James Leflang
CS325: Analysis of Algorithms
Portfolio Project
7 December 2020

Short Discussion on Sudoku

The Sudoku puzzle was introduced in 1979, and has become a popular addition to many printed publications. It features a n^2 x n^2 grid with n x n segments, and the rules are that any column, row, and segment can only contain one instance of some value from 1 to n^2. Sudoku has been proven to be NP-Complete in Yato and Seta (2003).[1]

Discussion on Verification Algorithms

Per Yato and Seta (2003), Sudoku can be reduced to the Latin Squares puzzle, to which solution verification algorithms are NP-Complete. Thus the proof for a NP-Complete solution verification algorithm for Sudoku is as such:
Given any solved puzzle of Sudoku, if a solution to Sudoku can be verified in polynomial time, then SUDOKU-VERIFY $\in$ NP. An algorithm would examine each row, column, and n x n square to insure there are unique instances of $\{i \mid 1 \leq i \leq n^2\}$. Since the described algorithm can verify a solved Sudoku in polynomial time, then SUDOKU-VERIFY $\in$ NP.
If SUDOKU-VERIFY $\leq_p$ LATIN-SQUARES, then SUDOKU-VERIFY $\in$ NP-Complete. Given that a Latin Squares puzzle is contrained by $\{i \mid 1 \leq i \leq n^2\}$, then the resulting n x n puzzle would be a subset of a Sudoku puzzle. Extending a Latin Squares to Sudoku puzzle entails that a Sudoku puzzle is a n^2 x n^2 Latin Square with $n \cdot (n \text{ x } n)$ sub-Latin Square puzzles. Thus, SUDOKU-VERIFY can be reduced to LATIN-SQUARES since Sudoku is a special Latin Square puzzle. If a solution to a Latin Square is a valid solution to the Sudoku, then SUDOKU-VERIFY can be reduced to LATIN-SQUARES. If a solution to a Sudoku is not a Latin Square, otherwise invalid, then SUDOKU-VERIFY cannot be reduced to LATIN-SQUARES.
Therefore we can conclude that SUDOKU-VERIFY $\in$ NP-Complete.

Determination of Runtime

The overall runtime of the verification algorithm is $O(n^2)$. This is due to the necessity of going through each of the n rows, columns, and subsquares, then linearly checking for the uniqueness of a cell given its relative position.

[1] Found in Graham Kendall, Andrew Parkes, and Kristian Spoerer. "A Survey of NP-Complete puzzles". In: *ICGA Journal* 31 (Mar. 2008), pp. 13–34. DOI: `10.3233/ICG-2008-31103`.