

Discrete Mathematics

Solving Propositional Constraints Automatically

Shin Hong

11 Sep, 2019

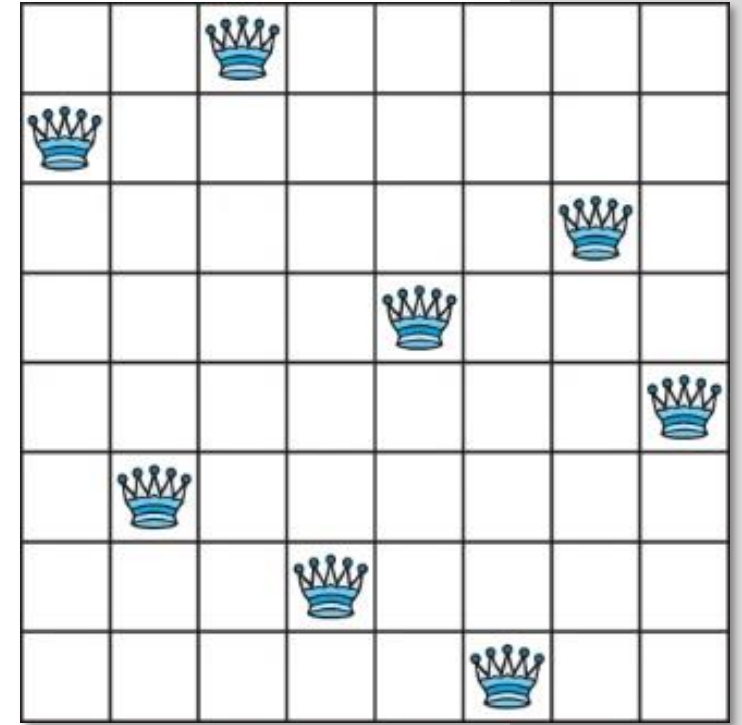
Application: N-Queen Problem

- Problem

- Place N Queens on a NxN grid, while not placing two Queens on the same vertical, horizontal or diagonal line

- Modeling

- Proposition $p_{i,j}$ indicates whether a Queen is placed at the i -row and at the j -th column



$$Q_1 = \bigwedge_{i=1..n} \bigvee_{j=1..n} p_{i,j}$$

$$Q_2 = \bigwedge_{i=1..n} \bigwedge_{j=1..n-1} \bigwedge_{k=j+1..n} \neg(p_{i,j} \wedge p_{i,k})$$

$$Q_3 = \bigwedge_{i=1..n} \bigwedge_{j=1..n-1} \bigwedge_{k=j+1..n} \neg(p_{j,i} \wedge p_{k,i})$$

$$Q_4 = \bigwedge_{i=2..n} \bigwedge_{j=1..n-1} \bigwedge_{k=1..min(i-1,n-j)} \neg(p_{i,j} \wedge p_{i-k,j+k})$$

$$Q_5 = \bigwedge_{i=1..n-1} \bigwedge_{j=1..n-1} \bigwedge_{k=1..min(n-i,n-j)} \neg(p_{i,j} \wedge p_{i+k,j+k})$$

$$Q_1 \wedge Q_2 \wedge Q_3 \wedge Q_4 \wedge Q_5$$

ional
nts
ically

SMT Solver Z3 (1/2)

3

- Microsoft Z3: <https://github.com/Z3Prover/z3>
 - Tutorial: <https://rise4fun.com/z3/tutorial>
- Input example: $(p \rightarrow q) \wedge (q \rightarrow \neg p) \wedge \neg(p \vee q)$

```
$ cat prop.txt
; prop.txt
(declare-const p Bool)
(declare-const q Bool)
(assert (and (=> p q)
             (=> q (not p))
             (not (or p q))))

(check-sat)
(get-model)
```

```
$ z3 prop.txt
sat
(model
  (define-fun q () Bool
    false)
  (define-fun p () Bool
    false)
)
```

Solving
Propositional
Constraints
Automatically

Discrete Math.

2019-09-12

SMT Solver Z3 (2/2)

4

- Microsoft Z3: <https://github.com/Z3Prover/z3>
 - Tutorial: <https://rise4fun.com/z3/tutorial>
- Input example: $\exists a \exists b \exists c (a > b + 2 \wedge a = 2c + 10 \wedge b + c \leq 1000)$

```
$ lia.txt
(declare-const a Int)
(declare-const b Int)
(declare-const c Int)
(assert (> a (+ b 2)))
(assert (= a (+ (* 2 c) 10)))
(assert (<= (+ c b) 1000))
(check-sat)
(get-model)
```

```
$ z3 lia.txt
sat
(model
  (define-fun b () Int
    0)
  (define-fun c () Int
    0)
  (define-fun a () Int
    10)
)
```

Solving
Propositional
Constraints
Automatically

Discrete Math.

2019-09-12

Application: Sudoku Puzzle (1/3)

5

- A Sudoku puzzle is represented as a 9x9 grid with nine 3x3 subgrids called subgrids
- Each cell has a number in 1 to 9
- The puzzle is solved by assigning a number to each cell so that every row, every column, and every of a subgrid contains each of the 9 numbers

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

Solving
Propositional
Constraints
Automatically

Discrete Math.

2019-09-12

Application: Sudoku Puzzle (2/3)

6

- $p(i, j, n)$ holds when row i and column j has n

- each cell is assigned with exactly one number

$$\bigwedge_{i=1}^9 \bigwedge_{j=1}^9 \bigwedge_{n=1}^8 \bigwedge_{m=n+1}^9 \neg(p(i, j, n) \wedge p(i, j, m))$$

$$\bigwedge_{i=1}^9 \bigwedge_{j=1}^9 \bigvee_{n=1}^9 p(i, j, n)$$

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

- each pre-assigned cell contains the given number

- E.g., $p(1,2,2) \wedge p(1,3,9) \wedge \dots \wedge p(2,4,5) \wedge \dots \wedge p(9,8,6)$

Solving
Propositional
Constraints
Automatically

Discrete Math.

2019-09-12

Application: Sudoku Puzzle (3/3)

7

- each row has every number between 1 and 9

$$\bigwedge_{i=1}^9 \bigwedge_{n=1}^9 \bigvee_{j=1}^9 p(i, j, n)$$

- each column has every number between 1 and 9

$$\bigwedge_{j=1}^9 \bigwedge_{n=1}^9 \bigvee_{i=1}^9 p(i, j, n)$$

- each subgrid has every number between 1 and 9

$$\bigwedge_{r=0}^2 \bigwedge_{s=0}^2 \bigwedge_{n=1}^9 \bigvee_{i=1}^3 \bigvee_{j=1}^3 p(3r + i, 3s + j, n)$$

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

Solving
Propositional
Constraints
Automatically

Discrete Math.

2019-09-12