complete, and that the corresponding problem of counting the number of solutions is #P-complete. (Refer to [6, Chapter 28] and [7,8] for definitions of #P-completeness and ASP-completeness, respectively.) In fact, hardness of rectangular-promise Spiral Galaxies implies hardness of Rectangular Galaxies. For Rectangular Galaxies, we further show that the problem remains NP/ASP/#P-complete even when all solutions use only 1×1, 1×3, and 3×1 rectangles (a stronger promise), and hence also if the puzzle is restricted to use only such rectangles. This special case is in some sense the smallest interesting scenario: if we restrict to only 1×1, 1×2, and 2×1 rectangles, then each center determines the shape of its containing polyomino (of area 1 or 2 depending on whether the center is at a cell center or edge midpoint, respectively), making puzzles easy to solve. Our proof for 1×1, 1×3, and 3×1 rectangles also establishes NP-completeness of finding a noncrossing matching in "distance-2 grid graphs", whose vertices are a subset of the integer lattice and whose edges connect vertices at Euclidean distance exactly 2.

In another variant of the puzzle, a subset of the centers is colored black, and the polyominoes that contain these centers reveals a picture once solved [3]; see Fig. 1(c). Logic puzzles whose solutions yield pictures are a popular genre; one famous example is the nonogram [9]. Such Spiral Galaxies puzzles are also the subject of mathematical fonts [10]. Constructing an interesting puzzle such that its solution is a given target shape is nontrivial: while a valid puzzle trivially exists, by simply placing a center in every grid cell, the resulting puzzle is clearly not interesting. We are thus also interested in finding the minimum number of centers such that there exist galaxies that exactly cover a given shape. We prove this puzzle design problem to be NP-complete.

1.1. Notation and preliminaries

In Spiral Galaxies, a ***board*** consists of an $m \times n$ grid of unit squares called ***cells***. A ***puzzle*** consists of a board and a set of ***centers*** placed at cell centers, edge midpoints, or grid vertices. Either all centers have the same color, or a subset of the centers may be colored black. A solution to the puzzle consists of a partition of the grid into polyominoes, called ***galaxies***, such that each galaxy contains a single center and is 180° rotationally symmetric about its center.

Our reductions are from the Planar Positive 1-in-3 SAT problem, a well-known NP-complete, ASP-complete, and #P-complete problem [11–13]:

Definition 1. An instance of the Planar Positive 1-in-3 SAT problem is a ***formula*** $F = (\mathcal{C}, \mathcal{V})$ consisting of a set $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$ of k ***clauses*** over ℓ ***variables*** $\mathcal{V} = \{x_1, x_2, \dots, x_\ell\}$. Each clause in F is a set of at most three variables, and in particular contains variables only in their positive form (no negation). The variable–clause incidence graph G is planar. A clause is satisfied if and only if it contains exactly one TRUE variable, and the formula F is satisfied if and only if all its clauses are satisfied. The goal is to find a Boolean assignment to the variables that satisfies the formula F.

1.2. Organization

The remainder of this article is organized as follows. In Section 2, we prove that Spiral Galaxies is NP/ASP/#P-complete when all solutions use rectangular galaxies. In Section 3, we show that Rectangular Galaxies is NP/ASP/#P-complete when

Fig. 2. (a) Filler gadget to fill the space in between all other gadgets. (b) Forced solution.

all solutions use rectangular galaxies of dimensions 1 × 1, 1 × 3, and 3 × 1. In Section 4, we relate this problem to finding noncrossing perfect matchings in distance-2 grid graphs, proving NP/ASP/#P-completeness of the latter. In Section 5, we show that minimizing the number of centers when generating a puzzle with a given output shape is NP-hard. In Section 6, we discuss the possibility of allowing multiple solutions in Spiral Galaxies puzzles, and we investigate a small puzzle related to a Spiral Galaxies font. Finally, we conclude in Section 7.

2. Rectangular-promise Spiral Galaxies

In this section, we show that solving Spiral Galaxies puzzles promised to have solutions with only rectangular galaxies is NP-complete and ASP-complete, and that the corresponding problem of counting the number of solutions is #P-complete.

We give a reduction from Planar Positive 1-in-3 SAT. Given an instance F of Planar Positive 1-in-3 SAT with incidence graph G, we show how to turn a rectilinear planar embedding of G into a Spiral Galaxies puzzle P such that a solution to P yields a solution to F, thereby showing NP-completeness. Furthermore, there will be a one-to-one correspondence between solutions of P and solutions of F, showing #P-completeness and ASP-completeness.

Overview and filler gadget. At a high level, our reduction consists of several gadgets: "variable" gadgets representing the variables of F; "wire" gadgets to connect variables to clauses; and "clause" gadgets to form the clauses of F. Refer ahead to Fig. 13 for a complete example of the reduction.

For each region of the board that is not part of these gadgets, we fill the region with a ***filler gadget***, which has a center in every cell of the region. Fig. 2 shows the filler gadget for a 2 × 2 region. More generally, in every 2 × 2 square within the region, we can argue locally that the four corresponding galaxies must each consist of a single cell (the one containing the center): the edges between cells must be galaxy boundaries to separate the centers into separate galaxies, and then 180° rotational symmetry forces galaxy boundaries around the 2 × 2 square. As long as the region is the union of such 2 × 2 squares, the filler gadget must consist entirely of single-cell galaxies, without any interaction with the other gadgets in the construction. On the other hand, if the region has a width-1 row or a height-1 column ("thickness 1"), then the galaxy at the center of that cell might include cells surrounding the filler gadget. We must therefore guarantee that every region between other gadgets is the union of 2 × 2 squares ("thickness 2"), so that the filler gadget has a forced solution of single-cell galaxies.

Variable gadget and loop. Fig. 3 shows the overall plan for a variable gadget: a thickness-1 "variable loop" that follows a long horizontal rectangle with regularly spaced bumps on the top and bottom sides, where each bump has zero or one connection to an incident wire (which has thickness 2).

In more detail, a ***variable loop*** is a thickness-1 loop built out of the subgadgets in Figs. 4 and 5. Every center is at a cell center, spaced modulo 3 along the thickness-1 loop. Each center in the middle of a straight piece can choose an $x \times 1$ rectangular galaxy for $x \in \{1, 3, 5\}$, which then forces the next galaxy along the straight part to be $6 - x \times 1$, etc., as in Fig. 4(b–d). Each corner subgadget of Fig. 5 forbids the center at distance 2 from the corner from having a 1 × 1 galaxy, as then the galaxy centered at distance 1 from the corner fails to be 180° rotationally symmetric; see Fig. 5(d). Our "bumpy rectangle" design from Fig. 3 guarantees that every straight portion of a variable loop is adjacent to at least one corner, and we can further arrange that all corners have the same alignment. These properties forbid one global pattern (subfigure (d) with blue 5 × 1 galaxies) and leave two possible solution patterns: all galaxies are 3 × 1 or 1 × 3 (subfigure (b)), or galaxies alternate between blue 1 × 1 and red 5 × 1/1 × 5 (subfigure (c)). These two solutions to the variable gadget correspond to setting the variable false or true, respectively.

Wire gadget. While thickness 1 was useful to force the variable gadget to have exactly two solutions, it seems difficult to copy that truth value onto multiple thickness-1 wires. We thus introduce thickness-2 ***wire gadgets*** as shown in Fig. 6. In their simplest form, a wire gadget is a vertical width-2 rectangle with a center at edge midpoints in the even rows modulo 2. The two intended solutions use alternating 2 × 1 and 2 × 3 galaxies as shown in Fig. 6(b–c), corresponding to false and true signals respectively.

Unfortunately, the basic wire also allows two nonrectangular solutions, as shown in Fig. 6(d–e). Each of these undesired solutions is prevented by a corresponding ***shift gadget*** shown in Figs. 7 and 8 respectively. In either gadget, covering the corner pixels in the middle row forces one side of the wire to use rectangles. By guaranteeing that each wire shifts at least once to the right and at least once to the left (including possibly one shift canceling out the other), we avoid the undesired solutions, forcing one of the two intended solutions in Fig. 6(b–c).

Each shift gadget can shift the wire horizontally by an arbitrary amount ≥ 3, as shown in Fig. 9. The height-1 transition row makes it easy to argue that the same solutions in Figs. 7 and 8 are forced. Thus a wire can approximately follow any desired y-monotone path that starts and ends with a vertical segment. The x coordinates of the start and end vertical

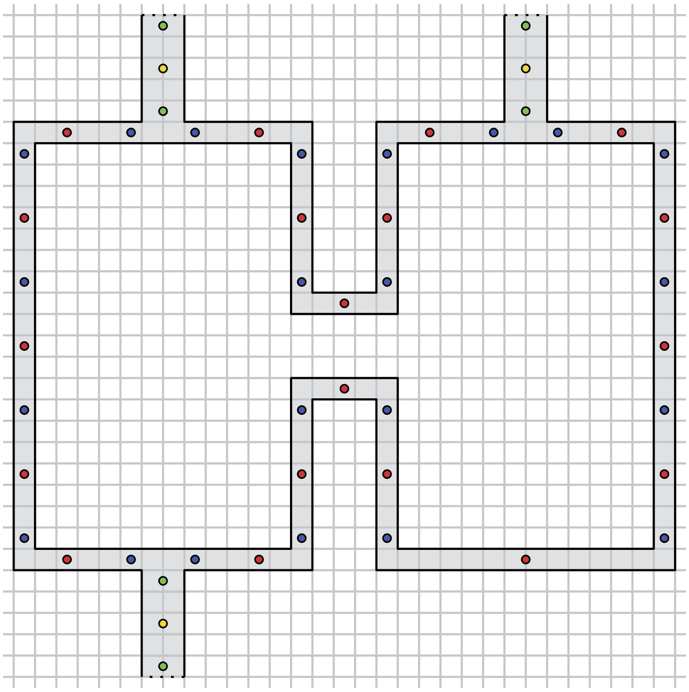


Fig. 3. Overall construction of a variable gadget as a sequence of bumps on the top and bottom, where each bump can have a single connection to an incident wire on that side (Fig. 10). Bumps without such a connection, such as the one in the bottom right, use a single center. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

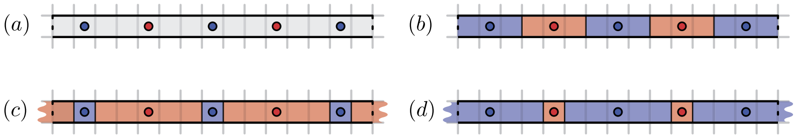


Fig. 4. (a) A straight piece of a variable gadget. (b–c) Two intended valid states. (d) Undesired but valid state.

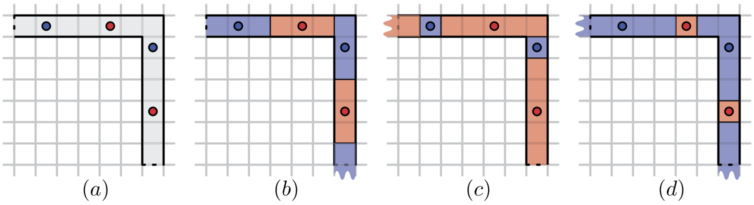


Fig. 5. (a) A corner of a variable gadget. (b–c) Two valid states. (d) The third state is no longer valid.

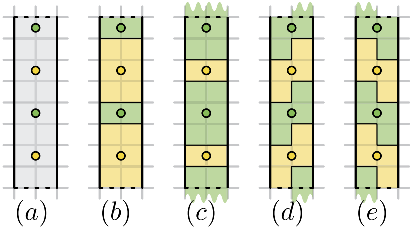


Fig. 6. (a) A wire gadget for connecting a variable to a clause. (b–c) Intended valid rectangular states. (d–e) Undesired but valid states.

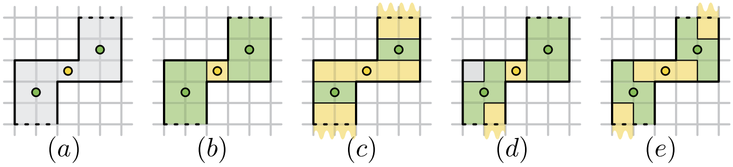


Fig. 7. (a) A shift to the right. (b—c, e) Three states are still valid. (d) The undesired wire state from Fig. 6(d) is no longer valid.

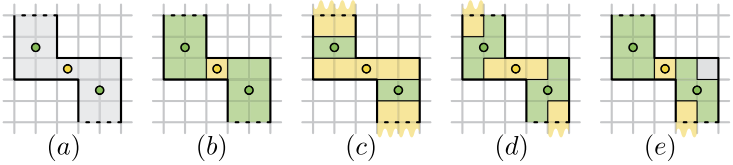


Fig. 8. (a) A shift to the left. (b—d) Three states are still valid. (e) The undesired wire state from Fig. 6(e) is no longer valid.

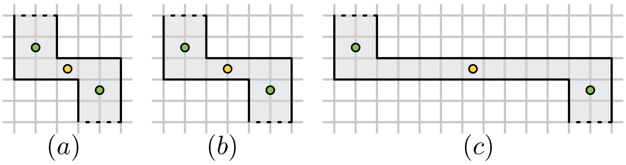


Fig. 9. Shift gadgets from Figs. 7 and 8 can shift by an arbitrary horizontal amount ≥ 3. (a, c) An even shift places the middle center at a cell center. (b) An odd shift places the middle center at an edge midpoint.

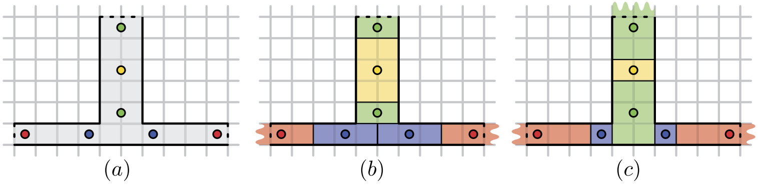


Fig. 10. (a) A place where a wire from the variable loop is connected. (b–c) Two compatible valid states.

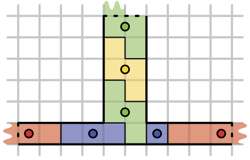


Fig. 11. Why we needed U-turns in wire gadgets: connections to variables are insufficient to force the intended rectangular solutions.

segments can be specified exactly; the only approximations are that the shifts need $\Omega(1)$ room to navigate, and we always use at least two shifts.

Connecting variables to wires. Fig. 10 shows how to communicate the value of a variable loop into the signal of a wire. The vertical thickness-2 wire attaches to two consecutive empty cells of a horizontal segment of the variable loop. If the wire's galaxies capture those two cells, then the two adjacent centers of the variable loop must have 1×1 galaxies, forcing alternation between 1×1 and 1×5 thereafter, as in Fig. 10(c). Conversely, if the wire's galaxies do not capture those two cells, then the two adjacent centers of the variable loop must capture them, forcing the one other solution of the variable loop with 1×3 rectangles, as in Fig. 10(b).

Fig. 11 shows that the unintended solutions of wire gadgets from Fig. 6(d–e) can propagate into the variable gadget. Thus connections to variable gadgets are insufficient to fix these problems, which is why we needed to introduce the shift gadgets of Fig. 7 and 8 to fix wire gadgets locally.

If we connected multiple wires to the same straight segment of a variable gadget, then the straight segment between the two connections would not be incident to any variable corners, so it could be in the state of Fig. 4(d). This is why we limit each bump of the variable loop to having at most one connection to a wire, so that every connection gadget from Fig. 10 is surrounded on either side by a corner gadget from Fig. 5.

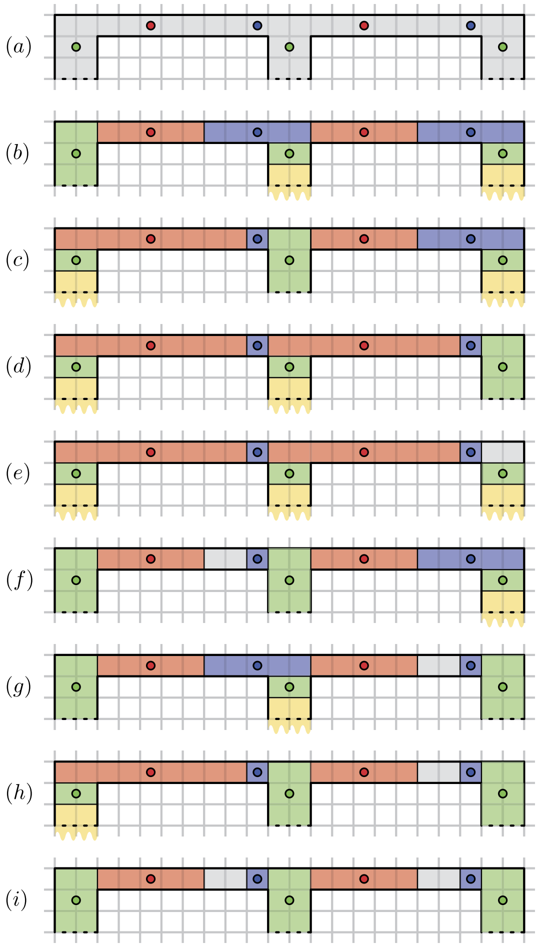


Fig. 12. The clause gadget connects three wire gadgets (green). (b–d) Valid solutions with exactly one true signal. (e) Impossibility with zero true signals. (f–h) Impossibility with two true signals. (i) Impossibility with three true signals.

Each connection between variable and wire removes one alternation from the variable gadget, as indicated by the coloring in Fig. 10. If the total number of connections from a variable gadget is odd, then we would have a parity mismatch around the variable loop. Fig. 3 illustrates a parity fix by placing a single center in the middle of a bump of the variable gadget that has no connection to a wire (similar to Fig. 9). For simplicity, we can use this construction for every bump with no connection to a wire. Because the two adjacent corners turn in the same direction, this center can be covered in exactly two ways: with a rectangle that is either the full horizontal width or 2 smaller.

Clause gadget. The last gadget is the clause gadget shown in Fig. 12, which implements a 1-in-3 constraint between three incident wire gadgets via a 1×22 rectangle. Although we draw all three wire gadgets as below the clause gadget, each wire could in fact be either above or below the clause gadget. Fig. 12(b–d) illustrates that the clause gadget has a (unique) solution when exactly one of the wires carries a true signal. Indeed, any true signal represented by a wire with a galaxy covering two cells of the clause gadget locally forces the galaxies of the one or two adjacent centers within the clause, which then propagates to force the galaxies of the other centers within the clause. This forced solution turns out to be inconsistent with any of the other wires carrying a true signal, causing overlapping galaxies or uncovered cells as shown in Fig. 12(e–i). Therefore the clause gadget has a (unique) solution if and only if exactly one of the incident wires carries a true signal.

Putting pieces together. For the global construction, we construct an orthogonal planar embedding of G where each vertex is represented by a horizontal line segment with endpoints on the grid, and each edge is represented by a polygonal line on the grid that is y-monotone and starts and ends with a vertical segment. Such an embedding can be obtained by constructing a planar straight-line embedding of G with vertices on an $O(n) \times O(n)$ grid [14]; applying the transformation of [15, Theorem 5] into a y-monotone orthogonal drawing where vertices are horizontal segments; and then staggering vertices with equal y coordinates. The resulting grid size is $O(n) \times O(n^2)$.

Fig. 13(a) shows a particularly structured planar embedding of G where the variables all lie on a horizontal line, every clause is either above or below this line, and variable–clause connections do not cross the line or each other. Then it is easy to draw each variable–clause connection with a y-monotone path that starts and ends with a vertical segment. Mulzer and

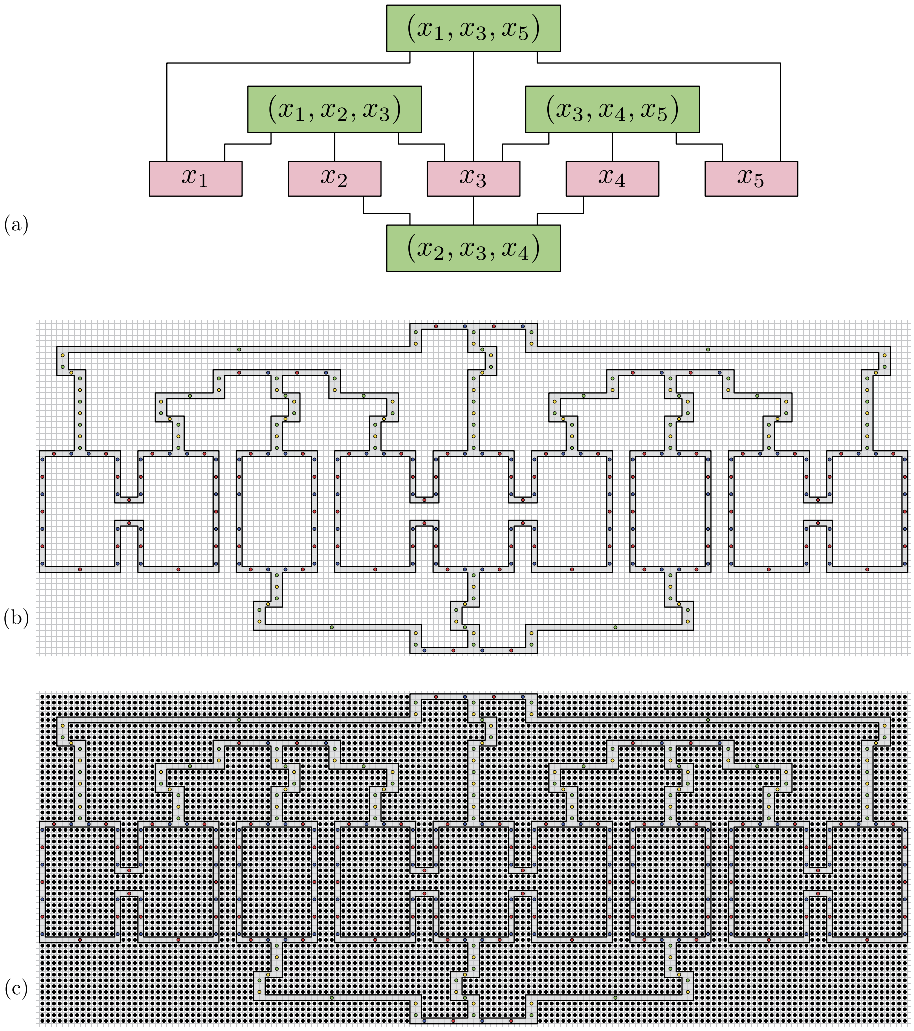


Fig. 13. (a) An embedded instance of Planar Positive 1-in-3 SAT, with all variables on a common horizontal line and all clauses either above or below the line. (b) The corresponding gadgets of Spiral Galaxies, excluding filler. (c) Full Spiral Galaxies puzzle with filler gadgets.

Rote [12] proved that Planar Positive 1-in-3 SAT is NP-complete in this form. Their reduction has also been observed to be the end of a chain of parsimonious reductions from 3SAT [16], so it establishes ASP- and #P-completeness too. We describe the more general reduction in the previous paragraph so as to not rely on this unpublished observation, but use the simpler form for figures.

As illustrated in Fig. 13(b), we scale up the orthogonal planar embedding by a constant factor; replace each variable or clause vertex with a variable gadget or clause gadget respectively, and replace each y-monotone edge with a wire gadget (including at least one right shift and at least one left shift). Finally, we place the filler gadget in the regions between other gadgets, as shown in Fig. 13(c). Because each gadget has a unique solution corresponding to a Boolean assignment, this reduction is parsimonious.

Finally we analyze alignment issues in this construction. The shift gadgets of Fig. 9 provide exact horizontal positioning, so there are no constraints on horizontal positioning of gadgets. However, vertical positioning controls the parity of the false and true values of a wire, as in the yellow and green coloring of Fig. 6. We place all variable loops to have the same vertical positioning modulo 4. Thus we obtain consistent heights modulo 4 for the row of a wire between a false (yellow) and true (yellow) center that needs to be aligned with a clause: for wires extending upward, this is the row above a true center, while for wires extending downward, this is the row below a true center. Observe in Fig. 3 that these rows have the same height modulo 4 within a single variable loop, and that wire gadgets of Fig. 6 and shift gadgets of Figs. 7–9 preserve this height modulo 4. We can then place all clause gadgets at such y coordinates.

A solution to an $m \times n$ Spiral Galaxies puzzle can be verified in polynomial time. We conclude:

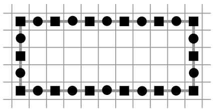


Fig. 14. We place centers (dots) on the middle point of potential edges (light gray) between any pair of boxes (squares). In the remainder of this section, we show only the boxes, not the centers (dots).

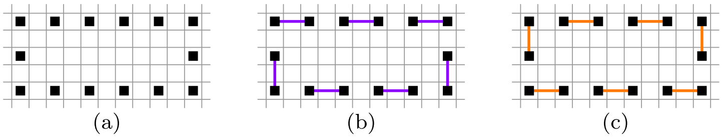


Fig. 15. (a) Variable gadget with two possible states (b) and (c) corresponding to a truth assignment of TRUE and FALSE of the corresponding variable.

Theorem 1. *Solving a Spiral Galaxies puzzle whose solutions have only rectangular galaxies is NP-complete and ASP-complete, and counting the number of solutions is #P-complete.*

3. Rectangular Galaxies with 1 × 1, 1 × 3, and 3 × 1 rectangles

In this section, we show that solving Rectangular Galaxies puzzles is NP-complete and ASP-complete, even with the promise that every solution uses only 1×1, 1×3, and 3×1 galaxies, and that the corresponding problem of counting the number of solutions is #P-complete. Again we give a reduction from PLANAR POSITIVE 1-IN-3 SAT. Given an instance F of PLANAR POSITIVE 1-IN-3 SAT with incidence graph G, we show how to turn a rectilinear planar embedding of G into a Rectangular Galaxies puzzle P such that a solution to P yields a solution to F, thereby showing NP-completeness. Furthermore, there will be a one-to-one correspondence between solutions of P and solutions of F, showing #P-completeness and ASP-completeness.

In this section, all galaxies will be 1×1, 1×3, or 3×1. To ease the description, our figures draw these galaxies as edges of length 2 (1×3 and 3×1) or nonexisting edges (1×1). Instead of drawing the galaxy centers (dots), we draw small boxes (squares) that reflect the endpoints of these edges. Boxes at distance 2 can be connected by a rectilinear edge, and centers will be located at the middle of each potential edge; see Fig. 14. Hence, any 1×3 or 3×1 galaxy will cover both boxes, denoted by an edge between these two boxes; any 1×1 galaxy will not extend over the boxes, and is shown by a nonexisting edge between the boxes.

Overview and filler gadget. At a high level, our reduction consists of two main gadgets: "variable" gadgets representing the variables of F; and "clause" gadgets to form the clauses of F.

As in Section 2, we fill any region not covered by these gadgets with a ***filler gadget***, which places a center at every cell of the region. Fig. 2 shows the filler gadget for a 2×2 region. The centers of a filler gadget must be contained in 1×1 galaxies, provided every uncovered region is the union of 2×2 squares.

Variable gadget. Fig. 15 illustrates the ***variable gadget***, which is an alternating pattern of boxes. While the figure shows the gadget in a simple rectangular form, variables can also turn arbitrarily, as long as each box has distance 2 to exactly two other boxes (i.e., the variable does not get too close to itself).

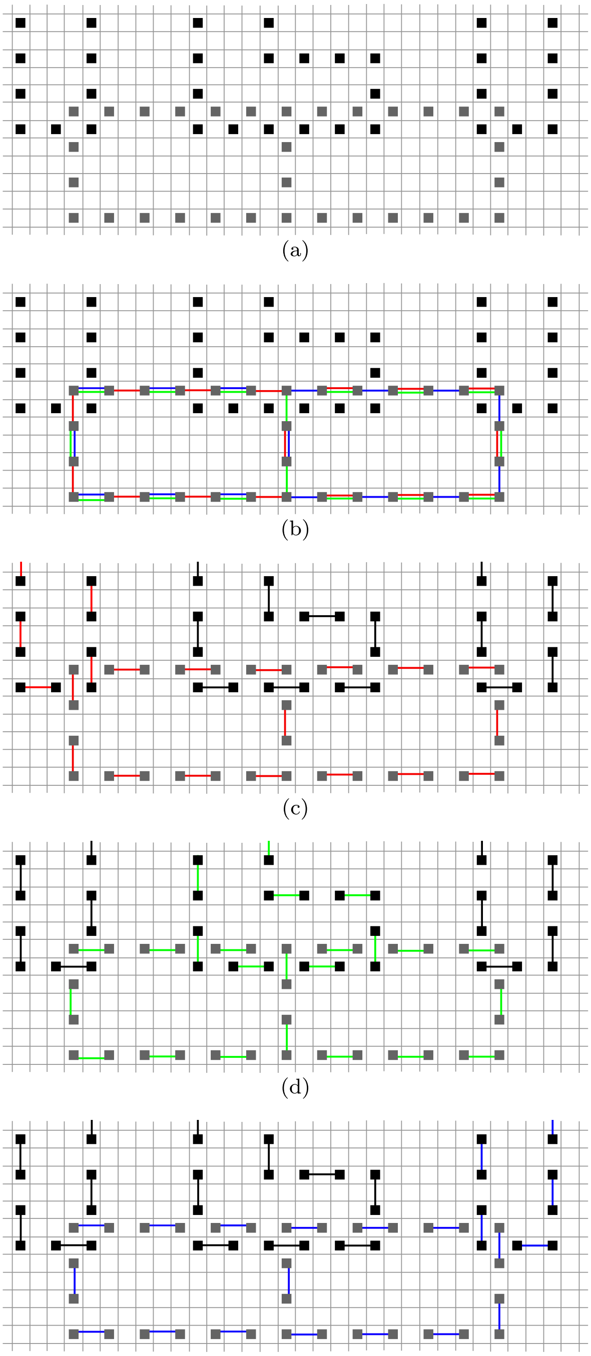
Lemma 1. *Each variable gadget has exactly two possible solutions (shown in Fig. 15(b) and (c)).*

Proof. Because of the filler gadgets, no galaxy centered at a center of the variable gadget can extend beyond the grid cells covered by the potential edges of the variable gadget. Hence, every rectangular galaxy must have a width or height of 1, which we label as "vertical" or "horizontal" respectively. Because the centers are placed with distance 2 and galaxies must be 180° rotationally symmetric about their center, horizontal and vertical galaxies have a width and height respectively in $\{1, 3\}$. Because galaxies may not overlap (we aim to decompose the grid), $1 \times 3/3 \times 1$ galaxies must alternate with 1×1 galaxies. There exist exactly two possible solutions that fulfill these conditions. □

Each of the two possible solutions corresponds to one truth assignment for the variable, TRUE and FALSE. We route each variable gadget to interact with every clause that contains the variable. Equivalently, we can imagine each variable gadget as having a "tentacle" to visit every incident clause.

Clause gadget. Fig. 16 shows the clause gadget, where the clause gadget itself is drawn in gray, and the three incident variable gadgets are drawn in black. There are three possible states of the clause gadget, drawn in red, green, and blue in Fig. 16(b). Each state forces exactly one of the variables' truth assignments to be TRUE and all the other variables to be FALSE, as shown in Fig. 16(c)–(e).

Putting pieces together. For the global construction, we start from a planar embedding of G with edges routed orthogonally on an $O(n) \times O(n)$ grid [17], scaled up by a constant factor. Then we locally replace each clause by a single clause


(a)

(b)

(c)

(d)

(e)

Fig. 16. (a) Clause gadget in gray, with the three incoming variable gadgets in black. (b) The three possible states of the clause gadget in red, green, and blue. (c)–(e) Each state of the clause gadget with the corresponding assignments of the variable gadgets; the (only) true variable is shown in the same color as the clause edges.

gadget, and replace each variable by a sufficiently large variable gadget that extends (via "tentacles") to all clauses in which it appears. That is, our variable gadgets are not small loops as shown in Fig. 15, but large loops that run through all the clause gadgets in which the corresponding variable appears. All variable gadgets use one parity (e.g., all boxes have even x and even y coordinates), while all clause gadgets use the other parity (e.g., all boxes have odd x and odd y coordinates). Hence, we can always place all gadgets and connect them together as desired. Because each gadget has a unique solution corresponding to a Boolean assignment, this reduction is parsimonious.

A solution to an $m \times n$ Rectangular Galaxies puzzle can be verified in polynomial time. We conclude:

Theorem 2. *Solving a Rectangular Galaxies puzzle whose solutions have only* 1×1, 1×3, *and* 3×1 *galaxies is NP-complete and ASP-complete, and counting the number of solutions is #P-complete.*

4. Noncrossing matching in distance-2 grid graphs

In this section, we prove an equivalence between Spiral Galaxies with 1×1, 1×3, and 3×1 galaxies and "noncrossing perfect matchings" in "distance-2 grid graphs". First we define the quoted terms. A ***distance-2 grid graph*** $G = (V, E)$ has vertices $V \subset \mathbb{Z} \times \mathbb{Z}$ at a finite subset of the integer lattice, and edges $E = \{(v, w) \mid v, w \in V, \|v - w\| = 2\}$ between all pairs of vertices at Euclidean distance exactly 2. A ***noncrossing perfect matching*** in a distance-2 grid graph $G = (V, E)$ is a partition of the vertices V into $|V|/2$ pairs such that the length-2 segments connecting paired vertices do not intersect (including at endpoints).

Next we observe that distance-2 grid graphs can always be decomposed into two independent components. Call a vertex $v = (x, y) \in V$ ***even*** if $x + y$ is even, and ***odd*** if $x + y$ is odd. Because edges connect vertices of distance exactly 2, which is even, vertices of different parity cannot be connected together. Thus even/odd forms a partition of any distance-2 grid graph. In the remainder, we restrict to ***even*** distance-2 grid graphs whose vertices are all even.

Call a Spiral Galaxies puzzle ***even*** if all centers are at cell centers and all *empty* cells are at even positions. We claim that there is a one-to-one correspondence between noncrossing perfect matchings in even distance-2 grid graphs and solutions to even Spiral Galaxies puzzles restricted to 1×1, 1×3, and 3×1 galaxies:

Lemma 2. *Given an even distance-2 grid graph, we can construct an even Spiral Galaxies puzzle restricted to* 1×1, 1×3, *and* 3×1 *galaxies with a one-to-one correspondence of the* $1 \times 3/3 \times 1$ *and the* 1×1 *galaxies and the matching and non-matching edges in noncrossing perfect matchings in the distance-2 grid graph, respectively, in polynomial time. Vice versa, we can reduce from a restricted even Spiral Galaxies puzzle to noncrossing perfect matching in an even distance-2 grid graph.*

Proof. Given an even distance-2 grid graph $G = (V, E)$, we construct an even Spiral Galaxies puzzle as follows. The board is the bounding box of the vertices V enlarged by $\frac{1}{2}$ on all sides, aligned so that every vertex of V is placed at the center of a board cell. For every *even* cell (x, y) of the board, we place a galaxy center at its center if and only if $(x, y) \notin V$; that is, the vertices of G correspond to empty cells of the board. For every *odd* cell (x, y) of the board, we always place a galaxy center at its center.

Similarly, from a given even Spiral Galaxies puzzle, we can construct an even distance-2 grid graph by creating a vertex for every empty board cell. (The edges are implied by the vertices by the definition of distance-2 grid graph.)

We claim that noncrossing perfect matchings of G correspond one-to-one with solutions to the Spiral Galaxies instance restricted to only 1×1, 1×3, and 3×1 galaxies. Indeed, all empty cells in the puzzle must be captured by a galaxy, and can be captured only in pairs on opposite sides of a galaxy center (in an odd cell), which in G corresponds to an edge connecting two vertices. □

Combining Lemma 2 and Theorem 2 (or equivalently, interpreting the boxes in figures as graph vertices), we obtain hardness for noncrossing perfect matching:

Corollary 1. *Noncrossing perfect matching in distance-2 grid graphs is NP-complete and ASP-complete, and counting the number of solutions is #P-complete.*

5. Minimizing centers in Spiral Galaxies for a given shape

In this section, we are given a shape $\mathcal{S}$ on a Spiral Galaxies board, and we aim to find the minimum number of centers such that there exist galaxies with these centers that exactly cover the given shape $\mathcal{S}$. We show that this problem is NP-complete by another reduction from Planar Positive 1-in-3 SAT.

Theorem 3. *It is NP-complete to minimize the number of centers on a Spiral Galaxies board such that galaxies with these centers exactly cover a given shape* $\mathcal{S}$.

For our reduction, the input is a Planar Positive 1-in-3 SAT formula, and the output is a "picture" — a subset of cells that should be "colored black". In the remainder of the section, we focus on the cells that are part of the picture; we will say that the cells which are not part of the picture are ***forbidden***.

First we will define some ***low-level gadgets***, which each have a constant size. Then we will build ***high-level gadgets***, which consist of multiple low-level gadgets, and which will be used to encode the 1-in-3-SAT formula.

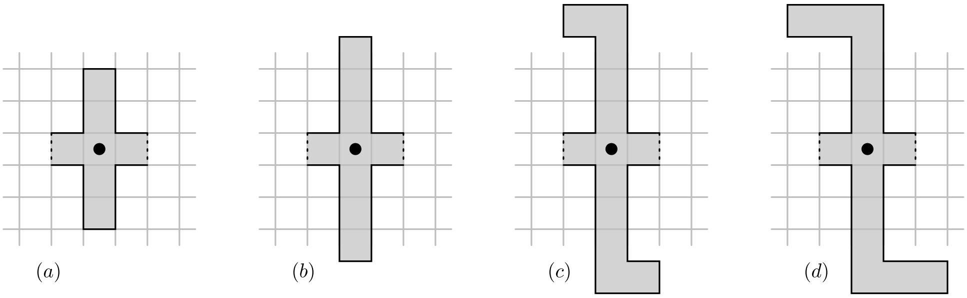


Fig. 17. The local center gadget must have at least one center. (a)–(d) Different shapes ensure we cannot include them in larger galaxies.

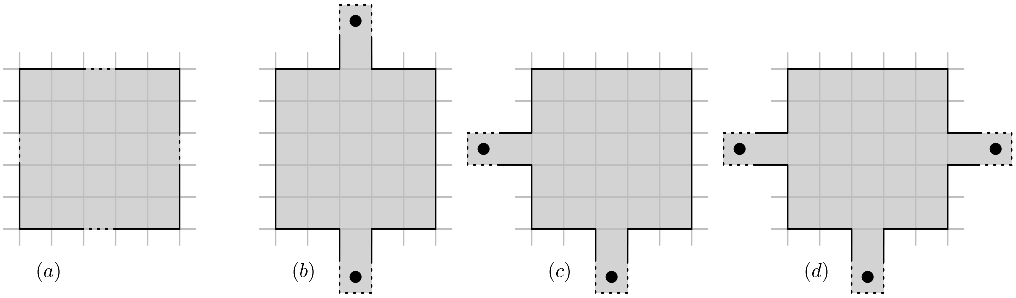


Fig. 18. (a) The block gadget. (b) A straight block. (c) A corner block. (d) A clause block.

5.1. Low-level gadgets

Local center gadgets. First we introduce ***local center gadgets***: thin constructions with unique shapes, which ensure that there must be at least one center in each of them; see Fig. 17. The idea will now be to construct a shape that can be covered with exactly this set of centers if and only if the 1-in-3-SAT instance is satisfiable.

Specifically, each local center gadget consists of a single cell with a ***desired galaxy center***. Its four direct neighbors are all empty, and its four diagonal neighbors are all forbidden. The cells left and right of the center will serve as connections to other low-level gadgets (we describe the gadgets as if they are vertically oriented; in the actual construction they may be rotated by 90°). The cells above and below the center are extended by symmetrically shaped paths of empty cells which are surrounded by forbidden cells.

We can immediately observe that, if there is *not* a center at the location of the desired galaxy center in a local center gadget, then either the entire gadget will be part of a larger galaxy with a center outside the gadget, or there must be at least two centers placed inside the gadget. In the first case, because of the symmetry of the galaxies, there must be a second copy of the local center gadget that is also part of the same galaxy. For this reason, we use not one but several different versions of the gadget. As we will see later, $O(1)$ different version will be sufficient.

Block gadgets. The second low-level gadget we use is the ***block gadget***. A block gadget is simply a 5×5 room surrounded by forbidden cells, except for possibly in the centers of its four sides. The idea is that a block gadget will not contain galaxy centers in an optimal solution, but rather will be connected to local center gadgets and be part of their galaxies. Depending on which of the sides of a block gadget are connected to other gadgets, we distinguish ***straight blocks***, ***corner blocks***, and ***clause blocks***; see Fig. 18.

Observation 1. Suppose we place local center gadgets and block gadgets in such a way that, for each block gadget B, all local center gadgets that B is connected to have different shapes. Then we must place at least one galaxy center inside each local center gadget.

End gadgets. Finally, our last low-level gadget is the ***end gadget***. An end gadget is similar to a bock gadget, but it is a 7×5 room and it has only one connection, at the center of one of its sides of length 5; see Fig. 19.

The idea is that, if we place an end gadget at the end of a chain that satisfies the conditions of Observation 1, then each end gadget will require at least one galaxy center to be placed inside it, because the center in the local center gadget next to it has only a 5×5 room on the other side (unless two end gadgets are directly connected, which is not useful). Depending on where we place it, more or less of the room is available to be included in neighboring galaxies.

Observation 2. If an end gadget is connected to a local center gadget, and on the other side of the local center gadget is not another end gadget, then we must place at least one galaxy center inside the end gadget.

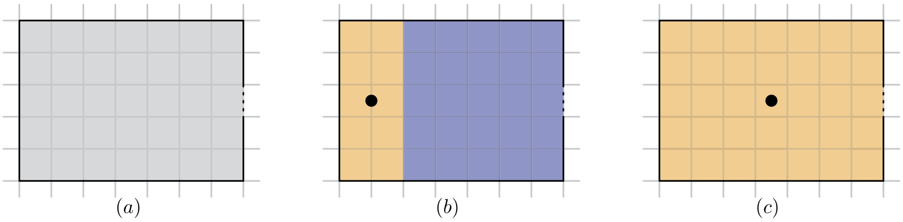


Fig. 19. An end gadget has one center, whether or not a block is used by the galaxy next to it.

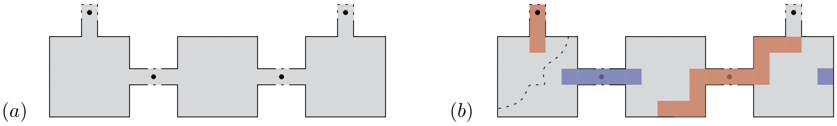


Fig. 20. (a) The fix gadget. (b) If the left block is split among multiple galaxies: contradiction.

5.2. High-level gadgets

With these low-level gadgets in place, we now turn to constructing high-level gadgets that will encode the variables and clauses of our 1-in-3-SAT instance.

Fix gadgets. First, we define the ***fix gadget***: an alternating sequence of four local center gadgets and three block gadgets making a U-turn as in Fig. 20. The purpose of this gadget is to ensure that, in an optimal solution, each 5×5 block will be completely included in a single galaxy, and not split among several galaxies.

Lemma 3. *In any chain of blocks that contains a fix gadget, an optimal solution requires that each block must completely belong to a single galaxy.*

Proof. First, by Observation 1, an optimal solution has exactly one center in each local center gadgets, and none in the block gadgets. This implies that if *any* block gadget in the chain does not completely belong to a single galaxy, then, by symmetry, *every* block gadget in the chain does not completely belong to a single galaxy.

Now, suppose we have a fix gadget consisting of blocks A, B, and C, where A and C are corner blocks but B is a straight block, and suppose A is split among multiple galaxies, say a red one at the top and a blue one at the right; refer to Fig. 20(b). Then, by symmetry of the blue galaxy, the center left pixel of B must also be blue and the bottom center pixel of B must also be red. Then, the top center pixel of C must be red (a priori, this could be the same or a different red galaxy — by connectivity it must be a new galaxy) and the center right pixel of C must be blue (again, either the same or a different blue galaxy) by symmetry of the (second) red galaxy. But when both the center left and top center pixels of C are red then C must be completely red, a contradiction. □

Variable gadgets. A ***variable chain*** consists of alternating local center gadgets and block gadgets starting and ending at end gadgets; see Fig. 21 for a variable chain. Each variable chain must include at least one fix gadget. Then, by Lemma 3, each block gadget on a variable chain must belong to exactly one of its two neighboring local center gadgets; this choice propagates throughout the entire chain and encodes the value TRUE or FALSE of one of the variables in our 1-in-3-SAT instance. Specifically, we show:

Lemma 4. *Consider a variable chain which*

- *has $k > 1$ local center gadgets and $k - 1$ block gadgets;*
- *contains at least one fix gadget; and*
- *has the property that every three consecutive local center gadgets have three different shapes.*

Then there exists no set of $k + 1$ or fewer galaxy centers that form a valid puzzle, and there are exactly two sets of $k + 2$ galaxy centers that form a valid puzzle, both of which include all k desired *galaxy centers.*

Proof. By Observation 2 and because $k > 1$, we must place one galaxy center inside each end gadget. By Observation 1 and because we vary the shapes of the local center gadgets, we must place at least one galaxy center at each of the k local center gadgets. Indeed, a galaxy centered in a local center gadget cannot include both adjacent local center gadgets, since they have different shapes, and a galaxy centered outside a local center gadget cannot include both incident local center gadgets, since they have different shapes. So, we need at least $k + 2$ centers. This is also sufficient, as shown by the

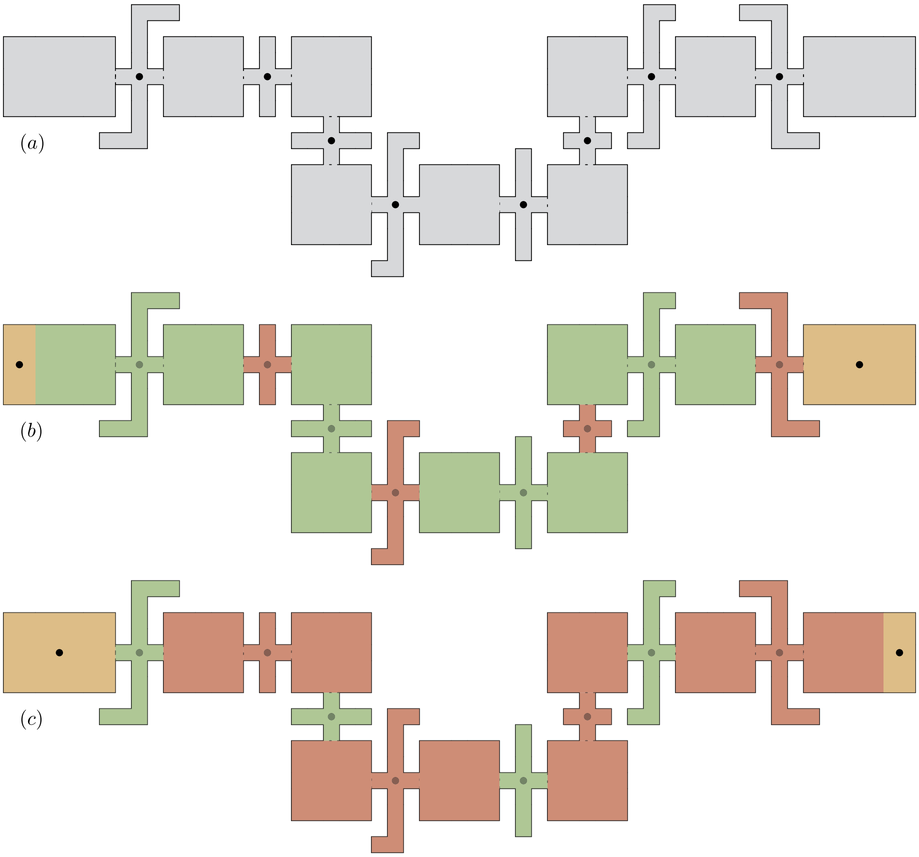


Fig. 21. A variable chain and its two possible truth assignments.

two solutions with $k+2$ centers in Fig. 21. Finally, by Lemma 3, we cannot have more than two possible shapes for each galaxy. □

Note that within one variable chain, we can realize the conditions of Lemma 4 with only three different local center gadget shapes, which we alternate cyclicly.

Split gadgets. In order to connect variables to multiple clauses, we need to be able to split them. Rather than splitting chains in the usual sense, our ***split gadget*** takes three independent variable chains and forces them to have the same state by combining three end gadgets into a single larger end gadget; see Fig. 22. This effectively creates a "split" chain with three remaining ends that have the same state; by repeating this we may create as many copies of a variable as required.

The idea behind the split gadget is simple: by combining three end gadgets into one large 7×15 room, we no longer know that we need three separate galaxy centers; in fact, applying Observation 2 to the three adjacent local center gadgets, we potentially only require to place one galaxy center instead of three. Observe that this is indeed achievable, but *only* when all three chains are in the same state, because otherwise the remaining shape of the room is not symmetric by 180° rotation (and thus cannot be covered by a single additional galaxy).

Observation 3. A split gadget requires only one galaxy center if and only if either all three adjacent local center gadgets do use a 5×5 area of its room, or if none of them do.

Clause gadgets. A clause of our 1-in-3-SAT formula is now simply encoded by a single clause block where three variable chains meet: it can be solved without using an additional center if and only if precisely one of the three chains needs to use the block.

In order to avoid that galaxies centered on clause blocks include local center gadgets, the three local center gadgets directly incident to the clause block must have distinct shapes. To achieve this, we may start with an edge coloring of the variable–clause incidence graph and use three unique shapes for each color. Every planar graph of maximal degree 3 admits an edge coloring with 4 colors, which results in 12 distinct shapes for our local center gadgets. It is possible, but not necessary, to reduce the number of shapes further.

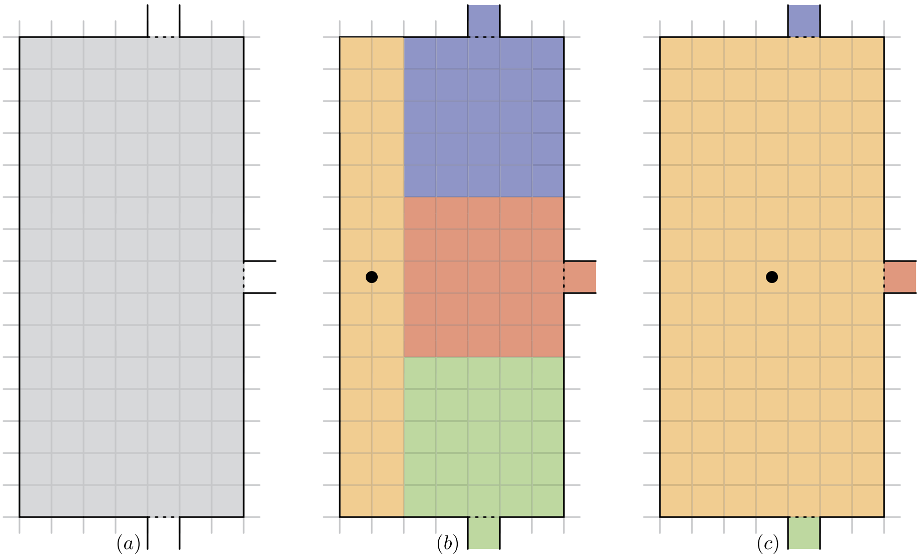


Fig. 22. The split gadget is essentially three end gadgets, but can only be solved with a single center if either all three blocks are used or all three are not used.

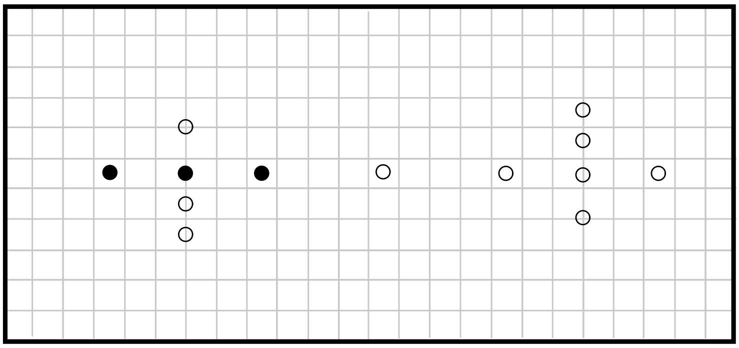


Fig. 23. Puzzle that can be solved for letters A, B, H, R, S, Z (E for disconnected galaxies).

5.3. Global construction

With our high-level gadgets now also in place, we can describe the global construction of our reduction. We start from a planar embedding of G with edges routed orthogonally on an $O(n) \times O(n)$ grid [17], scaled up by a constant factor. We locally replace each clause by a single clause block, each variable by a sufficient number of split gadgets connected by variable chains, and each edge by variable chains connecting variables to clauses.

Distance considerations. We must make sure that all our gadgets are aligned with the grid, and can be connected to each other. Furthermore, we need to ensure that the number of block gadgets in each variable chain has the same parity so that all clauses use positive forms of the variable.

We observe that we can achieve all of these requirements by simply adjusting the distance between adjacent block gadgets and allowing a slightly richer class of local center gadgets: instead of a width of 3 we may give them a width of any desired integer w. If w is even, we must place a galaxy center on an edge and not in the center of a cell, and we correspondingly also adjust the vertical arms of the gadget by giving them a width of 2 rather than 1.

This concludes the proof of Theorem 3.

6. Multiple solutions and the font puzzle

Standard Spiral Galaxies puzzles are designed to have a unique solution, with the intent that the puzzle solver finds that one solution. In this section, we briefly explore the possibility of creating a single puzzle with *multiple* solutions, such that each solution forms a different desired picture. As a case study, we ask for a single puzzle such that, for every letter of the alphabet, there exists a solution to the puzzle that resembles that letter. In Fig. 23, we provide a puzzle which has a

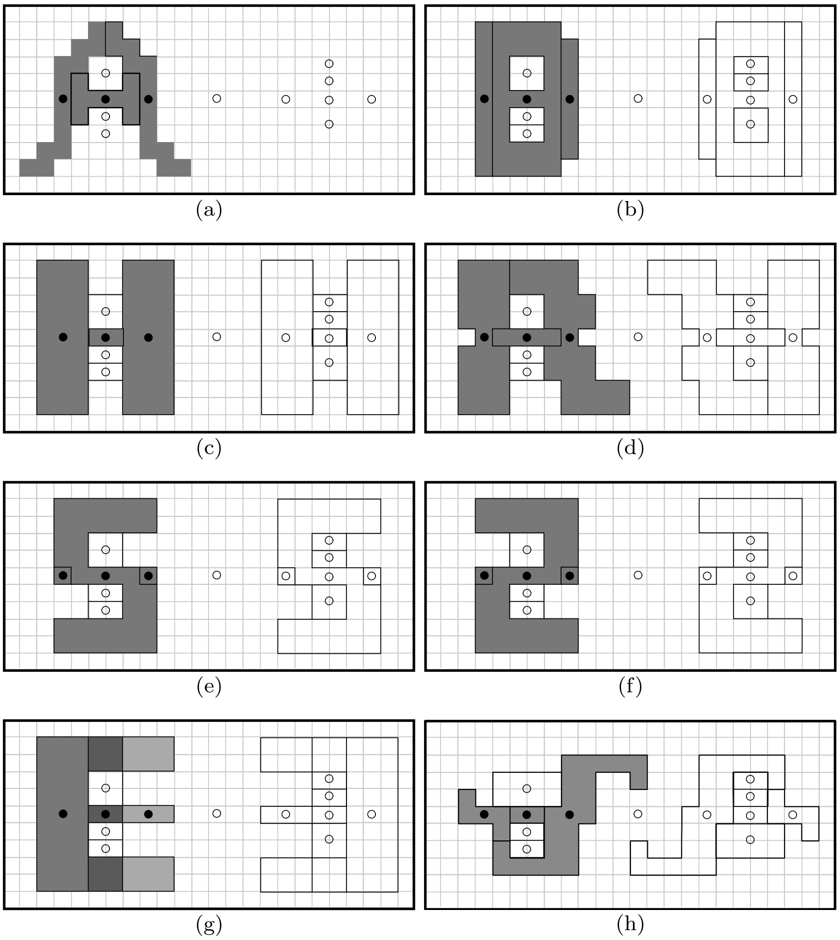


Fig. 24. (a–f) One puzzle with solutions drawing the letters A, B, H, R, S, and Z. (g) When we allow disconnected galaxies, we can also make the letter E. The three dark gray areas together form a single galaxy, and the three light gray areas as well. (h) The puzzle also has many "nonsense" solutions.

solution for the black letters A, B, H, P, R, S, Z (and when we allow for *disconnected* galaxies also for the letter E). See Fig. 24 for the solutions.

While our construction has solutions for several letters, for most letters of the alphabet it does not, and it also has many additional solutions which do not resemble letters at all. We leave as an open problem to construct puzzles which either solve for more letters, or for fewer nonletters, or both. More generally, this raises the question of whether it is possible to create puzzles with multiple solutions in such a way that *every* valid solution forms a meaningful picture.

7. Conclusion and discussion

We showed that both Rectangular Galaxies and rectangular-promise Spiral Galaxies are NP/ASP-complete even if the polyominoes are restricted to be rectangles of arbitrary size—and that counting the number of solutions is #P-complete. Moreover, we proved that Rectangular Galaxies remains NP/ASP/#P-complete even when only 1×1, 1×3, and 3×1 rectangles are possible. With the proof of the latter variant, we also showed that finding a noncrossing matching in distance-2 grid graphs is NP-complete. Finally, we proved NP-completeness of the design problem of finding the minimum number of centers such that there exist galaxies that exactly cover a given shape. The complexity of the latter problem when we aim for unique solutions remains open.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

We thank the anonymous referees for helpful comments. C. S. was partially supported by Jubileumsanslaget från Knut och Alice Wallenbergs Stiftelse.

References

[1] E. Demaine, M. Löffler, C. Schmidt, Rectangular spiral galaxies are still hard, in: 37th European Workshop on Computational Geometry (EuroCG 2021), 2021.

[2] Nikoli, Tentai Show, http://nikoli.co.jp/en/puzzles/astronomical_show.html.

[3] Grandmaster Puzzles, Spiral Galaxies rules and info, https://www.gmpuzzles.com/blog/spiral-galaxies-rules-info/. (Accessed 4 April 2018).

[4] E. Friedman, Spiral Galaxies puzzles are NP-complete, https://erich-friedman.github.io/papers/spiral.pdf, Mar. 2002.

[5] G. Fertin, S. Jamshidi, C. Komusiewicz, Towards an algorithmic guide to Spiral Galaxies, Theor. Comput. Sci. (2015) 26–39.

[6] V.V. Vazirani, Approximation Algorithms, Springer Publishing Company, Incorporated, 2010.

[7] T. Yato, Complexity and completeness of finding another solution and its application to puzzles, Master's thesis, University of Tokyo, 2003.

[8] T. Yato, T. Seta, Complexity and completeness of finding another solution and its application to puzzles, IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 86 (5) (2003) 1052–1060.

[9] N. Ueda, T. Nagao, NP-completeness results for NONOGRAM via parsimonious reductions, Tech. Rep. TR96-0008, Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan, May 1996.

[10] W. Anderson, E.D. Demaine, M.L. Demaine, Spiral Galaxies font, in: J. Beineke, J. Rosenhouse (Eds.), The Mathematics of Various Entertaining Subjects (MOVES 2017), vol. 3, Princeton University Press, 2019, pp. 24–30, see also, https://erikdemaine.org/fonts/spiralgalaxies/.

[11] M.E. Dyer, A.M. Frieze, Planar 3DM is NP-complete, J. Algorithms 7 (2) (1986) 174–184.

[12] W. Mulzer, G. Rote, Minimum-weight triangulation is NP-hard, J. ACM 55 (2) (2008) 11:1–11:29, https://doi.org/10.1145/1346330.1346336.

[13] H.B.H. III, M.V. Marathe, V. Radhakrishnan, R.E. Stearns, The complexity of planar counting problems, SIAM J. Comput. 27 (4) (1998) 1142–1167, https://doi.org/10.1137/S0097539793304601.

[14] W. Schnyder, Embedding planar graphs on the grid, in: Proceedings of the 1st Annual ACM-SIAM Symposium on Discrete Algorithms, San Francisco, California, USA, 1990, pp. 138–148.

[15] T.C. Biedl, Height-preserving transformations of planar graph drawings, in: C.A. Duncan, A. Symvonis (Eds.), Revised Selected Papers from the 22nd International Symposium on Graph Drawing, Würzburg, Germany, in: Lecture Notes in Computer Science, vol. 8871, 2014, pp. 380–391.

[16] E.D. Demaine, Lecture 7: Planar sat, Lecture video, MIT class 6.890: Algorithmic Lower Bounds: Fun with Hardness Proofs (Fall 2014), https://courses.csail.mit.edu/6.890/fall14/lectures/L07.html.

[17] T.C. Biedl, G. Kant, A better heuristic for orthogonal graph drawings, Comput. Geom. 9 (3) (1998) 159–180, https://doi.org/10.1016/S0925-7721(97)00026-6.