SAT-Based Approach to Solving Sudoku

Lakki Thapa, Supreme Chaudhary, Ashwot Acharya, Bishesh Bohora

November 28, 2025

Department of Mathematics · Kathmandu University

Outline

Introduction

Sudoku

Sudoku is a logic-based puzzle that originated in Japan in 1986 and gained widespread international popularity after 2005.A general sudoku puzzle consists of a $n^2 \times n^2$ grid further sectioned into smaller $n \times n$ squares.

How it's played? (9x9)

- Fill each row, column and squares (3x3) with digits from 0 to 9.
- None of the digits should be repeated in each row, column and square.
- Initially clues are provided i.e some digits are already filled.

SAT

The Boolean Satisfiability Problem (SAT)

A propositional logic formula, also called Boolean expression, is built from variables, operators AND (conjunction), OR (disjunction), NOT (negation), and parentheses. A formula is said to be satisfiable if it can be made TRUE by assigning appropriate logical values (i.e. TRUE, FALSE) to its variables. The Boolean satisfiability problem (SAT) is, given a formula, to check whether it is satisfiable. (Wikipedia)

Eg. $p \wedge (q \vee r)$ is satisfiable, with the assignment $p = \mathit{TRUE}, q = \mathit{FALSE}, r = \mathit{TRUE}$.

SAT Solvers

SAT solvers are algorithms or software tools that determine whether a given Boolean formula is satisfiable (i.e., if there exists an assignment of true/false values to variables that makes the formula true). If satisfiable then they provide the satisfying assignments.

Why SAT for Sudoku? Though there are multiple ways to figure out solution for this game like backtracking, We've chosen this because it,

- Aids generalization to sudoku puzzles of higher order.
- Proves unsatisfiablility in case of no solutions to given puzzle.
- We were interested in SAT problem and Applying SAT Solvers for a familiar game is a interesting as well.

A Formal Justification

- The **SAT problem** is **NP-complete** [Cook-Levin Theorem].
- A **generalized Sudoku** is also **NP-complete** [Yato and Seta, 2003].
- Being NP-complete, generalized Sudoku is in **NP** and can be **polynomial-time reduced** to any NP-hard problem, including SAT.
- Therefore, in the **worst-case scenario**, any Sudoku puzzle can be **transformed into a SAT problem**.
- For typical finite Sudoku puzzles, the size is much smaller than the general case, but this connection highlights that **efficient SAT-based solving is possible and formally justified**.

Problem Statement

Problem Statement

To investigate how Sudoku puzzles can be efficiently encoded into CNF and solved using SAT inference techniques, and to determine the optimal encoding and SAT solver for higher dimension Sudoku problems.

Objectives

Main Objective

- To Build a SAT-based Sudoku solver.

Specific Objectives

- To Represent the game sudoku in computer system
- To Convert the state of the Game sudoku to Conjective Normal Form
- To implement a sat solver for standard 9×9 sudoku
- To test and compare sat solvers
- To test for scalability in $n^2 \times n^2$ sudoku solver

Methodology

Methodology Overview

- Representation of sudoku requires encoding the game by encoding algorithms (eg boolean encoding , pair wise encoding etc)
- Encoded sudoku will be transformed to CNF form (boolean non-CNF expression may also expressed in CNF)
- Evaluation SAT solver algorithms (eg DPLL , CDCL)
- Implementation of required SAT solver algorithm for 9×9 sudoku (17 cues problems)
- Test the algorithm's effectiveness for higher dimension sudoku

Expected Results

Expected Outcomes

- To build a encoder that converts Sudoku to Conjective Normal Form
- To build a SAT solver for 9×9 sudoku
- Build a SAT solver for higher dimension Sudoku (e.g 16×16 sudoku or 25×25 sudoku)

Timeline

Timeline

Work	Duration
Literature review of SAT solvers for sudoku	Week 1
Study of various CNF encoding methods	week 2
Encoding implementation for 9×9 sudoku	week 3
Study of various SAT solver Algorithms	week 4
Implementation of SAT solver for 9×9 sudoku	week 5
Scability test for $n^2 \times n^2$ sodoku	week 6
Report writing and paper finalization	week 7

References

References

- Lynce, I., & Ouaknine, J. (2006). *Sudoku as a SAT Problem*.
- Simonis, H. (2005). *Sudoku as a Constraint Problem*.
- Bacchus, F. (2002). *Exploiting the Computational Tradeoff of More Reasoning and Less Searching*.
- Bacchus, F., & Winter, J. (2003). *Effective Preprocessing with Hyper-Resolution*.
- Colbourn, C. (1984). *The Complexity of Completing Partial Latin Squares*.
- Crawford, J., & Auton, L. (1993). *SAT Problem Cross-Over Experiments*.
- Robinson, J. A. (1965). *Resolution Principle*.
- Yato, T., & Seta, T. (2002). *Puzzle Complexity*.

Thank You