References

[1] R. Kimball. *Path Puzzles*. Enigami Puzzles & Games, 2013.

[2] New York Times Wordplay. Roderick kimball's path puzzles. `https://archive.nytimes.com/wordplay.blogs.nytimes.com/2014/07/28/path-2/`, October 2022. Accessed: 2022-11-14.

[3] Jeffrey Bosboom, Erik D Demaine, Martin L Demaine, Adam Hesterberg, Roderick Kimball, and Justin Kopinsky. Path puzzles: Discrete tomography with a path constraint is hard. *Graphs and Combinatorics*, 36(2):251–267, 2020.

[4] Scott Kim. What is a puzzle. In *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*, pages 35–39, 2008.

[5] Erik D Demaine. Playing games with algorithms: Algorithmic combinatorial game theory. In *International Symposium on Mathematical Foundations of Computer Science*, pages 18–33. Springer, 2001.

[6] Graham Kendall, Andrew Parkes, and Kristian Spoerer. A survey of NP-complete puzzles. *ICGA Journal*, 31(1):13–34, 2008.

[7] Robert A Hearn and Erik D Demaine. *Games, puzzles, and computation*. CRC Press, 2009.

[8] Ryuhei Uehara. Computational complexity of puzzles and related topics. *Interdisciplinary Information Sciences*, pages 1–22, 2023.

[9] Ayaka Ishibashi, Yuichi Sato, and Shigeki Iwata. NP-completeness of two pencil puzzles: Yajilin and Country Road. *Utilitas Mathematica*, 88:237–246, 2012.

[10] Daniel Andersson. Hashiwokakero is NP-complete. *Information Processing Letters*, 109(19):1145–1146, 2009.

[11] Daniel Andersson. Hiroimono is NP-complete. In *International Conference on Fun with Algorithms*, pages 30–39. Springer, 2007.

[12] Chuzo Iwamoto and Tatsuaki Ibusuki. Kurotto and Juosan are NP-complete. In *The 21st Japan Conference on Discrete and Computational Geometry, Graphs, and Games (JCDCG3 2018)*, pages 46–48, 2018.

[13] Chuzo Iwamoto and Tatsuaki Ibusuki. Polynomial-time reductions from 3SAT to Kurotto and Juosan puzzles. *IEICE Transactions on Information and Systems*, 103(3):500–505, 2020.

[14] Daniel Král, Vladan Majerech, Jiří Sgalla, Tomáš Tichý, and Gerhard Woegingerd. It is tough to be a plumber. *Theoretical computer science*, 313:473–484, 2004.

[15] Richard Kaye. Minesweeper is NP-complete. *Mathematical Intelligencer*, 22(2):9–15, 2000.

[16] Chuzo Iwamoto and Tatsuya Ide. Moon-or-Sun, Nagareru, and Nurimeizu are NP-complete. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, page 2021DMP0006, 2022.

[17] Nobuhisa Ueda and Tadaaki Nagao. NP-completeness results for Nonogram via parsimonious reductions. Technical report, Department of Computer Science, Tokyo Institute of Technology, 1996.

[18] Brandon P McPhail. *Complexity of puzzles: NP-Completeness results for Nurikabe and Minesweeper*. PhD thesis, Reed College, 2003.

[19] Markus Holzer, Andreas Klein, and Martin Kutrib. On the NP-completeness of the Nurikabe pencil puzzle and variants thereof. In *Proceedings of the 3rd International Conference on FUN with Algorithms*, pages 77–89. Citeseer, 2004.

[20] Yasuhiko Takenaga, Shintaro Aoyagi, Shigeki Iwata, and Takumi Kasai. Shikaku and Ripple Effect are NP-complete. *Congressus Numerantium*, 216:119–127, 2013.

[21] Yato Takayuki. On the NP-completeness of the Slither Link puzzle. *IPSJ SIGNotes ALgorithms*, 2000.

[22] 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(5):1052–1060, 2003.

[23] Léo Robert, Daiki Miyahara, Pascal Lafourcade, Luc Libralesso, and Takaaki Mizuki. Physical zero-knowledge proof and NP-completeness proof of Suguru puzzle. *Information and Computation*, 285:104858, 2022.

[24] 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.

[25] Chuzo Iwamoto and Tatsuya Ide. Five Cells and Tilepaint are NP-Complete. *IEICE Transcations on Information and Systems*, 105(3):508–516, 2022.

[26] Erik D. Demaine, Jayson Lynch, Mikhail Rudoy, and Yushi Uno. Yin-Yang Puzzles are NP-complete. In *33rd Canadian Conference on Computational Geometry (CCCG) 2021*, 2021.

[27] Sagnik Saha and Erik D Demaine. ZHED is NP-complete. In *Proceedings of the 34th Canadian Conference on Computational Geometry (CCCG 2022)*, 2022.

[28] Erik D Demaine, Yoshio Okamoto, Ryuhei Uehara, and Yushi Uno. Computational complexity and an integer programming model of Shakashaka. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 97(6):1213–1219, 2014.

[29] Andrew Bartlett, Timothy P Chartier, Amy N Langville, and Timothy D Rankin. An integer programming model for the Sudoku problem. *Journal of Online Mathematics and its Applications*, 8(1), 2008.

[30] Curtis Bright, Jürgen Gerhard, Ilias Kotsireas, and Vijay Ganesh. Effective problem solving using SAT solvers. In *Maple Conference*, pages 205–219. Springer, 2019.

[31] Enrico Christopher Reinhard, Muhammad Arzaki, and Gia Septiana Wulandari. Solving Tatamibari Puzzle Using Exhaustive Search Approach. *Indonesia Journal on Computing (Indo-JC)*, 7(3):53–80, 2022.

[32] Made Indrayana Putra, Muhammad Arzaki, and Gia Septiana Wulandari. Solving Yin-Yang Puzzles Using Exhaustive Search and Prune-and-Search Algorithms. *(IJCSAM) International Journal of Computing Science and Applied Mathematics*, 8(2):52–65, 2022.

[33] Michael R Garey and David S Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. WH Freeman & Co., 1979.

[34] HJ Ryser. Combinatorial properties of matrices of zeros and ones. *Canadian Journal of Mathematics*, 9:371–377, 1957.

[35] Gabor T Herman and Attila Kuba. *Discrete tomography: Foundations, algorithms, and applications*. Springer Science & Business Media, 2012.

[36] Hendrik Jan Hoogeboom, Walter A Kosters, Jan N van Rijn, and Jonathan K Vis. Acyclic constraint logic and games. *ICGA Journal*, 37(1):3–16, 2014.

[37] Michael Huth and Mark Ryan. *Logic in Computer Science: Modelling and Reasoning about Systems, 2nd Edition*. Cambridge university press, 2004.

[38] Martin Hořeňovský. Modern sat solvers: fast, neat and underused (part 1 of *n*). `https://codingnest.com/modern-sat-solvers-fast-neat-underused-part-1-of-n/`, April 2019. Accessed: 2023-5-22.

[39] Mordechai Ben-Ari. *Mathematical Logic for Computer Science, 3rd Edition*. Springer Science & Business Media, 2012.

[40] Alan M Frisch and Paul A Giannaros. SAT encodings of the at-most-k constraint: Some old, some new, some fast, some slow. 2010.

[41] Paul Maximilian Bittner, Thomas Thüm, and Ina Schaefer. SAT Encodings of the At-Most-k Constraint: A Case Study on Configuring University Courses. In *Software Engineering and Formal Methods: 17th International Conference, SEFM 2019, Oslo, Norway, September 18–20, 2019, Proceedings 17*, pages 127–144. Springer, 2019.

[42] Van-Hau Nguyen, Van-Quyet Nguyen, Kyungbaek Kim, and Pedro Barahona. Empirical Study on SAT-Encodings of the At-Most-One Constraint. In *The 9th International Conference on Smart Media and Applications*, pages 470–475, 2020.

[43] Martin Hořeňovský. Modern sat solvers: fast, neat and underused (part 3 of *n*). `https://codingnest.com/modern-sat-solvers-fast-neat-and-underused-part-3-of-n/`, April 2019. Accessed: 2023-5-22.

[44] Bengt Aspvall, Michael F Plass, and Robert Endre Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information processing letters*, 8(3):121–123, 1979.

[45] Sara Brunetti and Alain Daurat. An algorithm reconstructing convex lattice sets. *Theoretical Computer Science*, 304(1):35–57, 2003.

[46] Jonathan Irvin Gunawan. Understanding Unsolvable Problem. *Olympiad in Informatics*, 10:87–98, 2016.

[47] Lutz Prechelt. An empirical comparison of seven programming languages. *Computer*, 33(10):23–29, 2000.

[48] Erik D. Demaine. Path Puzzles Font. `https://github.com/edemaine/font-pathpuzzles`, October 2022. Accessed: 2022-10-11.