Hamiltonian Paths and Cycles in NP-Complete Puzzles

Marnix Deurloo
Utrecht University, The Netherlands

Mitchell Donkers
Utrecht University, The Netherlands

Mieke Maarse
Utrecht University, The Netherlands

Benjamin G. Rin[1]
Utrecht University, The Netherlands

Karen Schutte
Utrecht University, The Netherlands

Abstract

We show that several pen-and-paper puzzles are NP-complete by giving polynomial-time reductions from the Hamiltonian path and Hamiltonian cycle problems on grid graphs with maximum degree 3. The puzzles include *Dotchi Loop*, Chains, Linesweeper, *Arukone*$_3$ (also called Numberlink$_3$), and *Araf*. In addition, we show that this type of proof can still be used to prove the NP-completeness of *Dotchi Loop* even when the available puzzle instances are heavily restricted. Together, these results suggest that this approach holds promise in general for finding NP-completeness proofs of many pen-and-paper puzzles.

2012 ACM Subject Classification Theory of computation → Problems, reductions and completeness

Keywords and phrases Hamiltonicity, NP-completeness, complexity theory, pen-and-paper puzzles

Digital Object Identifier 10.4230/LIPIcs.FUN.2024.11

Acknowledgements We would like to thank Hein Duijf, Jonathan Grube, Rosalie Iemhoff, Dominik Klein, Michael Moortgat, and the anonymous referees for their helpful comments and suggestions.

1 Introduction

Pen-and-paper puzzles are often NP-complete. Famous among such results are the NP-completeness of Sudoku [26] and Minesweeper [17], not least because of the worldwide popularity of these particular puzzles. But innumerable other such results exist (see, e.g., [1], [2], [4], [7], [18], and [19], among countless others). Typically, to say a puzzle is NP-complete means that the problem of deciding whether a given instance of the puzzle is solvable is an NP-complete problem. Other problems regarding puzzles may also be considered (see, e.g., [24]), but the present paper addresses only solvability problems.

Often, the proofs of NP-completeness for pen-and-paper puzzles work by reduction from generally useful NP-complete problems such as 3SAT. However, some puzzles may resist this approach, or at least seem more naturally suited to be proved NP-complete by other reduction strategies. In particular, some puzzles take place on a grid and require the solver to construct a kind of path or loop. For puzzles in this category, it seems plausibly viable to prove their NP-completeness by reducing a variant of the Hamiltonian path or cycle problem

[1] Corresponding author. Authors listed alphabetically.

12th International Conference on Fun with Algorithms (FUN 2024).
Editors: Andrei Z. Broder and Tami Tamir; Article No. 11; pp. 11:1–11:25
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Figure 1 A grid graph (left) and a grid graph with specified terminal nodes s and t (right).

to them – particularly, a variant involving grid graphs. The present paper[2] considers five[3] puzzles, four of which (*Dotchi Loop,* Chains, Linesweeper, *Arukone*$_3$) directly require finding paths or cycles on square grids. The other, *Araf*, requires the solver to construct *regions,* but even there, our reduction shows how to force one such region to be shaped as a (Hamiltonian) path. In all cases, we find the Hamiltonian path or cycle problem on grid graphs with maximum degree 3 to be a successful basis for an NP-completeness proof. We surmise that this approach is broadly applicable to a wide array of other pen-and-paper puzzles.[4]

1.1 Preliminaries

▶ **Definition 1.** *A* grid graph *is a finite graph* $G = \langle V, E\rangle$ *whose nodes have integer coordinates, with edges between all and only pairs of nodes with Euclidean distance 1. That is,* $V \subseteq \mathbb{Z} \times \mathbb{Z}$ *and* $E = \{(\langle x, y\rangle, \langle x', y'\rangle) \in (\mathbb{Z} \times \mathbb{Z}) \times (\mathbb{Z} \times \mathbb{Z}) \mid |x - x'| + |y - y'| = 1\}$.

When discussing the (non-)existence of a path from a node s *(the* start *node) to* t *(the* end *node), we call* s *and* t terminal nodes*. Such a path is a* cycle *if* $s = t$ *and is* Hamiltonian *if it visits each node in* G *exactly once.*

▶ **Example 2.** In Figure 1 we see two grid graphs. The first has Hamiltonian paths, but no cycle. The second has a Hamiltonian path from specified node s to specified node t.

▶ **Definition 3.** *The problem* HC3G (the Hamiltonian cycle problem on grid graphs with max. degree 3) *is the problem of determining whether a given grid graph with max. degree 3 has a Hamiltonian cycle.*

The problem HP3G (the Hamiltonian path problem on grid graphs with max. degree 3) *is the problem of determining whether a given grid graph with max. degree 3 and specified nodes* s *and* t *has a Hamiltonian path from* s *to* t*.*

It is well known since [22, Thm. 2] that both of these problems are NP-complete:

▶ **Theorem 1** (Papadimitriou & Vazirani, see [22, Thm. 2])**.** *HP3G is NP-complete.*

---

[2] This work has its origins in four bachelor theses completed between 2019 and 2023 by authors of this paper under supervision of the fourth author (see [5], [6], [20], [23]).

[3] In Section 7, we also consider a restricted class of *Dotchi Loop* instances, for a total of six NP-completeness proofs.

[4] This proof strategy is not without precedent. For example, such an approach is taken in [3] to prove the complexity of Amazons puzzles, and [26] proves the ASP-completeness of Slitherlink. The present paper's earliest results, Theorems 3 and 4, were found by the third author in [20], albeit without restricting the graphs to maximum degree 3, making the proofs more complex than needed. Later, between the completion of the present paper and its appearance in print, we came to learn of some recent findings posted on ArXiv by Hadyn Tang for many loop and path puzzles [25]. These findings show that under certain conditions, puzzles can be proved NP-complete by constructing only gadgets for grid graph nodes with degree exactly 3, removing the need for gadgets of degree 1 or 2. Indeed, these conditions seem to obtain for at least some of the puzzles examined below, such as *Dotchi Loop* and Linesweeper. Accordingly, it is possible to simplify proofs wherever such gadgets are not needed. However, such gadgets are typically straightforward to construct anyway, and in any case we often find their inclusion instructive. It remains unclear how many puzzles, if any, are amenable to our approach but not Tang's.

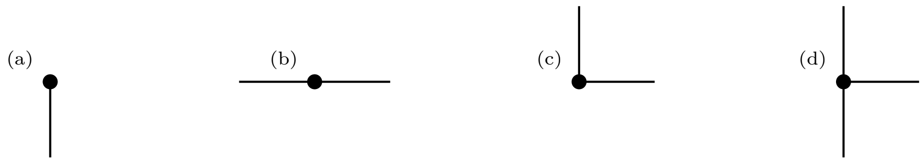

Figure 2 Four node shapes. (Rotations are also possible.)

▶ **Corollary 1.** *HC3G is NP-complete.*

While the corollary is not explicitly stated in [22], it can be proven by the same proof method as used for Theorem 1 but with one less step.[5] An explicit proof can also be found in [3].

Although the authors of [22] may not have predicted the future explosive popularity of pen-and-paper puzzles, nor the academic interest in their computational complexity, their result seems almost ready-made for NP-completeness proofs of many such puzzles – or, more precisely, NP-hardness proofs. Typically, proving that these puzzles are *in* NP is trivial, as verifying the correctness of a given solution attempt is easy to do in polynomial time. The present paper's results are no exception. Moreover, with respect to NP-hardness, the reader can verify that each reduction presented here is easily computable in polynomial time. Accordingly, we focus hereafter only on presenting the reductions. The early ones have relatively simple gadgets, but in subsequent proofs the intricacy increases.

1.2 General scheme

We call gadgets that represent nodes *rooms.* Rooms typically contain portions corresponding to edges, called *hallways* (or *corridors*). Rooms are usually square, or nearly so, and are surrounded on all sides by *walls* – arrangements of cells that prevent the simulated path/cycle from passing through them. Walls are typically a key – perhaps *the* key – subgadget for the success of the sorts of reductions presented in this paper. Their function is to force the simulated path/cycle to go only in the directions we want (i.e., to adjacent rooms) rather than wandering chaotically. On a side of a room with a hallway, the room's wall will naturally have a gap serving as an *exit* (or *entrance,* since the graph is undirected).

In the case of reductions from HP3G, rooms come in one of four distinct shapes (up to rotation), depending on their degree: a given graph can have one type of degree-1 node, two types of degree-2 node, and one type of degree-3 node (see figure 2). Additionally, a room's design may differ depending on whether the node it represents is the graph's starting node s, ending node t, or neither. However, since degree-1 nodes must always be terminal in any Hamiltonian path, we need not design gadgets for nonterminal degree-1 nodes.[6]

[5] The proof of Theorem 1 works by reduction from a known NP-complete variant of the Hamiltonian cycle problem on directed graphs, wherein one of the steps involves transforming nodes of a given directed graph into nodes of a grid graph with max. degree 3. Here, pains are taken to make the resulting grid graph suitable for checking the presence of a certain path (from specified nodes s and t) rather than a cycle (see [22, Fig. 2]). By dispensing with this effort and simply skipping this step (i.e., ignoring [22, Fig. 2b] and instead transforming all nodes of the directed graph in accordance with [22, Fig. 2a]), we obtain a proof of the NP-completeness of HC3G.

[6] Trivially, the reduction can easily map a graph containing such a node to any unsolvable puzzle instance. So we can assume the given graph contains no such nodes. On another note, for similar reasons, we can also assume the graph is connected.

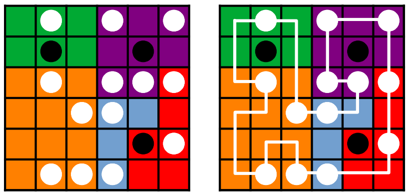

Figure 3 A *Dotchi Loop* puzzle instance (left) and its solution (right) [12]. Colors indicate regions.

In the case of reductions from HC3G, we can completely ignore *all* degree-1 nodes, since no graph with a degree-1 node can have a Hamiltonian cycle.[7] This leaves only three types of room to construct for such reductions, corresponding to (b)-(d) in Figure 2. That said, degree-1 rooms are usually easy to construct, so we can just as well include such gadgets in our proof and have our reduction transform graphs with degree-1 nodes into corresponding puzzle instances just like any other graph, with the understanding that such instances will necessarily be unsolvable.

2 *Dotchi Loop*

We begin with a proof for *Dotchi Loop*. The reduction is simple, but it straightforwardly demonstrates the use of HC3G for such proofs. In Section 7, we present a more complex proof showing *Dotchi Loop* to be NP-complete even when the set of possible puzzle instances is heavily restricted in a certain way.

Dotchi Loop [12] is a Japanese pen-and-paper puzzle consisting of a grid of cells divided into contiguous and non-overlapping *regions*. Some cells are empty and some contain a white or black circle. The solver must join cells orthogonally to create a single closed loop that passes through all white circles and avoids all black circles. (See Figure 3.) It is not necessary to visit every empty cell. The loop is forbidden to cross or overlap itself. Within any given region, the loop must either turn 90° at all white circles or pass straight through all white circles of that region. For brevity, we call any White circle through which the loop passes straight a *straight circle*. We call the others *turn circles*. For example, the white circle in the northwest region is straight and the ones in the southwest region are turn circles. A region is *straight* (respectively, *turn*) if its white circles are straight (respectively, turn).

▶ **Theorem 2.** Dotchi Loop *is NP-complete.*

Proof. We give a reduction from HC3G. Wall subgadgets are easy to construct by using rows of black circles. With these, we can straightforwardly build rooms with appropriate hallways. Each fits in a 3×3 area and contains a white circle at the center. We fill the remaining space in each room with black circles, except for pathways toward the central cell from every available entrance. See Figure 4. Each room is its own region. As with all these reductions, rooms are placed in similar relative positions to those of the corresponding nodes and edges in the input graph. An example can be seen in Figure 5.

To confirm the correctness of the reduction, observe that white circles in each room guarantee, per the rules, that a solution's loop will necessarily visit that room. So the similarly shaped cycle in the given grid graph is Hamiltonian. Conversely, if the grid graph is Hamiltonian, a similarly shaped loop can clearly cover all white cells. This completes the proof. ◀

[7] Again, such a graph can trivially be mapped to any unsolvable instance of the puzzle at hand. And, again, we can do the same with any graph that isn't connected.

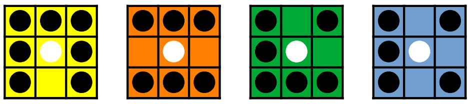

Figure 4 All *Dotchi Loop* rooms (up to rotation). The first is unnecessary (see Section 1.2).

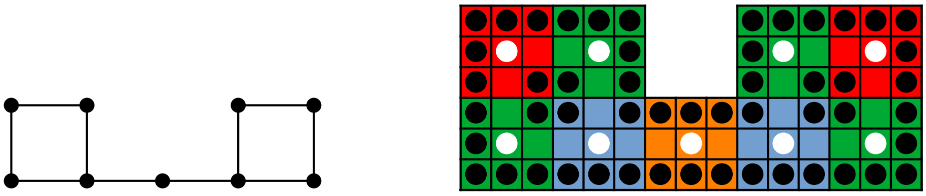

Figure 5 A graph (left) and corresponding *Dotchi Loop* puzzle instance (right). Colors distinguish neighboring regions (though distinct regions with no shared border may share colors in the figure).

▶ **Example 4.** The graph in Figure 5 has nine nodes. The corresponding puzzle instance has nine rooms. Each room here is a separate *Dotchi Loop* region.

3 Chains

A Chains [13] puzzle instance consists of a grid of cells, some of which contain natural numbers. The solver must join cells orthogonally such that each cell numbered $i > 1$ is connected by a "line" (path) of length i to some other cell also containing i. Cells with number $i = 1$ must be in lines of length 1 (i.e., to themselves). Lines cannot overlap themselves or other paths. See Figure 6.

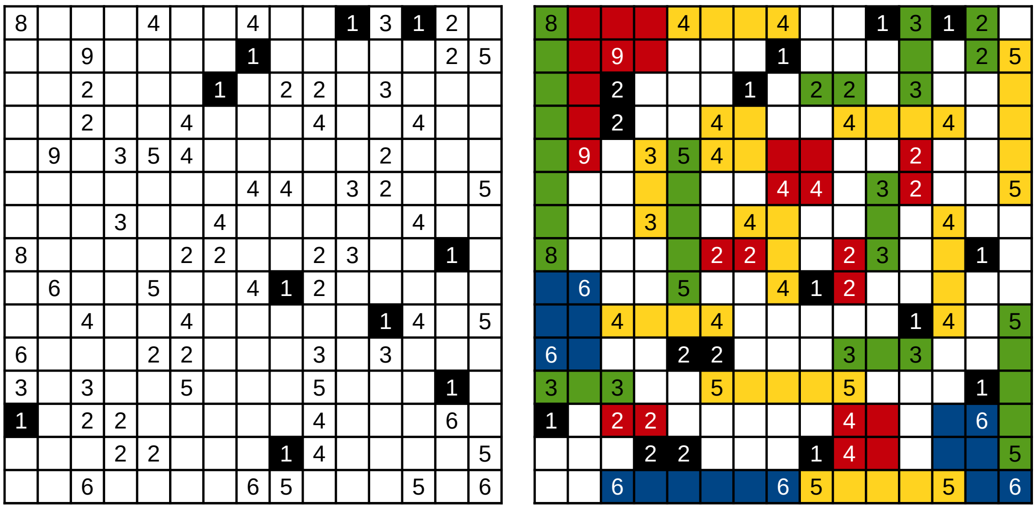

Figure 6 A Chains puzzle instance (left) and its solution (right) [14].

▶ **Theorem 3.** *Chains is NP-complete.*

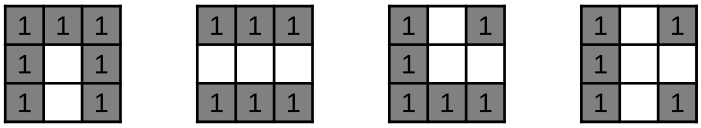

Figure 7 Chains nonterminal room gadgets. (Rotations are also possible.)

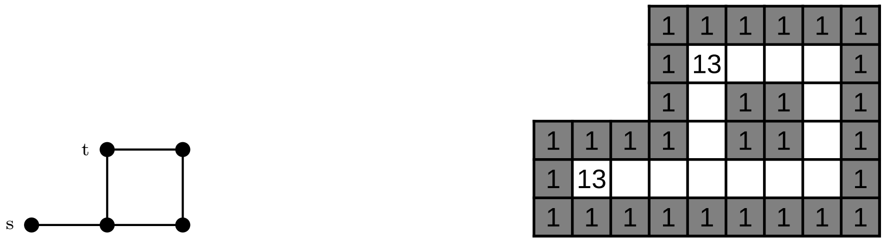

Figure 8 A graph (left) and equivalent Chains puzzle instance (right).

Proof. We give a reduction from HP3G. Since lines may not cross, walls are simply constructed out of cells containing 1s. Room gadgets consist of 3×3 blocks of cells. The gadgets for each nonterminal node type are presented in Figure 7. As usual, rooms are placed in relative positions corresponding to those of the given graph's nodes.

The rooms representing terminal nodes are constructed identically to the nonterminal nodes, except that we add the number $3m - 2$ to their central cells, where m is the number of nodes in the graph. See Figure 8 for an example.

To verify the reduction's correctness, first observe that if a Hamiltonian path exists between s and t, then the two corresponding central cells can indeed connect via an identically shaped line of length $3(m-1)+1 = 3m-2$, since (i) the number of cells needed to move from one room's central cell to the next is always three, (ii) the path from s to t uses $m-1$ edges, so the similarly shaped line in the puzzle connects $m-1$ pairs of central cells from adjacent rooms, and (iii) cell s itself must be counted in the total length.

Conversely, the puzzle instance is constructed in such a way that there is virtually no flexibility for the solver. From the center of any room, any movement of three steps leads to the center of another room. Therefore, a line connecting the two cells numbered $3m-2$ (which must have length $3m-2$, by the rules) visits exactly m rooms, making the similarly shaped path from s to t in the graph Hamiltonian. ◀

▶ **Example 5.** The graph in Figure 8 has five nodes, so $3m - 2 = 3(5) - 2 = 13$.

4 Linesweeper

Linesweeper [15], like Chains, consists of a grid of cells, some of which contain nonnegative[8] integers. The solver must find a loop through the grid such that every numbered cell is orthogonally or diagonally adjacent to precisely that number of cells visited by the loop. An example is shown in Figure 9.

[8] Note that typical instances of Linesweeper have no cells with 0s, but the puzzle's definition does explicitly allow them – see [21].

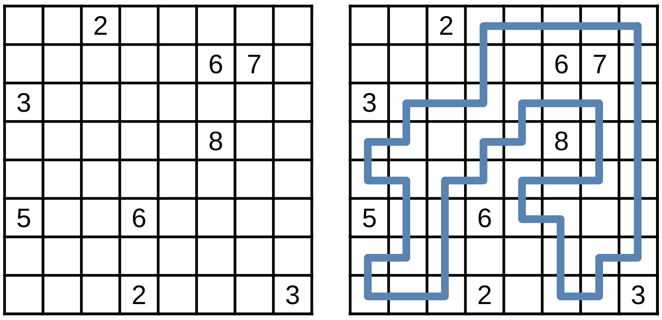

Figure 9 A Linesweeper puzzle instance (left) and its solution (right) [16].

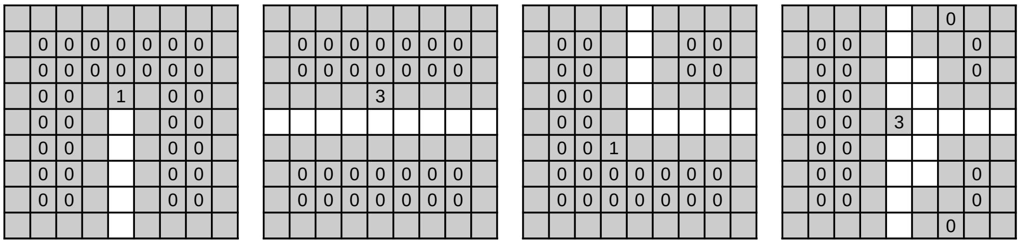

Figure 10 All Linesweeper rooms (up to rotation). Cells clearly inaccessible to the solution loop are shaded as a visual aid. Note that we could also use only the last room and add 0s to block exits.

▶ **Theorem 4.** *Linesweeper is NP-complete.*

Proof. We give a reduction from HC3G. Walls are created by use of 0s. Under the rules, the loop cannot touch any cells adjacent to a cell containing a 0, so each 0-cell creates a 3×3 impenetrable area. Room gadgets are shaped rather similarly to those of Chains and *Dotchi Loop*, though larger to accommodate the peculiar needs of the degree-3 room. In or adjacent to the center of each room is a cell with the number 1 or 3, depending on the room type. See Figure 10.

It is straightforward to see that the gadgets other than the degree-3 room are solvable[9] in exactly one way. The degree-3 room is solvable in three ways, displayed in Figure 11. A full example of the reduction is shown in Figure 12.

To verify the reduction's correctness, we first note that if a given graph has a Hamiltonian cycle, the corresponding Linesweeper instance is clearly solvable by a loop of similar shape. For the converse, suppose the puzzle instance is solvable. Observe that the 1 or 3 in each room guarantees that the solution loop must visit that room. As noted above, every room without degree 3 can be traversed by the loop in only one way, and so it is easy to see that our needed cycle in the graph takes precisely the same shape traversing the corresponding node. For the degree-3 rooms, the reader can verify that the three solutions presented

[9] We acknowledge that only a puzzle instance as a whole, not a subset (such as a single room) can be formally said to be solved or have a solution. However, as a mild abuse of terminology, we may say throughout this paper that an indication (usually pictorial, but possibly written) of things for a puzzle solver to do with respect to a given room is a *solution* to the room if the indication would describe part of a correct solution attempt for an entire puzzle instance containing that room. We will say the room is *solved* if a solution (in this sense) has been given for it.

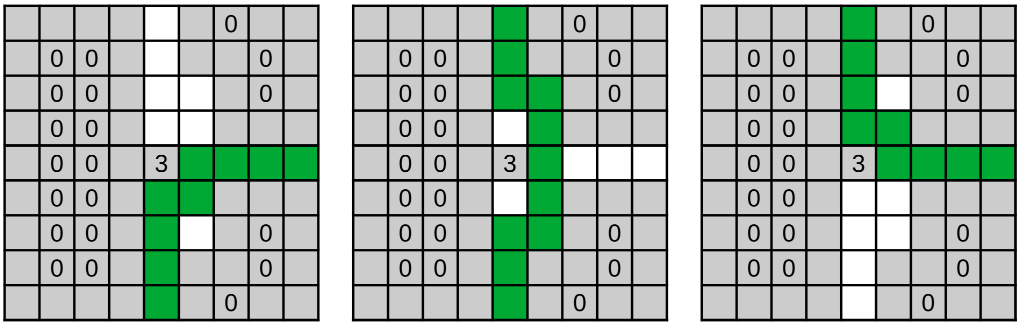

Figure 11 Three solutions for a degree-3 Linesweeper room.

in Figure 11 are the only possible ways to solve such rooms. Consequently, we see that whichever direction the solution loop takes through a degree-3 room, a Hamiltonian cycle through the input graph can take the same direction. Thus a given grid graph with max. degree 3 is Hamiltonian if and only if the corresponding Linesweeper instance is solvable, as required. ◀

▶ **Example 6.** In Figure 12 we see a full example of the reduction.

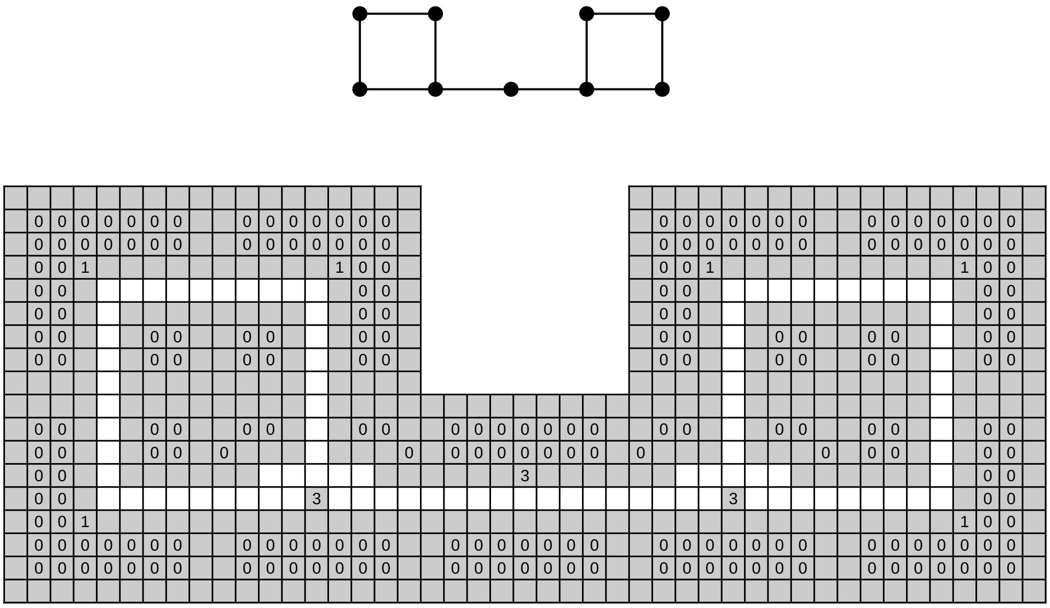

Figure 12 A graph and its corresponding Linesweeper puzzle instance.

5 *Arukone*$_3$

Arukone$_3$ (written *Arukone*^{3} in [11]) is another puzzle consisting of a grid of blank and natural-numbered cells, with the additional stipulation that each number that is present appears exactly twice. This puzzle is a variant of *Arukone* (also called Number Link, Nanbarinku, or

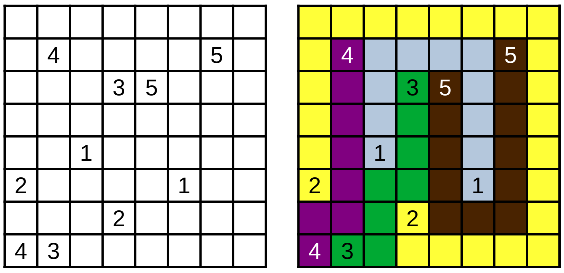

Figure 13 An *Arukone*$_3$ puzzle instance (left) and its solution (right) [11].

Flow – see [9]) and *Arukone*$_2$ (see [10]), which have already been considered.[10] The *Arukone*$_3$ variant contains just one added rule beyond *Arukone*$_2$ (the 2×2 rule – see below), but this rule markedly affects the nature of the puzzle.

The goal is to connect each pair of equal numbers with a path (called a *line*) of orthogonally adjacent cells that never crosses itself or another line and never covers a 2×2 area. The puzzle is solved when all number pairs are connected and all empty cells are filled. Figure 13 shows an example.

▶ **Theorem 5.** Arukone$_3$ *is NP-complete.*

Proof. Our reduction for this puzzle is from HP3G. Accordingly, the grid graph should have two distinguished terminal nodes, s and t. Recall that this makes reductions from HP3G require the construction of more room types than in reductions from HC3G. In the case of *Arukone*$_3$, the two rooms corresponding to nodes s and t will share a number S in their central cells. We call the line connecting the two S-cells (assuming the puzzle instance is solvable) the *Hamilton line.* In this proof, there is no relevant distinction to make between a starting room and ending room, so we have only one gadget design for, say, the degree-3 terminal room. (In the proof of Theorem 6 for *Araf,* such a distinction is necessary and more room types will therefore be needed.)

In our reduction to *Arukone*$_3$, the "walls" will consist of many number pairs arranged as dominoes (see, e.g., Figure 14). Since the numbers in a given domino can only be connected to each other, and since these dominoes are so tightly packed together, the only connection possible for them is a line of length 2. In general, we say that two equally numbered cells are connected *directly* if the line connecting them has length 2, and *indirectly* otherwise. In our diagrams, wall cells are shaded for readability. Additionally, rather than display a unique number pair for each domino within walls, we simply draw a ring (see, e.g., the bottom-right horizontal dominoes in Figure 14) to indicate that this pair of adjacent cells must be directly connected in any solution.[11] However, not all dominoes will be displayed this way; some (non-wall) dominoes that may be possible to connect indirectly will be shown with letters standing for natural numbers – in the present figure, a, b, c, and d. (As discussed above, the central S is also a natural number.) It is easy to see that the walls constructed here serve their intended purpose sufficiently: since lines cannot intersect, the mass of directly connected dominoes prevents another line (in particular, the Hamilton line) from getting through.

[10] See [1]. Also notable is [19], which shows the NP-completeness of a further *Arukone* variant.

[11] For most of these dominoes, the need for direct connection is obvious because they have no empty space around them. For some – e.g., the vertical dominoes on the left of Figure 14 – direct connection is forced because any indirect connection would violate the 2×2 rule.

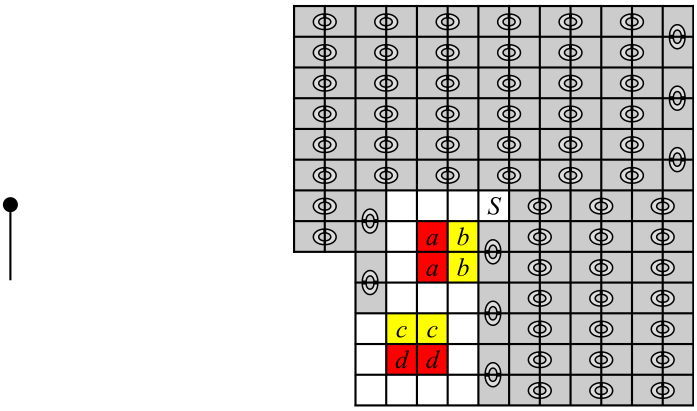

Figure 14 A degree-1 node and *Arukone*$_3$ degree-1 room. Colors are only a visual aid.

Rooms in this reduction are 13×13 square areas, minus ten cells for each corridor out. These cells are occupied by the adjacent room. The room in Figure 14 has one corridor, which consists of all the empty cells in the southwest quadrant forming a rough δ-shape. The exit is west of the cc domino. Later it will become clearer how rooms in this reduction fit together.

One oddity in this reduction, in contrast to the rest in this paper, is that rooms in this reduction are tilted 45° clockwise from the nodes they represent. For example, Figure 14 depicts a degree-1 room in which the outgoing corridor heads southwest, representing a degree-1 node with an outgoing edge to the *south.*

The aforementioned non-wall dominoes in Figure 14 with numbers a, b, c, and d form what we call *pillars.* The two dominoes aa and bb together comprise one pillar and the dominoes cc and dd comprise another. In general, a pillar diagonally touching the central cell (here, the a-b pillar) is called an *inner pillar* and a pillar more distant (here, c-d) is called an *outer pillar.* In the present gadget, all four of these dominoes must be connected directly (the degree-1 room bears only one possible solution, wherein pillar dominoes must connect directly to avoid blocking the line from S). However, in other gadgets this will not always hold. As we will see, the purpose of pillars, in rooms with more than one possible solution, is to be able to connect indirectly and fill the cells of corridors not in use by the Hamilton line. Since the rules of *Arukone*$_3$ require that all cells be filled, pillars are a key part of the construction that make the reduction work. We will see, furthermore, that when a pillar domino needs to connect indirectly to fill unused corridor cells, it can do so without breaking the 2×2 rule.

Figure 15(a) depicts a nonterminal straight degree-2 room with northwest and southeast corridors, corresponding to a nonterminal straight degree-2 node with west and east edges. Figure 15(b) depicts a nonterminal corner degree-2 room with northeast and southeast corridors, corresponding to a nonterminal corner degree-2 node with north and east edges.

The straight room, we claim, is solvable in exactly one way, with no possibility for the pillars to have indirect connections – similarly to the degree-1 room. In this unique solution, the empty cells must be all filled by a single line passing completely through the room. (It will turn out that this line is the Hamilton line.) To verify this, observe by inspection that the central cell cannot be reached by any possible line that would indirectly connect a domino within this room. So the central cell must be in some line extending outside the room. Since the central cell, like any cell without a number, can never be the endpoint its line, its line

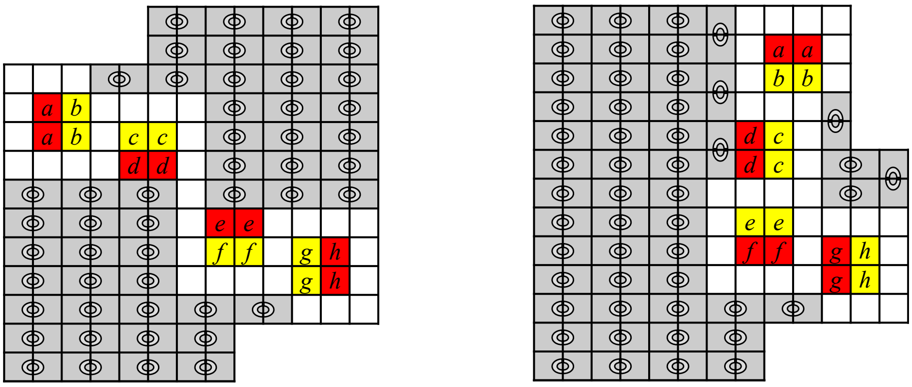

Figure 15 Nonterminal degree-2 straight and corner *Arukone*$_3$ rooms.

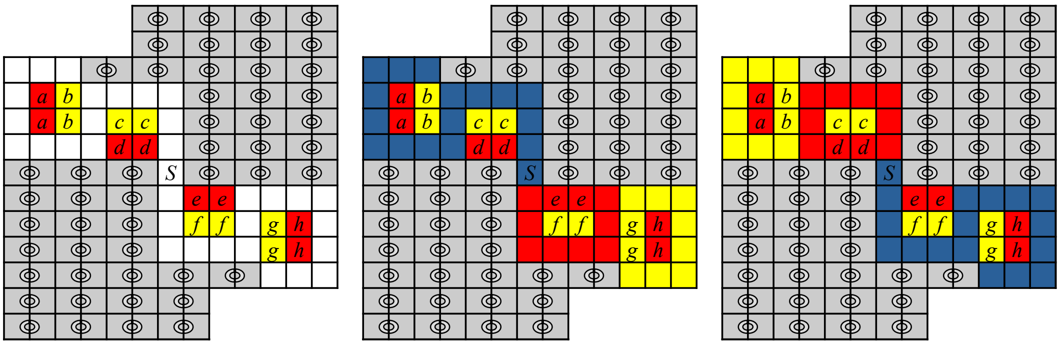

Figure 16 An *Arukone*$_3$ terminal degree-2 straight room and its two solutions.

must extend out through both exits (in the process, filling both of the two corridors entirely). In sum, this room is solved only by running a line from one exit to the other, passing through the center cell along the way. This behavior matches that of nonterminal straight degree-2 nodes in solved HP3G instances.

The corner room is similar. Its solution requires a single line containing the center, both exits, and (consequently) all other white cells. The only difference in the argument is in the reason why the center is inaccessible to pillar connection lines. For the straight room, it was because of physical inaccessibility, but in the corner room, the 2×2 rule plays a role in preventing an inner pillar domino's line from filling the center. In fact, we can see by inspection that all pillar domino lines in the corner room must connect directly. As with the straight room, the unique possible solution for the present gadget matches that of the corresponding node type in solved HP3G instances.

In contrast to these two nonterminal degree-2 rooms, the terminal degree-2 rooms can (unsurprisingly) be solved two ways, neither of which can use both corridors (because the cell with number S cannot be a middle point of its line). The two possible solutions, shown in Figure 16, correspond precisely to the two possible behaviors of terminal straight degree-2 nodes in solved HP3G instances. Here the use of pillars to fill otherwise empty corridor space exemplifies our earlier description.

Figure 17 shows the terminal corner degree-2 room. We claim this room is solvable in two ways, much like the terminal room already discussed. Note that in Figure 17 there are two possible cells for the Hamilton line to visit in its first step from cell S. If it visits the cell

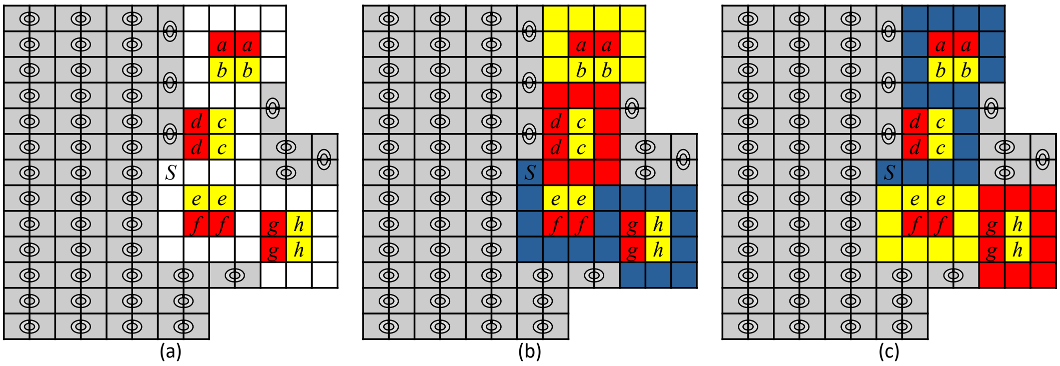

Figure 17 An $Arukone_3$ terminal degree-2 corner room and its two solutions.

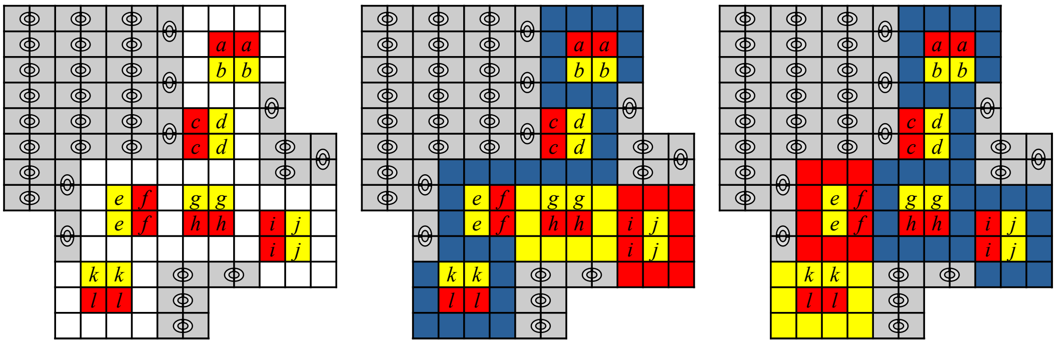

Figure 18 An $Arukone_3$ nonterminal degree-3 room and two of three solutions.

south of S, then it cannot also visit any cells north of the center row, as that would leave behind white cells that pillar dominoes could never fill without breaking the 2×2 rule. By a similar argument, if the line originating from S heads east, then it cannot visit any cells *south* of the center row. In either case, the cells not filled by the Hamilton line can and must be filled by the pillars, as seen in the figure.

Figure 18 displays a nonterminal degree-3 room. We claim there are precisely three possible solutions (two, up to symmetry), in concordance with nonterminal degree-3 nodes in solved HP3G instances. First, note that the central cell must, as before, be part of a line that traverses precisely two of the exits. If they are the northeast and southwest exits, then the solution must be as in Figure 18(b) (other attempts – for instance, using the three white cells in the central column – fail for reasons similar to those previously discussed). Likewise, if they are the northeast and southeast, the solution must be as in the third figure. The last case (southwest and southeast) is symmetrical to this.

Finally, Figure 19 displays a terminal degree-3 room. As noted for the other terminal rooms, the cell numbered S must be an endpoint of its line, so its line must visit precisely one of the three cells adjacent to it. (The other two must be filled by inner pillars.) If it moves east, then the other two cells neighboring S must be filled by inner pillars (see Figure 19(b)), after which there is no option but to take the northeast exit (Figure 19(c)). The other two cases – moving one cell south or west (not shown) – are similar (forcing east and south exits, respectively). Again, in this gadget we see all, and only, the possible behaviors corresponding to those of terminal degree-3 nodes in HP3G instances.

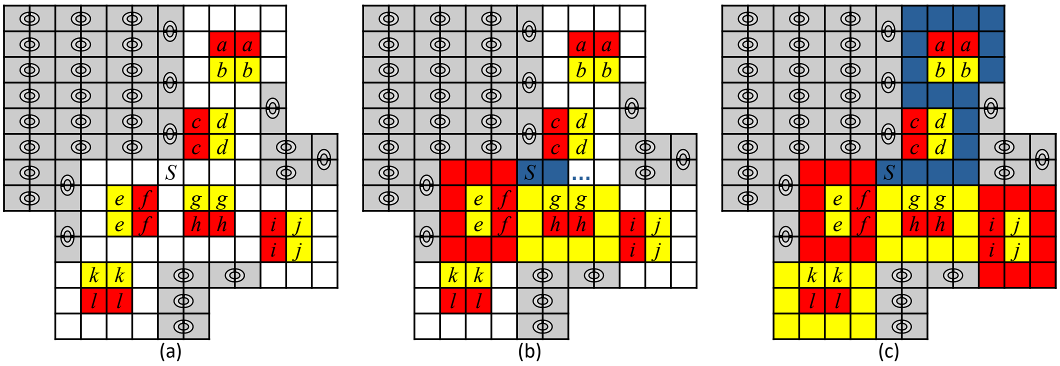

Figure 19 *Arukone*$_3$ terminal degree-3 room (a), a partial solution (b), and a full solution (c).

This completes the discussion of individual gadgets. An example puzzle instance showing how they fit together is given in Figure 20. Extra walls can be added to make the instance rectangular. This completes the proof. ◀

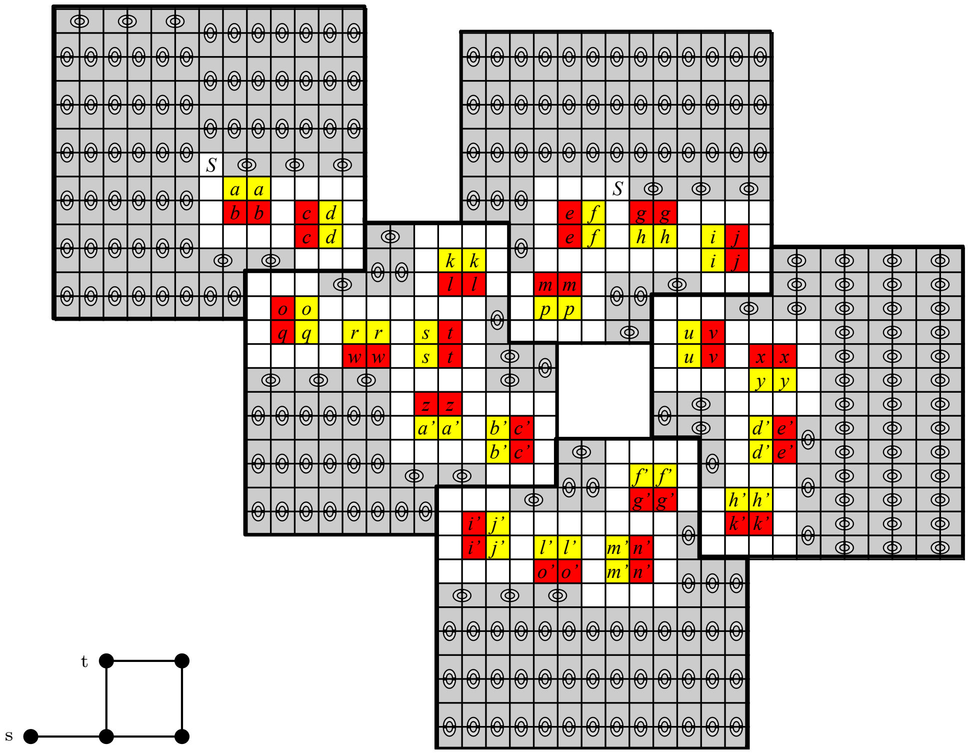

Figure 20 A graph and its corresponding *Arukone*$_3$ instance. (Further walls in white space not shown.) The instance has a 45° clockwise tilt compared to the graph.

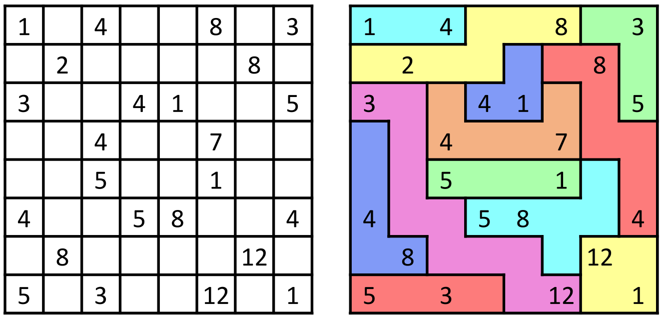

Figure 21 An *Araf* puzzle instance (left) and its solution (right). [8].

6 Araf

Araf, like Linesweeper and Chains, consists of a grid of cells that each may contain a natural number. (Unlike *Arukone*$_3$, *Araf* need not have precisely two of each present number.) The goal is to divide the grid into contiguous regions such that (i) each cell is part of exactly one region, (ii) each region contains exactly two numbers, and (iii) the size of each region's area is strictly between those two numbers. Figure 21 presents an example.

▶ **Theorem 6.** Araf *is NP-complete.*

Proof. We give a reduction from HP3G. In this reduction, it will be necessary to construct eleven rooms, not seven as in *Arukone*$_3$. This is because we have separate designs for rooms representing start nodes and end nodes, rather than a single design for both types of terminal node for any given node degree/shape. We will begin with the rooms for start nodes, then present the rooms for nonterminal nodes, and then finally present the rooms for end nodes.

Let us say two cells are *connected* in a solution if they occupy the same region. We may at times identify a numbered cell with its number, if no confusion arises. Thus we may say, e.g., that the numbers a and b in a given room are connected.

In all rooms we construct, the walls are built out of dominoes that each consist of a 1 and a 3 (see Figure 22). By rule (iii), the two domino cells must be together in a region of area exactly 2, as long as no nearby numbered cells provide other options. We construct the rooms in such a way that no problem with this arises. Technically, while it is conceivable that a large area filled with wall dominoes could be solved in more than one way (e.g., if a horizontal 3-1 domino is above a horizontal 1-3 domino, these four cells can be paired off vertically or horizontally), these differences have no meaningful effect on the puzzle solution or our proof. In particular, the region corresponding to the grid graph's Hamiltonian path (hereafter, the *Hamilton region*) is unaffected. Accordingly, we will not distinguish between solutions with identical Hamilton regions that differ only by small variations such as with the 3-1 pairings.

Importantly, all the rooms in this proof are constructed in such a way that they can't affect each other. The sides of each room are filled with 1s and there is a 3 in each corner. Thus the cells on the side of a room will be unable to share a region with cells on the side of an adjacent room. Therefore, it will be impossible for a solution to have any region include cells from two or more gadgets, with the exception of the Hamilton region.

If the grid graph has a Hamiltonian path, the size of the Hamilton region in the puzzle is straightforward to calculate. As we will see, the start room will have nine cells available for the Hamilton region, the end room will have seven, and the nonterminal rooms will each have thirteen. Hence the region has $9 + 7 + 13(n - 2) = 13n - 10$ cells, where n is the number of nodes in the grid graph.

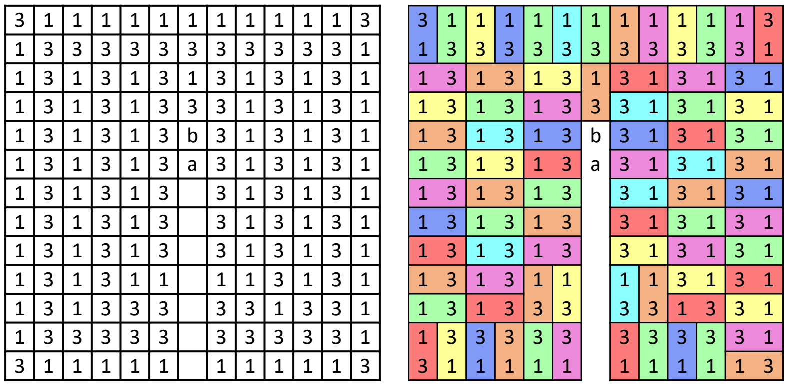

Figure 22 An *Araf* degree-1 start room (left) and solution.

We now define $a = 13n - 11$ and $b = a + 2 = 13n - 9$. The room representing the start node will contain a domino with a and b, arranged so that these two numbers must connect in any solution. Their region, with forced size $13n - 10$, will be the Hamilton region.

Note that the rest of the numbers in the puzzle will all be 9 or less, so the numbers a and b will be by far the greatest.

In Figure 22 the degree-1 start room is shown. There are eighty 3-1 regions and two (large) numbers a and b, which we now argue must share a region (the Hamilton region). Observe that b cannot connect to one of the 3s next to it – for, if it did, its region would need to have size at least 4, which is not possible in the given room. So b must connect to a. A similar observation will hold for the other start rooms.

Since a and b are connected, the 3s and 1s in this gadget must connect to each other. Since each such connection yields a region of size 2 under *Araf's* rules, there must be $80 \cdot 2 = 160$ cells filled. There are $13 \times 13 = 169$ total cells in this gadget, so the nine remaining cells, including a and b, must be occupied by the Hamilton region.

Figure 23 shows the straight degree-2 start room. We argue that a solution for it must contain seventy-four 3-1 regions, one 9-7 region, one 9-1 region and one 6-1 region. This results in at least $74 \cdot 2 + 8 + 2 + 2 = 160$ filled cells.

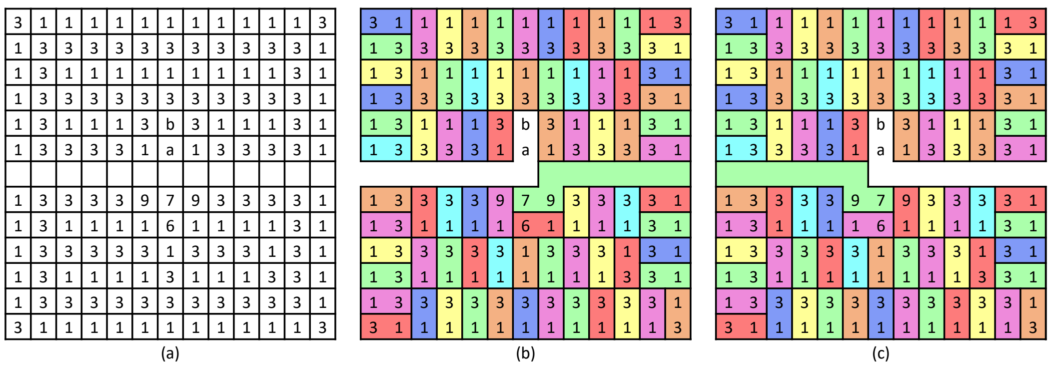

Figure 23 An *Araf* degree-2 straight start room (left) and solutions.

For all 1s except those adjacent to a or 6, it is obvious that they must connect to a neighboring 3. For each of the 1s next to a, if it would connect elsewhere then the only options would be a 7 or 9. However, the 7 is impossible because this would block the a and b,

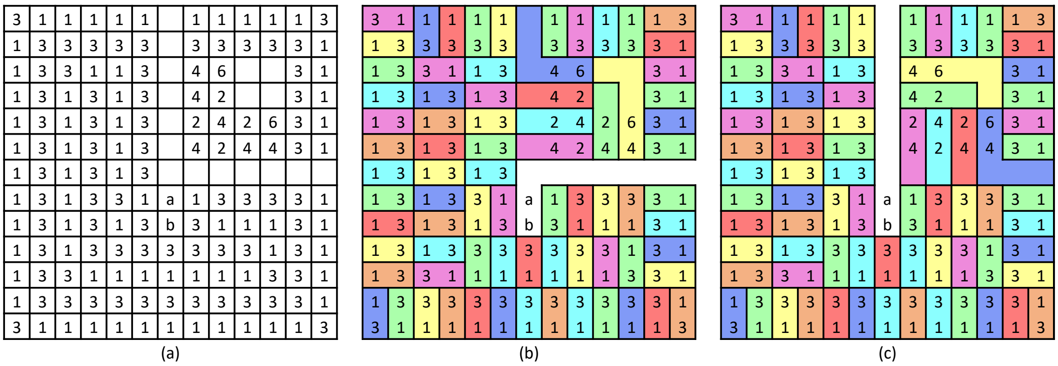

Figure 24 An *Araf* degree-2 corner start room (left) and solutions.

whose region requires far more than just two cells. Likewise for the far 9. For the close 9 (two spaces below the 1), connecting would leave a 3 from the same quadrant as the 1 unpaired, forcing the 3 to connect to the 7 or remaining 9. But this, even if somehow possible, would again block off a and b. So each 1 next to a must connect to a neighboring 3.

The 7 must now connect to a 9, as it cannot connect to a 6 by the rules of *Araf*. The resulting 9-7 region requires exactly eight cells. To avoid blocking the a and b, this region must be as in Figure 23(b) or (c). All other cells in the central row must remain open to be filled by the b-a (Hamilton) region.

We thus see that the remaining 9 cannot connect to the adjacent 3, because the resulting region would need at least four cells, including two central row cells. Hence the 9 must connect to the adjacent 1, forming a region that can and must use just two cells.

Finally, the 6-1 region can also only occupy two cells, because of the limited space around it. Adding this up with the other regions, 160 cells are filled, leaving nine for the Hamilton region. This can be done exactly two ways, depicted in Figure 23.

Figure 24 shows the corner degree-2 start room. We argue that a solution must have sixty-nine 3-1 regions, four 4-2 regions, and two 6-4 regions. This results in $69 \cdot 2 + 4 \cdot 3 + 2 \cdot 5 = 160$ filled cells. So the remaining nine cells must be filled by the Hamilton region.

Here, the 6s must connect to the adjacent 4s because those 4s have no alternative. Further, the b cannot connect to a 3 as argued previously, so it must connect to a and form the Hamilton region. The 1 to the right of a cannot make a 1-4 region, as that blocks the east hallway while also forcing a 2-4 region above to block the north hallway. This leaves two, somewhat symmetrical, ways to solve the room, as seen in Figure 24.

Figure 25 shows the degree-3 start room, constructed similarly to the corner degree-2 start room but with a southeast quadrant mirroring the northeast. Forced in its solution are fifty-eight 3-1 regions, eight 4-2 regions, and four 6-4 regions. This results in $58 \cdot 2 + 8 \cdot 3 + 4 \cdot 5 = 160$ filled cells, leaving nine for the Hamilton region. In each of the two eastern quadrants, the numbers' regions can together fill either a 7×6 or 6×7 rectangular area, resulting in three ways to solve the room. Two are shown in the figure. The third, with Hamilton region heading north, mirrors the first.

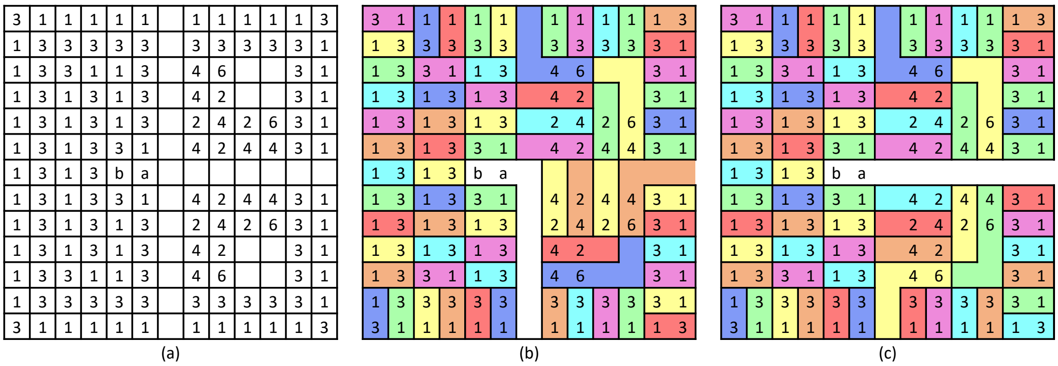

Figure 25 An *Araf* degree-3 start room (left) and two of three solutions.

We now present the nonterminal rooms. As degree-1 nonterminal rooms are unnecessary, we begin in Figures 26 and 27 with the degree-2 nonterminal rooms, which require no comment.

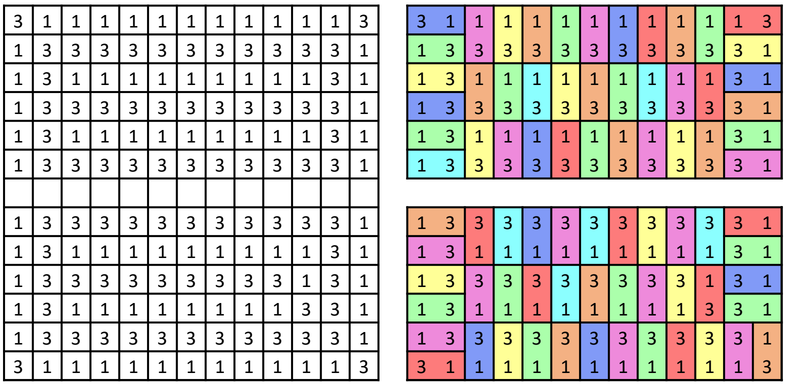

Figure 26 An *Araf* degree-2 straight nonterminal room (left) and its solution.

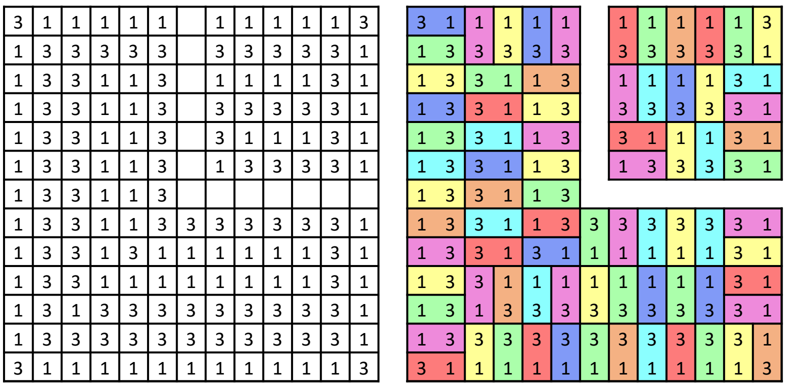

Figure 27 An *Araf* degree-2 corner nonterminal room (left) and its solution.

The degree 3 nonterminal room, shown in Figure 28, is more complex. We claim it has thirty-nine 3-1 regions, two 4-1 regions, seven 5-3 regions, two 6-3 regions, two 7-5 regions, two 8-6 regions and two 6-4 regions. Two solutions are shown, and a third symmetrical to the second is also possible.

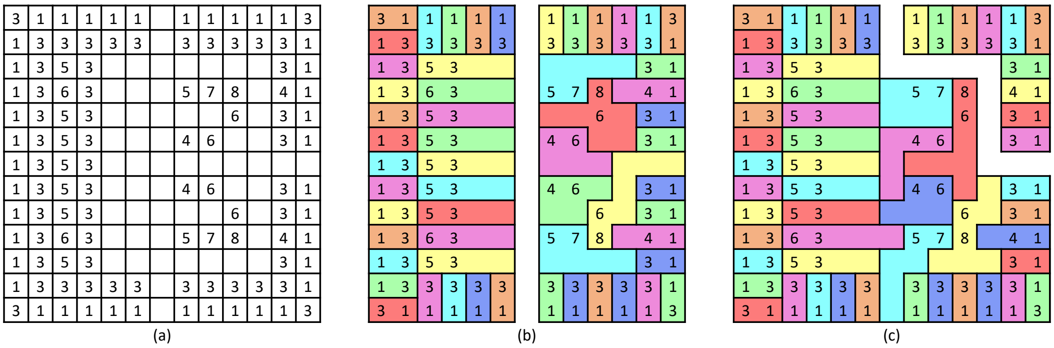

Figure 28 An *Araf* degree-3 nonterminal room (left) and two of the solutions.

Note that all cells in the west side will always be filled. In the east is some flexibility. However, observe that all 1s must connect to the adjacent 3s and 4s. This leaves two areas that each contain numbers 4, 5, 6, 6, 7, and 8. In each, since a 6 can never connect to 5 or 7, the 6s must connect to the 8 and 4. So the 5 must connect to 7. This yields regions of exactly five, exactly six, and exactly seven cells.

The two 6-3 regions can fill either four or five cells and the two 4-1 regions can fill two or three cells. This means that the total area covered by regions of numbers in the room is between 154 and 158 cells, leaving between eleven and fifteen for the Hamilton region.

The images in Figure 28 show the intended ways to solve this room, but unlike the previous rooms, this one offers freedom for the solver to place regions in other ways that are at least locally acceptable within the rules. See Figure 29. We argue, however, that these lead to global failure for solving the puzzle instance as a whole, or at any rate do not render solvable any puzzle instances whose corresponding graphs have no Hamiltonian paths.

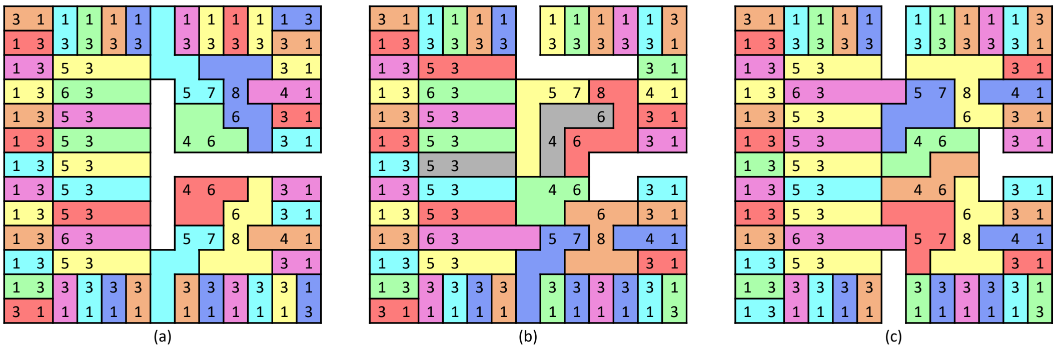

Figure 29 Alternative solution attempts for *Araf* degree-3 nonterminal room.

First we note that since eleven or more cells are unfilled by regions of numbers in the room itself, the Hamilton region must visit there. The only question is whether/how it exits. But failing to exit, as in image (a), leaves the Hamilton region with at least three endpoints: the start room, the end room, and this room. The only way for this to occur is if there is some room (with degree 3) elsewhere whose three hallways are all used. But this is impossible, as it would require nineteen cells for the Hamilton region (the normal thirteen plus six more), which exceeds the maximum (fifteen) established above. Solution attempts such as (b) and (c) can be similarly disregarded.

Having now handled all start and nonterminal rooms, we finally turn to end rooms. All of these except for the straight degree-2 room are constructed identically to their corresponding start rooms, but with a 3-1 or 1-3 domino in place of a and b. Their solutions are the same as the start rooms', except with the two cells formerly occupied by a and b now enclosed within a two-cell region, leaving seven (not nine) cells for occupation by the Hamilton region. We show just one example (Figure 30).

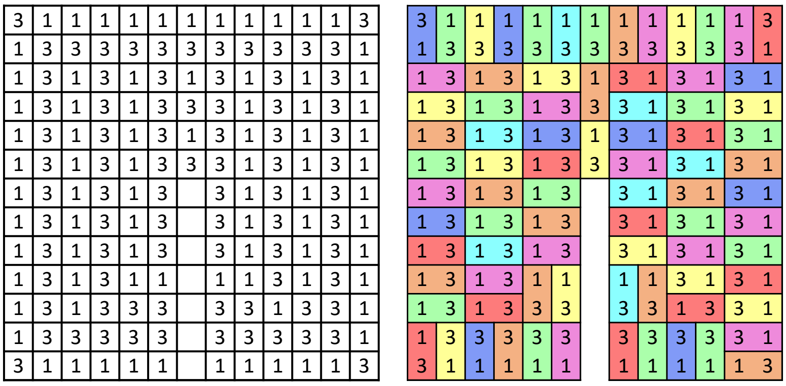

Figure 30 An *Araf* degree-1 end room (left) and its solution.

For the case of the straight degree-2 end room, Figure 31 shows its layout and one of its two solutions. The other is symmetrical. We claim that the room contains sixty-nine 3-1 regions, two 8-6 regions and two 6-4 regions. This results in $69 \cdot 2 + 2 \cdot 7 + 2 \cdot 5 = 162$ filled cells, with the remaining seven cells being filled by the Hamilton region.

Observe that each 6 adjacent to an 8 must connect to the 8, because the only alternative is to connect to a 3, which would leave a 1 optionless. The other 6s must then connect to their neighboring 4s, since the 4s have no other option.

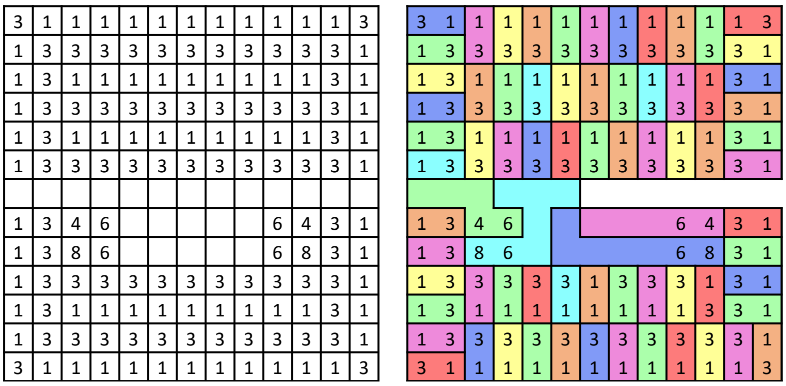

Figure 31 An *Araf* degree-2 end room (left) and one of two solutions.

Having covered all the gadgets, we conclude the proof. In Example 7 we show a full example of an *Araf* puzzle instance that simulates a grid graph. ◀

▶ **Example 7.** The graph in Figure 32 has five nodes, so we have $a = 5(13) - 11 = 54$ and $b = a + 2 = 56$. Figure 33 gives the puzzle instance's solution.

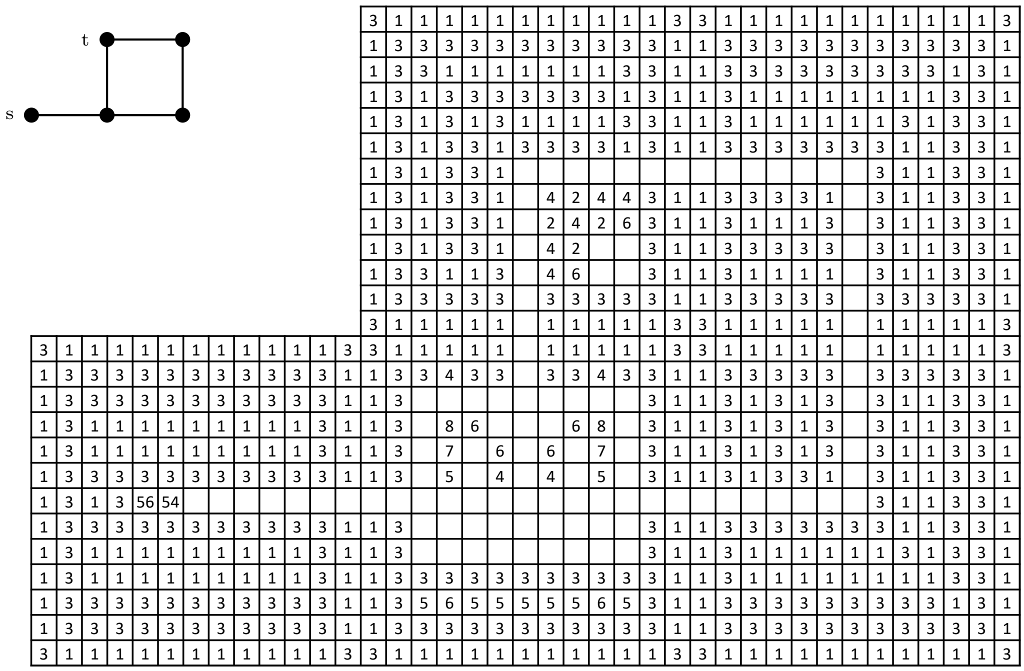

Figure 32 Example graph and its corresponding *Araf* instance.

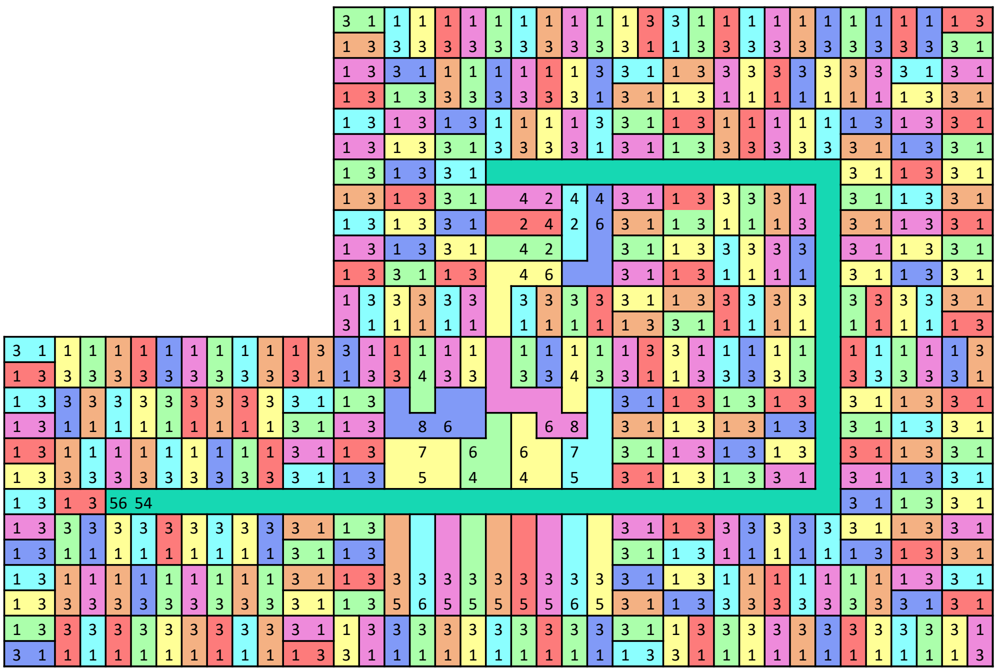

Figure 33 *Araf* example solution with 56-54 region representing a Hamiltonian path.

7 *Dotchi Loop*, revisited

We now come full circle to our first puzzle, *Dotchi Loop*. In this section we show that *Dotchi Loop* remains NP-complete even when the set of available puzzle instances is restricted so that black circles are forbidden from being orthogonally or diagonally adjacent. Let us call the solvability problem for *Dotchi Loop* under these conditions *Restricted Dotchi Loop*. It is easy to see that the formation of walls in Restricted *Dotchi Loop* is much more difficult than in unrestricted *Dotchi Loop* (Section 2).

▶ **Theorem 7.** *Restricted* Dotchi Loop *is NP-complete.*

Proof. We again give a reduction from HC3G. Although the formation of walls in the manner of our earlier proof is difficult or impossible, the key idea in our construction is now the use of *the loop itself* to form a sort of wall. Here we make use of the fact that the loop is forbidden to cross itself. By forcing the loop to surround each room, we give it no opportunity to wander astray.

The degree-3 room is shown in Figure 34. Here, the letter F is not part of the gadget, but just a label for our reference (see below). Degree-2 rooms (not shown) are constructed similarly, but with changes we now describe.

All rooms have 6×6 regions in the corners with black circles arranged as in the figure. The **Gg-Gt-Tg-Tt** square is also the same as in the figure. On a side with no exit, between the two 6×6 regions in the corners is a rectangle like the **Ga-Gf-Ta-Tf** rectangle shown, rotated appropriately. In a side with an exit, there is a rectangle resembling the **Gu-Gz-Tu-Tz** rectangle shown, rotated appropriately. For example, the degree-2 corner room with hallways north and east is identical to the degree-3 room depicted, except with its **Ug-Ut-Zg-Zt** rectangle replaced by a left-rotated copy of the **Ga-Gf-Ta-Tf** rectangle.

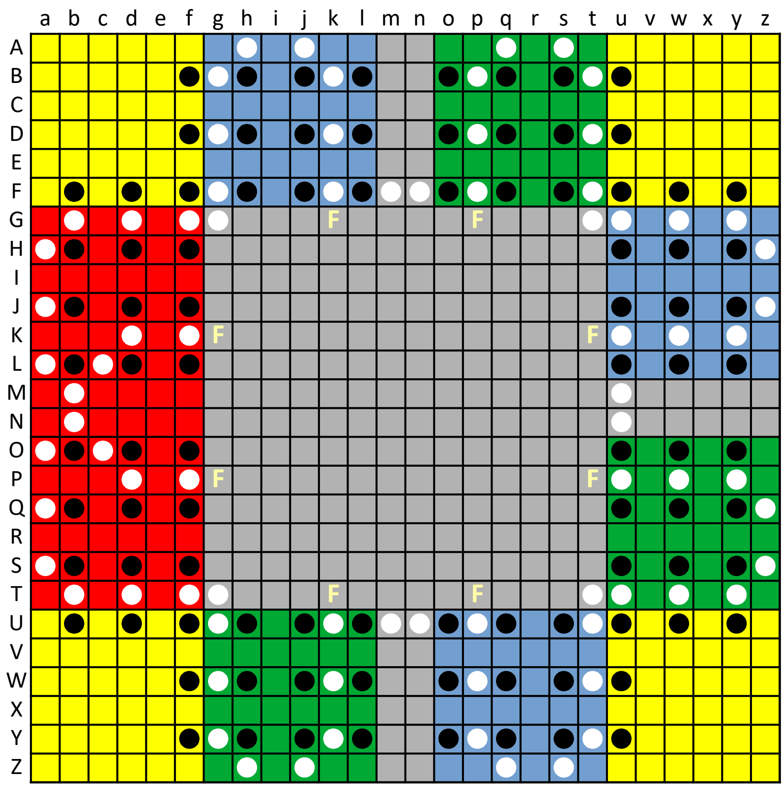

Figure 34 Restricted *Dotchi Loop* degree-3 room. There are twelve distinct regions. As in Figure 5, colors distinguish neighboring regions, but distant regions may be shown with similar color.

We now describe how these rooms work, using the degree-3 room as an example. We first claim that all regions containing white circles are straight regions, except for each room's large central light gray region (containing the **Gg-Gt-Tg-Tt** square and its 2×6 attachments – e.g., between **Mu** and **Nz**). To prove this, note that a region containing a white circle flanked by two black circles to form a black-white-black I-tromino (e.g., at **Jf-Lf** or **Lb-Ld**) must be straight, because such a white circle gives the loop no space to turn through it.

Further, once the loop enters a cell adjacent to any of the straight white circles, it must go immediately onto that white circle, as going elsewhere would make that circle unsolvable. Therefore, a sequence of such I-trominoes with one-space gaps between them, such as the sequence **Bf-Bh**, **Df-Dh**, **Ff-Fh**, must be solved with a straight line directly through the white circles, as seen in Figure 35.

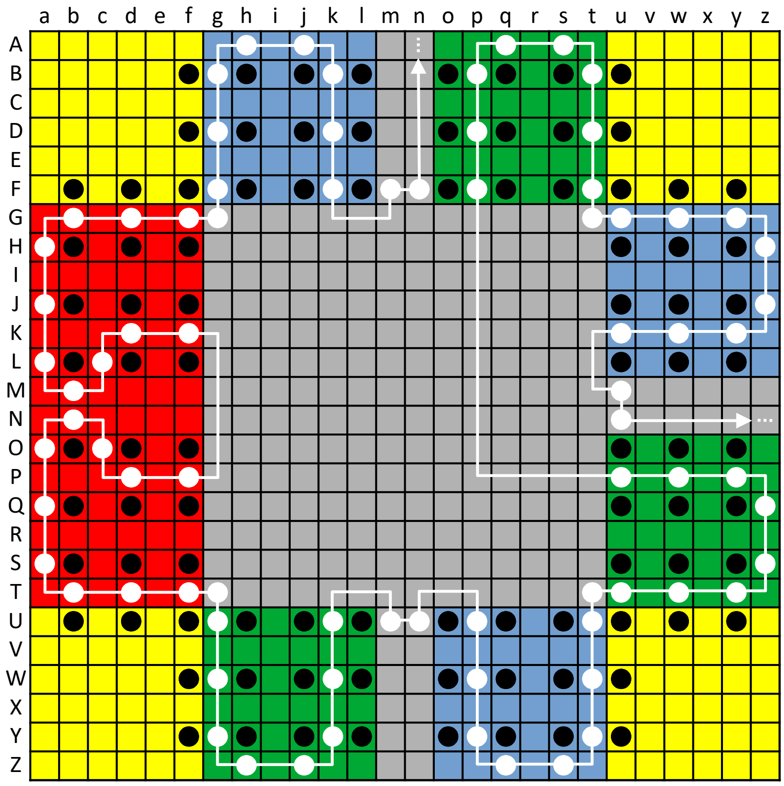

Figure 35 One of three (non-equivalent) solutions to Restricted *Dotchi Loop* degree-3 room.

We can also see that the cells earlier marked "F" (e.g., cell **Gk**) can be considered *forced entrances/exits* from the large central region to the side regions, because the white circles adjacent to them are necessarily straight. Finally, we can infer that the large central region must be a turn region, because the circle at **Gg** (among others) cannot be a straight circle without causing the loop to block off one of the two adjacent (now known to be straight) white circles.

These facts together suffice to force the entire solution to be as in Figure 35, for a loop traveling through this room using the north and east entrances (*mutatis mutandis* for north-south or east-south) – save for inconsequential differences such as getting things done in a slightly different order, or making some irrelevant additional movements within the large empty space in the central region. That is to say, deviating meaningfully from the depicted solution would prevent one or more of the white circles from being crossed properly (turn or straight, according to its region).

We now see how the loop is forced to form its own self-impenetrable wall around the room, save for an opening at each of exactly two of the 2×6 areas mentioned earlier. These areas are the "corridors" of this reduction and each one joins with one from a neighboring room (as they are aligned at the center). Accordingly, we now see that the rooms work as they need to. ◀

8 Discussion and Future Research

8.1 Application of HP3G and HC3G

We have seen how HP3G and HC3G are useful graph-theoretic computational problems for giving reductions to some pen-and-paper puzzles, particularly those occurring on square grids with a kind of path- or loop-finding aspect. But even *Araf* was successfully proven NP-complete using one of these, despite the lack of any overt requirement to construct a path or loop in the puzzle's rules. Therefore, HP3G and HC3G seem to be at least moderately broadly applicable problems that should be taken seriously as candidates for future such reductions – certainly, at least, when proofs using common choices such as 3SAT are not forthcoming.

8.2 Restricted puzzle versions

In general, we find it interesting to consider whether a given puzzle remains NP-complete when the set of available puzzle instances is restricted in some way. We explored this in Section 7 for *Dotchi Loop*. In deliberately making the task difficult by eliminating the means to create easy walls, we became forced to find an alternative approach using the loop's path itself. Similar such results may be available for the other puzzles. For example, we may consider Linesweeper without the use of 0-cells (despite footnote 8). We suspect that HP3G and HC3G can still serve as bases for reductions to these restricted puzzles, but the task gets more challenging the more tools we deprive ourselves of using to (for example) build walls.

Restricted versions of puzzles compare to the original versions in a way analogous to how 3SAT compares to SAT, or HP3G and HCG3 respectively compare to the Hamiltonian path and cycle problems on arbitrary graphs. As potential bases for reductions in future NP-completeness proofs, they require the construction of fewer gadgets than the originals. But even in the likely scenario that no future NP-completeness proofs use these particular puzzles as bases for reductions, we still find the possibility of nontrivially restricted versions being NP-complete inherently interesting.

8.3 HP4G and HC4G

As a side remark, we are grateful for the existence of Theorem 1 and Corollary 1, which have allowed us to give reductions without needing to construct degree-4 rooms. Had we been forced to construct such rooms in addition to the others, our task would have been rather harder. While a degree-4 room is in some cases very simple to construct after seeing the lower-degree rooms, in other cases their design is not so trivial. *Dotchi Loop,* Chains, and *Arukone*$_3$ are easy cases (see Figure 36(a)-(c)). Linesweeper is more challenging, but could be done as in Figure 36(d), if the lower-degree rooms are appropriately padded with 0s to match the dimensions. But *Araf* and Restricted *Dotchi Loop* bear no constructions we have been able to find. For *Araf,* the lower-degree rooms do not fit a clear pattern that can be straightforwardly extrapolated to degree 4, and overall we simply have not found any design that works. For Restricted *Dotchi Loop,* such a pattern very clearly does exist, but a naive attempt to design degree-4 rooms this way allows a loop to enter and exit the same degree-4 room twice, resulting in some non-Hamiltonian graphs getting mapped to solvable puzzle instances. (By contrast, the rooms in Figure 36 all prevent this.) Although these last two puzzles are provably NP-complete anyway, and degree-4 rooms are of course possible to build "the long way" by reducing HP4G → HP3G → *Araf* and HC4G → HC3G → Restricted DL, we wonder as a matter of pure curiosity whether these puzzles admit "direct" degree-4 room designs – i.e., ones as small and natural as those of the lower-degree rooms.

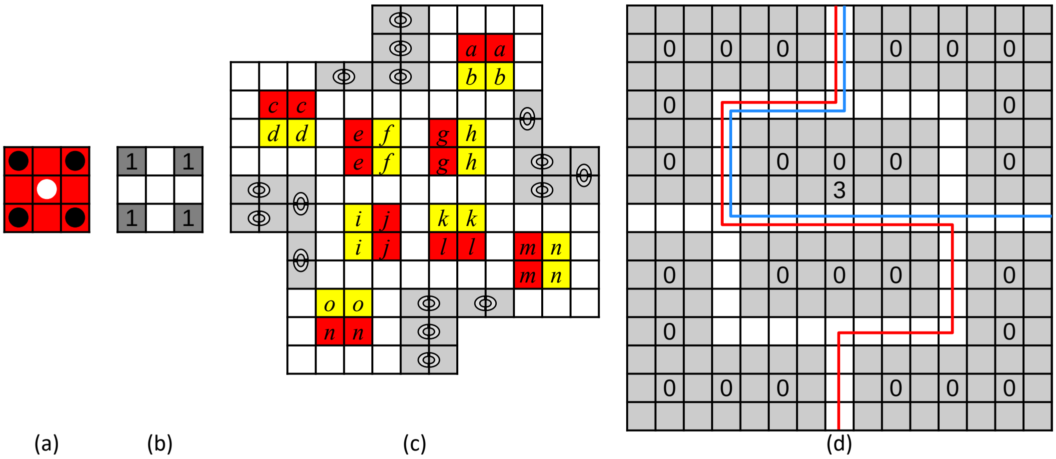

Figure 36 Degree-4 rooms of *Dotchi Loop,* Chains, *Arukone*$_3$, and Linesweeper. Rooms (b) and (c) can be made terminal by inserting an appropriate number in the center. Room (d) is given with two of six solutions shown.

References

1 Aaron B. Adcock, Erik D. Demaine, Martin L. Demaine, Michael P. O'Brien, Felix Reidl, Fernando Sánchez Villaamil, and Blair D. Sullivan. Zig-Zag Numberlink is NP-complete. *Journal of Information Processing*, 23(3):239–245, 2015. doi:10.2197/IPSJJIP.23.239.

2 Aviv Adler, Jeffrey Bosboom, Erik D. Demaine, Martin L. Demaine, Quanquan C. Liu, and Jayson Lynch. Tatamibari is NP-complete. In Martin Farach-Colton, Giuseppe Prencipe, and Ryuhei Uehara, editors, *10th International Conference on Fun with Algorithms (FUN 2021)*, volume 157 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 1:1–1:24, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.FUN.2021.1.

3 Michael Buro. Simple Amazons endgames and their connection to Hamilton circuits in cubic subgrid graphs. In T. Anthony Marsland and Ian Frank, editors, *Computers and Games, Second International Conference, CG 2000, Hamamatsu, Japan, October 26-28, 2000, Revised Papers*, volume 2063 of *Lecture Notes in Computer Science*, pages 250–261. Springer, 2001. doi:10.1007/3-540-45579-5_17.

4 Erik D. Demaine, Jayson Lynch, Mikhail Rudoy, and Yushi Uno. Yin-Yang Puzzles are NP-complete. In Meng He and Don Sheehy, editors, *Proceedings of the 33rd Canadian Conference on Computational Geometry, CCCG 2021, August 10-12, 2021, Dalhousie University, Halifax, Nova Scotia, Canada*, pages 97–106, 2021.

5 Marnix Deurloo. On the NP-completeness of Dotchi Loop and restricted sets thereof. Bachelor's thesis, Utrecht University, 2023.

6 Mitchell Donkers. The NP-completeness of pen and paper puzzles. Bachelor's thesis, Utrecht University, 2021. URL: https://studenttheses.uu.nl/handle/20.500.12932/1383.

7 Markus Holzer and Oliver Ruepp. The troubles of interior design–A complexity analysis of the game Heyawake. In Pierluigi Crescenzi, Giuseppe Prencipe, and Geppino Pucci, editors, *Fun with Algorithms, 4th International Conference, FUN 2007, Castiglioncello, Italy, June 3-5, 2007, Proceedings*, volume 4475 of *Lecture Notes in Computer Science*, pages 198–212. Springer, 2007. doi:10.1007/978-3-540-72914-3_18.

8 Otto Janko and Angela Janko. Araf. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Araf/index.htm.

9 Otto Janko and Angela Janko. Arukone. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Arukone/index.htm.

10 Otto Janko and Angela Janko. Arukone^2. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Arukone-2/index.htm.

11 Otto Janko and Angela Janko. Arukone^3. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Arukone-3/index.htm.

12 Otto Janko and Angela Janko. Dotchi loop. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Dotchi-Loop/index.htm.

13 Otto Janko and Angela Janko. Ketten. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Ketten/index.htm.

14 Otto Janko and Angela Janko. Ketten, nr. 7. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Ketten/007.a.htm.

15 Otto Janko and Angela Janko. Linesweeper. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Linesweeper/index.htm.

16 Otto Janko and Angela Janko. Linesweeper, nr. 082. Accessed on 2024-02-20. URL: https://www.janko.at/Raetsel/Linesweeper/082.a.htm.

17 Richard Kaye. Minesweeper is NP-complete. *The Mathematical Intelligencer*, 22:9–15, 2000. doi:10.1007/BF03025367.

18 Jonas Kölker. Kurodoko is NP-complete. *Journal of Information Processing*, 20(3):694–706, 2012. doi:10.2197/ipsjjip.20.694.

19 Kotsuma Kouichi and Takenaga Yasuhiko. NP-completeness and enumeration of Number Link puzzle. *IEICE Technical Report. Theoretical Foundations of Computing*, 109(465):1–7, 2010.

20 Mieke Maarse. The NP-completeness of some lesser known logic puzzles. Bachelor's thesis, Utrecht University, 2019. URL: https://studenttheses.uu.nl/handle/20.500.12932/33867.

21 Jak Marshall. Play my puzzle game: Linesweeper. September 30, 2010. URL: https://103percent.blogspot.com/2010/09/play-my-new-puzzle-game.html.

22 Christos H. Papadimitriou and Umesh V. Vazirani. On two geometric problems related to the travelling salesman problem. *Journal of Algorithms*, 5(2):231–246, June 1984. doi:10.1016/0196-6774(84)90029-4.

23 Karen Schutte. The NP-completeness of three logic puzzles. Bachelor's thesis, Utrecht University, 2021.

24 Allan Scott, Ulrike Stege, and Iris Rooij. Minesweeper may not be NP-complete but is hard nonetheless. *The Mathematical Intelligencer*, 33.4:5–17, 2011. doi:10.1007/s00283-011-9256-x.

25 Hadyn Tang. A framework for loop and path puzzle satisfiability NP-hardness results, 2022. arXiv:2202.02046.

26 Takayuki Yato and Takahiro Seta. Complexity and completeness of finding another solution and its application to puzzles. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 86-A(5):1052–1060, 2003. URL: http://search.ieice.org/bin/summary.php?id=e86-a_5_1052.