Sudoku is Hard

Eve Routledge

Durham University

NP complete

Reduction

karp reduction

- Sudoku
- Latin Square
- Triangulated Tripartite Graph
- 3SAT
- SAT

Levin Cooke Theorem

Sudoku $\leq_p$ Latin Square

Lemma: Let S be a Sudoku problem with the following construction

$$S(i,j) = \begin{cases} 0 & \text{when } (i,j) \in S_l \\ ((i-1 \bmod n)n + \lfloor i-1/n \rfloor + j - 1) \bmod n^2 + 1 & \text{otherwise} \end{cases} \tag{1}$$

where $S_l = \{(i,j) | \lfloor i-1/n \rfloor = 0 \text{ and } (j \bmod n) = 1\}$. Then there exists an augmentation S' to complete the sudoku puzzle if and only if the square L such that $L(i,j/n) = (S'(i,j)-1)/n+1$ for all $(i,j) \in S_l$ is a Latin square.

1	2	3	4	5	6	7	8	9
4	5	6	7	8	9	1	2	3
7	8	9	1	2	3	4	5	6
2	3	4	5	6	7	8	9	1
5	6	7	8	9	1	2	3	4
8	9	1	2	3	4	5	6	7
3	4	5	6	7	8	9	1	2
6	7	8	9	1	2	3	4	5
9	1	2	3	4	5	6	7	8

S(i,j), n=3, blanks removed

	2	3		5	6		8	9
	5	6		8	9		2	3
	8	9		2	3		5	6
2	3	4	5	6	7	8	9	1
5	6	7	8	9	1	2	3	4
8	9	1	2	3	4	5	6	7
3	4	5	6	7	8	9	1	2
6	7	8	9	1	2	3	4	5
9	1	2	3	4	5	6	7	8

S(i,j)

1	4	7
4	7	1
7	1	4

$S_l(i,j)$

1	2	3
2	3	1
3	1	2

L(i,j)

Latin Square $\leq_p$ Triangulated Tripartite Graph

Triangulated Tripartite Graph $\leq_p$ 3 SAT

3 SAT $\leq_p$ SAT

Given a truth assignment t check each clause is satisfied, if all are satisfied return True else False, this algorithm is at most the length of C multiplied by the length of B. $O(BC)$ is polynomial, a polynomial verifier exists.

Given a SAT instance with the input sets of B and C. C is in conjunctive normal form (every clause set can be converted to an equivalent set in CNF form [**?**]) such that $\forall c \in C$ and for some $b_1, ..., b_n \in B$, $c = b_1 \vee b_2 \vee ... \vee b_n$. For each $c \in C$ with more than 3 literals we can transform these to a new set of clauses of length 3.

For $c = b_1 \vee b_2 \vee ... \vee b_n$ we introduce a new literal: a_1 to give $b_1 \vee b_2 \vee a_1$, $\bar{b_1} \vee a_1$, $\bar{b_2} \vee a_1$ and $a_1 \vee b_3 \vee ... \vee b_n$. Then $a_1 \vee b_3 \vee ... \vee b_n$ becomes $b_3 \vee b_4 \vee a_2$, $\bar{b_3} \vee a_2$, $\bar{b_4} \vee a_2$ and $a_1 \vee a_2 \vee b_5 \vee ... \vee b_n$. This continues at most $n/2$ times to give $a_1 \vee ... \vee a_{n/2}$ or $a_1 \vee ... \vee a_{n/2} \vee b_n$ if n is odd.

Because we can convert a clause larger than 3 into multiple clauses of at most 3 literals in linear time ($O(n/2 + n/4 + ...) = O(n)$) this means we can reduce SAT to 3SAT in polynomial time.

As SAT is NP-complete by the Cook-Levin Theorem, this proves 3SAT is NP-Complete. $\square$

Stochastic Methods

References