Proceedings Paper:

Gent, Ian Philip, Jefferson, Christopher Anthony and Nightingale, Peter William orcid.org/0000-0002-5052-8634 (2018) Complexity of n-Queens completion (extended abstract). In: Lang, Jérôme, (ed.) Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence , pp. 5608-5611.

10.24963/ijcai.2018/794

Complexity of n-Queens Completion (Extended Abstract)[*]

Ian P. Gent, Christopher Jefferson, Peter Nightingale
School of Computer Science, University of St Andrews, Fife KY16 9SX, UK
ian.gent@st-andrews.ac.uk, caj21@st-andrews.ac.uk, pwn1@st-andrews.ac.uk

Abstract

The n-Queens problem is to place n chess queens on an n by n chessboard so that no two queens are on the same row, column or diagonal. The n-Queens Completion problem is a variant, dating to 1850, in which some queens are already placed and the solver is asked to place the rest, if possible. We show that n-Queens Completion is both NP-Complete and #P-Complete. A corollary is that any non-attacking arrangement of queens can be included as a part of a solution to a larger n-Queens problem. We introduce generators of random instances for n-Queens Completion and the closely related Blocked n-Queens and Excluded Diagonals Problem. We describe three solvers for these problems, and empirically analyse the hardness of randomly generated instances. For Blocked n-Queens and the Excluded Diagonals Problem, we show the existence of a phase transition associated with hard instances as has been seen in other NP-Complete problems, but a natural generator for n-Queens Completion did not generate consistently hard instances. The significance of this work is that the n-Queens problem has been very widely used as a benchmark in Artificial Intelligence, but conclusions on it are often disputable because of the simple complexity of the decision problem. Our results give alternative benchmarks which are hard theoretically and empirically, but for which solving techniques designed for n-Queens need minimal or no change.

1 Introduction

The n-Queens problem is to place n chess queens on an n by n chessboard so that no two queens are on the same row, column or diagonal. This puzzle dates to 1848, and only two years later a variant was introduced by Nauck [1850] in which some number of queens are pre-placed and the solver is asked to place the rest, if possible. This is the n-Queens Completion problem and Figure 1 shows the first known instance studied.

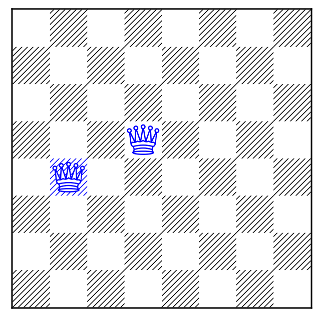

Figure 1: This is the first published instance of the n-Queens Completion problem, by Nauck [1850]. The reader may enjoy attempting to place 6 more queens on the chessboard so that no two queens attack each other. Is it possible? If so, how many different ways are there to do it? The answers to these questions are given below in Figure 2.

We will show that the n-Queens Completion problem is NP-Complete and #P-Complete, discuss solvers for the problem, and empirically analyse randomly generated instances. The n-Queens Completion problem may be one of the simplest NP-Complete problems to explain to people who understand the rules of chess. The problem is "Given an $n \times n$ chessboard on which some queens are already placed, can you place a queen in every remaining row so that no two queens attack each other?"

1.1 History of n-Queens

The n-Queens problem has an extraordinary history for such an apparently unassuming problem, both generally and inside Artificial Intelligence. Formerly, and incorrectly, attributed to Gauss, the problem's history was clarified by Campbell [1977]. The 8-Queens problem was introduced by Bezzel [1848] and by Nauck [1850] (possibly independently). The latter publication attracted the interest of Gauss, who even made a small mistake in studying the problem.[1] The generalisation to n-Queens has attracted the interest of many

[*]This paper is an extended abstract of an article in the Journal of Artificial Intelligence Research [Gent *et al.*, 2017].

[1]He reported finding 76 solutions but later realised that four of those were erroneous so had only 72.

other mathematicians, whose results are surveyed by Bell and Stevens [2009]. The first paper to describe backtracking search on computer, presented in 1958, does so for the n-Queens problem [Walker, 1960]. Since then it has been used as an example and/or benchmark problem in many classic AI papers, for example at least six with more than 400 citations on Google Scholar at time of writing [Golomb and Baumert, 1965; Bitner and Reingold, 1975; Mackworth and Freuder, 1985; Minton *et al.*, 1992; Selman *et al.*, 1992; Crawford *et al.*, 1996].

1.2 Complexity of *n*-Queens

The complexity of the n-Queens problem is often misunderstood. The decision problem is solvable in constant time since there is a solution for all $n > 3$ so is only NP-hard if P=NP. A witnessing solution can be constructed easily [Bell and Stevens, 2009] but note that the witness (a set of n queens) requires $n \log n$ bits to specify but this is not polynomial in the size of the input, which is only $\log n$ bits. The n-Queens problem has often been incorrectly called NP-hard, even in well-cited papers [Mandziuk, 1995; Martinjak and Golub, 2007; Shah-Hosseini, 2009; Nakaguchi *et al.*, 1999, each with at least 29 citations]. The counting version of the problem, i.e. to determine how many solutions to n-Queens there are, is sequence A000170 of the Online Encyclopedia of Integer Sequences [Sloane, 2016]. The sequence is currently known only to $n = 27$, for which the number of solutions is more than 2.34×10^{17}. No approach better than optimised exhaustive search seems to be known: e.g. the $n = 27$ total was counted using a massively parallel search using FPGAs [Preußer, 2016]. Hsiang *et al.* [2004] show that solving the n-Queens counting problem is "beyond the #P-class". Bell and Stevens [2009] states that this means that there is no closed form expression in n for the number of solutions, but Chaiken *et al.* [2015] claim to give one.[2]

1.3 Generalising to *n*-Queens Completion

Cadoli and Schaerf [2006] studied the n-Queens Completion problem without studying its decision or counting complexity. The most closely related work to ours is by Martin [2007], who proves a rather different generalisation of n-Queens to be NP-complete, the key difference being that some squares can be marked as stopping attacks. In particular, this means that solutions to Martin's problem need not be solutions to the n-Queens problem, unlike n-Queens Completion.

We are contributing to a rich literature on the complexity of puzzles and games. One of the earliest results in the area is that generalised chess is EXPTIME-complete [Fraenkel and Lichtenstein, 1981]. Amongst other games to have been proved NP-complete or harder are the card solitaire games Klondike [Longpré and McKenzie, 2009], Freecell [Helmert, 2003] and Black Hole [Gent *et al.*, 2007], the Sudoku puzzle [Takayuki and Takahiro, 2003], video games like Pac-Man [Viglietta, 2014], and casual games such as Minesweeper [Kaye, 2000], Candy Crush Saga and Bejeweled [Walsh, 2014; Guala *et al.*, 2014]. Many other games are surveyed by Kendall *et al.* [2008] and by Demaine and Hearn [2009]. In each case, the complexity-theorist must define a generalised class of instances. Unlike some of the examples above, this step is completely natural for n-Queens Completion.

[2] It is unclear to us if this is a mathematical dispute or simply a dispute on what it means to be a closed form expression, but in any case Chaiken *et al.*'s formula has not been used to extend knowledge of the number of solutions beyond $n = 27$.

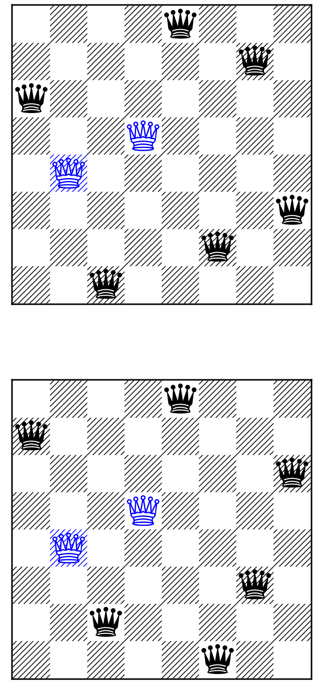

Figure 2: The two possible solutions of the 8-Queens Completion instance from Figure 1.

1.4 Controversy Surrounding *n*-Queens for Benchmarking

Because of the ease of finding a solution, the n-Queens problem has been subject to repeated controversy in AI over whether it should be used as a benchmark at all. For example a sequence of papers argued the point in the pages of SIGART Bulletin in the early 1990s [Sosic and Gu, 1990; Johnson, 1991; Bernhardsson, 1991; Gu, 1991; Valtorta, 1991], and then in 2014 the issue was raised again in a blog post by Smet [2014]. We resolve this issue in the sense that, as an NP- and #P-Complete problem, n-Queens Completion does provide a valid benchmark problem. Similarly, the Quasigroup Completion Problem (which is to complete a partially filled latin square) is NP-Complete [Colbourn, 1984] and is a challenging and popular benchmark [Gomes and Selman, 1997], whereas constructing a latin square from scratch is trivial. A very closely related problem, "Blocked n-Queens", has previously been used for benchmarking without complexity guarantees [Namasivayam and Truszczynski, 2009]. Our results show as a corollary that Blocked n-Queens is NP-Complete and #P-Complete. We explore the practical difficulty of these problems, and a new variant, the Excluded

Diagonals Problem. For Blocked n-Queens and the Excluded Diagonals Problem, we show the existence of a phase transition associated with hard instances, but we were not able to generate consistently hard instances for the n-Queens Completion problem.

1.5 Theoretical Results

In the journal paper [Gent *et al.*, 2017] we give definitions of a sequence of problems, where the first problem is n-Queens Completion, and the last is a variant of Boolean Satisfiability (SAT) that is NP- and #P-Complete, with the Excluded Diagonals Problem at an intermediate stage. The proof proceeds by a sequence of polynomial reductions, starting with the last problem and ending with n-Queens Completion.

Acknowledgements

We thank Jeff Erickson for raising the question of the complexity of (a slight variant of) n-Queens Completion [Erickson, 2014] and Stanislav Živný for bringing his post to our attention by a Facebook post on 1 January 2015. We thank Torsten Schaub, Martin Gebser, Bilal Hussain, and András Salamon. We would like to thank the EPSRC for partially funding this work through grants EP/K015745/1, EP/M003728/1, and EP/P015638/1. In addition, Dr Jefferson is funded by a Royal Society University Research Fellowship.

References

[Bell and Stevens, 2009] Jordan Bell and Brett Stevens. A survey of known results and research areas for n-queens. *Discrete Mathematics*, 309(1):1 – 31, 2009.

[Bernhardsson, 1991] Bo Bernhardsson. Explicit solutions to the n-queens problem for all n. *SIGART Bulletin*, 2(2):7, 1991.

[Bezzel, 1848] Max Bezzel. Schachfreund. *Berliner Schachzeitung*, 3:363, 1848. Cited in [Campbell, 1977].

[Bitner and Reingold, 1975] James R. Bitner and Edward M. Reingold. Backtrack programming techniques. *Communications of the ACM*, 18(11):651–656, 1975.

[Cadoli and Schaerf, 2006] Marco Cadoli and Marco Schaerf. Partial solutions with unique completion. In Oliviero Stock and Marco Schaerf, editors, *Reasoning, Action and Interaction in AI Theories and Systems, Essays Dedicated to Luigia Carlucci Aiello*, volume 4155 of *Lecture Notes in Computer Science*, pages 101–115. Springer, 2006.

[Campbell, 1977] Paul J. Campbell. Gauss and the eight queens problem: A study in miniature of the propagation of historical error. *Historia Mathematica*, 4(4):397 – 404, 1977.

[Chaiken *et al.*, 2015] Seth Chaiken, ChristopherR.H. Hanusa, and Thomas Zaslavsky. A q-queens problem. ii. the square board. *Journal of Algebraic Combinatorics*, 41(3):619–642, 2015.

[Colbourn, 1984] Charles J. Colbourn. The complexity of completing partial latin squares. *Discrete Applied Mathematics*, 8:25–30, 1984.

[Crawford *et al.*, 1996] James M. Crawford, Matthew L. Ginsberg, Eugene M. Luks, and Amitabha Roy. Symmetry-breaking predicates for search problems. In Luigia Carlucci Aiello, Jon Doyle, and Stuart C. Shapiro, editors, *Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR'96), Cambridge, Massachusetts, USA, November 5-8, 1996.*, pages 148–159. Morgan Kaufmann, 1996.

[Demaine and Hearn, 2009] Erik D Demaine and Robert A Hearn. Playing games with algorithms: algorithmic combinatorial game theory. In *Games of No Chance 3*, pages 3–56. Cambridge University Press, 2009.

[Erickson, 2014] Jeff Erickson. Complexity of n-queens-completion? Theoretical Computer Science Stack Exchange, December 2014. URL:http://cstheory.stackexchange.com/q/28002 (version: 2014-12-29).

[Fraenkel and Lichtenstein, 1981] Aviezri S Fraenkel and David Lichtenstein. Computing a perfect strategy for $n \times n$ chess requires time exponential in n. *Journal of Combinatorial Theory, Series A*, 31(2):199 – 214, 1981.

[Gent *et al.*, 2007] Ian P. Gent, Christopher Jefferson, Tom Kelsey, Inês Lynce, Ian Miguel, Peter Nightingale, Barbara M. Smith, and Armagan Tarim. Search in the patience game 'Black Hole'. *AI Communications*, 20(3):211–226, 2007.

[Gent *et al.*, 2017] Ian P. Gent, Christopher Jefferson, and Peter Nightingale. Complexity of *n*-queens completion. *Journal of Artificial Intelligence Research*, 59:815–848, 2017.

[Golomb and Baumert, 1965] Solomon W. Golomb and Leonard D. Baumert. Backtrack programming. *Journal of the ACM*, 12(4):516–524, 1965.

[Gomes and Selman, 1997] Carla P. Gomes and Bart Selman. Problem structure in the presence of perturbations. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI 97)*, pages 221–226, 1997.

[Gu, 1991] Jun Gu. On a general framework for large-scale constraint-based optimization. *SIGART Bulletin*, 2(2):8, 1991.

[Guala *et al.*, 2014] Luciano Guala, Stefano Leucci, and Emanuele Natale. Bejeweled, Candy Crush and other match-three games are (NP-) hard. In *2014 IEEE Conference on Computational Intelligence and Games*, 2014.

[Helmert, 2003] Malte Helmert. Complexity results for standard benchmark domains in planning. *Artificial Intelligence*, 143(2):219–262, 2003.

[Hsiang *et al.*, 2004] Jieh Hsiang, D.Frank Hsu, and Yuh-Pyng Shieh. On the hardness of counting problems of complete mappings. *Discrete Mathematics*, 277(1–3):87 – 100, 2004.

[Johnson, 1991] W. Lewis Johnson. Letter from the editor. *SIGART Bulletin*, 2(2):1, 1991.

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

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

[Longpré and McKenzie, 2009] Luc Longpré and Pierre McKenzie. The complexity of solitaire. *Theoretical Computer Science*, 410(50):5252–5260, November 2009.

[Mackworth and Freuder, 1985] Alan K. Mackworth and Eugene C. Freuder. The complexity of some polynomial network consistency algorithms for constraint satisfaction problems. *Artificial Intelligence*, 25(1):65–74, 1985.

[Mandziuk, 1995] Jacek Mandziuk. Solving the n-queens problem with a binary Hopfield-type network. synchronous and asynchronous model. *Biological Cybernetics*, 72(5):439–446, 1995.

[Martin, 2007] Barnaby Martin. On the complexity of a derivative chess problem. *CoRR*, arXiv:cs/0701049 [cs.CC], 2007.

[Martinjak and Golub, 2007] I. Martinjak and M. Golub. Comparison of heuristic algorithms for the n-queen problem. In *2007 29th International Conference on Information Technology Interfaces*, pages 759–764, June 2007.

[Minton *et al.*, 1992] Steven Minton, Mark D. Johnston, Andrew B. Philips, and Philip Laird. Minimizing conflicts: A heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58(1-3):161–205, 1992.

[Nakaguchi *et al.*, 1999] Toshiya Nakaguchi, JIN'NO Kenya, and Mamoru Tanaka. Hysteresis neural networks for n-queens problems. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 82(9):1851–1859, 1999.

[Namasivayam and Truszczynski, 2009] Gayathri Namasivayam and Miroslaw Truszczynski. Blocked n-queens. `https://dtai.cs.kuleuven.be/events/ASP-competition/Benchmarks/BlockedQueens.shtml`, 2009.

[Nauck, 1850] Franz Nauck. Schach: Eine in das Gebiet der Mathematik fallende Aufgabe von Herrn Dr. Nauck in Schleusingen. *Illustrirte Zeitung*, 14(361):352, June 1850. Cited in [Campbell, 1977].

[Preußer, 2016] Thomas B. Preußer. 27-queens puzzle: Massively parellel enumeration and solution counting. `https://github.com/preusser/q27`, September 2016.

[Selman *et al.*, 1992] Bart Selman, Hector J. Levesque, and David G. Mitchell. A new method for solving hard satisfiability problems. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI-92)*, pages 440–446, 1992.

[Shah-Hosseini, 2009] Hamed Shah-Hosseini. The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm. *International Journal of Bio-Inspired Computation*, 1(1-2):71–79, 2009.

[Sloane, 2016] N. J. A. Sloane. The on-line encyclopedia of integer sequences. `https://oeis.org`, 2016.

[Smet, 2014] Geoffrey De Smet. Cheating on the n queens benchmark. `http://www.optaplanner.org/blog/2014/05/12/CheatingOnTheNQueensBenchmark.html`, May 2014.

[Sosic and Gu, 1990] Rok Sosic and Jun Gu. A polynomial time algorithm for the n-queens problem. *SIGART Bulletin*, 1(3):7–11, 1990.

[Takayuki and Takahiro, 2003] YATO Takayuki and Seta Takahiro. 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.

[Valtorta, 1991] Marco Valtorta. Response to "explicit solutions to the n-queens problem for all n". *SIGART Bulletin*, 2(4):7–5, February 1991.

[Viglietta, 2014] Giovanni Viglietta. Gaming is a hard job, but someone has to do it! *Theory of Computing Systems*, 54(4):595–621, 2014.

[Walker, 1960] Robert J Walker. An enumerative technique for a class of combinatorial problems. In Richard Bellmann and Marshall Hall Jr, editors, *Combinatorial Analysis: Proceedings Of The Tenth Symposium In Applied Mathematics Of The American Mathematical Society*, pages 91–94. American Mathematical Society, 1960.

[Walsh, 2014] Toby Walsh. Candy Crush is NP-hard. *CoRR*, arXiv:1403.1911 [cs.CC], 2014.