22/8/2015
Russell's cross, Section 1.

3 Ideas:
1) Art & design

2) Multidimensional cross, e.g. crossword puzzles, etc.

3) Reductions at finer grain, e.g., \( \text{2SAT} \in \text{TIME}(n^{1.5}) \)

Great complexity or NP-complete; meta-algorithmic aspects (i.e., input and output are representations of algorithms)

- Hard problem, circuits, program verification, circuit minimization, compiler design...

Breadth: circuits: introduced in Korkor & Sacharni, \( \log x \) on \( n \)-bit in \( 2^n \) size \( S(2^n) \)

For any function we know how to consume, keep lower bound issues.

(Notre Dame, 3).

Proof: circuits: \( C_{\text{up}} \), \( C_{\text{down}} \), \( 2^n \) upper bound

Exercise: prove that \( \text{Circuit Size}(R_{\text{up}}, 2^n) = \Omega(2^n) \)

Terms in \( \text{Circuit Size}(C_{\text{up}} \cap C_{\text{down}}) \leq \sum \text{Circuit Size} \)

\( \text{SAT} \): satisfiable \( \text{Circuit Size} \)

\( \text{K-SAT} \): \( k \)-circuit \( \leq \text{K-SAT} \)

Depth and unbounded fan-in circuit.

Randomly \( \lambda \), \( \nu \) or, \( \delta \), \( \alpha \), \( \beta \), \( \gamma \), \( \delta \), \( \alpha \), \( \beta \), \( \gamma \)

AC^0-equivalent \( \leftrightarrow \) polynomial-size \( \exists \)-uniform \( \forall \)-shredded memory

\( \text{Size}(C) \leq \text{Exp}(\sum C_{\text{up}}(\text{Circuit Size})) \), for \( \delta = \text{poly}\)

AJ & C:
First saw, sipper, bao, cai, Hasmed.

K-HSAT: when \( K \)-cut \( \leq 0 \), count \( 12 \times 12 \) for \( 12 \times 12 \) signs

Best cases: for \( K \)-SAT: \( 2^{n(1-\alpha(K))} \), Schoning, PPSZ, IMP.

Different techniques, same expression

For \( K \)-HSAT: IMP.

Random restricted \( \text{K-SAT} \)

\( \text{Restricted K-SAT} \) method: \( \text{P} \), \( \text{Q}, \text{R} \), \( \text{S}, \text{T} \)

Håstad Switching lemma: \( \text{let P \in K-CUT} \)

Decision tree: \( \text{ Knows the unique incorrect vertex}$ \text{for some vertices, e.g., P \in K-CUT} \)

Håstad Switching lemma: \( \text{P} \) \text{is too weak, possible dual vertex is } P$
For $c = 0$ and $s ightarrow z$, decision tree of $D(z) = T$. Decision tree of $D(z) = 0 \Rightarrow D(z) = 0$. If $D(z) = 0$, $D(z)$ is a canonical decision tree. For a CNF: Query on unsat clauses, then subqueries and simplify. Then next unsat clause.

If an unused empty clause, set $\alpha$ to true. If no unsat, output.

Most of switching lemma

Let $G$ be a $k$-CNF. Let $p$ be a random vector. Query on unsat clauses. Then $\Pr[\text{depth}(CD \cap G)] \leq \frac{1}{(1 - p)k^2} \leq \left(1 - \frac{1}{1 - p/k} \right)^{k^2}$

If $p \geq 1/k$, then each clause has decent chance of not disappearing. springs some round. For proving:

After one round, each clause loses at most $1/k$. After $i$ rounds, there are $\frac{n}{(2i)^k}$ clauses. After $i$ rounds, there $\frac{n}{(2i)^k} < \frac{1}{(1 - p/k)^k}$ rounds. So for rounds $n \leq \frac{1}{(1 - p/k)^k}$. For random $\mathbf{p}$, let $p = 1/k$. Divide input into $\log n$ rounds. Or use such partitions, even though are restricted.