Solving Sudoku Using Artificial Intelligence

By, Eric Pass
What is Sudoku?

- Puzzle found in
  - News papers
  - Books
  - On toilet paper?

- The rules are
  - Each cell is assigned a value between 1 and N
  - Each row, column and unit must contain each number from 1 to N exactly once
Easy Right?

- Yeah...about as easy as figuring out a way to fold proteins into a specific shape

- Or as easy every other NP-Complete problem
So how do you solve Sudoku Puzzles?

- Slowly (I did just say it was NP-complete right)
- By using clever techniques like
  - Backtrack Searching
  - Forward Checking
  - Incorporate Heuristics such as
    - Minimum Remaining Values (MVR)
    - Probabilistic approaches

- Constraint Propagation with MVR!

- Or convert Sudoku into
  - An exact cover problems and use Donald Knuth’s “Algorithm X” and the dancing links technique
  - A Boolean Satisfiability problem and use a SAT solving algorithm like DPLL Algorithm
My Solver

- Implements the 5 different solving algorithms
  - Simple backtracking
  - Forward checking
  - Forward checking + MVR
  - Forward checking + Probabilistic Heuristics
  - Constraint Propagation

Which one is the most effective when trying to solve 9 by 9 puzzles of varying difficulty?
Simple Backtracking

- Why did this not work?
  - Did not use all the information available to it leading unnecessary checking

- What is good about it?
  - It is easy and intuitive to implement
  - It is the backbone of the more sophisticated algorithms
Forward Checking

- This is better but not great
  - Still 10% of each difficulty of puzzle goes unsolved
- What is good about it?
  - Starts to utilize more information to make more informed assignments
  - It is the starting point for the constraint propagation algorithm
Backtracking + MVR

- This is actually pretty good
  - Is 50% faster than simple backtracking
  - Solves the second highest amount of puzzles
- What is good about it?
  - Performs well in practice
  - Allows for pruning of sub trees
Backtracking + Probabilistic Heuristic

- Worse than MVR
  - Does not allow for the same kind of pruning
- What is good about it?
  - Still better than simple backtracking but worse than forward checking
Backtracking + Probabilistic Heuristic

- Performs great!
  - Does not allow for the same kind of pruning
  - Can solve all 9 by 9 and most 16 by 16
  - Only one which can fill in empty 25 by 25
- Why is it good?
  - The algorithm can cascade the effect of assigning a cell throughout the problem and realize if this is an invalid assignment sooner
Overall Comparison of Solvers

Percentage of Puzzles Solved per Solver

- Simple Backtracking
- Forward Checking
- MRV
- Probabilistic
- CSP

The CSP Solver Wins!