On the possibility of faster SAT algorithms

Mihai Pătraşcu Ryan Williams

presented by Tomáš Gavenčiak

Problems

The examined problems:

CNF-SAT: Given a CNF fromula F (in form $\bigwedge_{i=1}^{m}(\bigvee_j (x_{i,j})))$ with n variables and m caluses, decide if F is satisfiable.

Best known algorithm: CNF-SAT in time $2^{n(1-1/O(\log(m/n)))}\text{poly}(m)$.

k-SAT: CNF-SAT with each clause of size at most k.

Best known algorithm: k-SAT in time $2^{n(1-1/\Theta(k))}\text{poly}(m)$.

Reduced to problems:

k**-DomSet:** Given a graph G, is there $D \subset V_G$, $|D| \leq k$ such that D dominates G? That is $\forall v \in V_G \exists d \in D : v \in N[d]$

Best known algorithm: k-DomSet in time $O(n^{k+o(1)})$.

2-SAT+2Clauses: Given a formula F with 2 arbitrary clauses C_1, C_2 such that $F - C_1 - C_2$ is 2-SAT, is F satisfiable?

Best known algorithm: 2-SAT+2Clauses in time $O(mn + n^2)$.

HornSAT+k**Clauses:** Given a formula F with k arbitrary clauses C_i such that $F - \bigcup C_i$ is HornSAT, is F satisfiable? In HornSAT, each clause contains at most one positive literal.

Best known algorithm: HornSAT+kClauses in time $O(n^k(n + m))$.

3-party disjointness: For $S_1, S_2, S_3 \subset [m]$, three parties are given (S_1, S_2), (S_1, S_3) and (S_2, S_3) respectively. Using a deterministic protocol, decide if $S_1 \cap S_2 \cap S_3 = \emptyset$.

Best known algorithm: 3-party disjointness communicating $O(m)$ bits, $O(km/2^k)$ bits for k-party disjointness.

d**-Sum:** Given a set of n integers, decide if there are d integers that sum to zero.

Best known algorithm: d-Sum in time $O(n^{\lceil d/2 \rceil}\text{poly}(\log n)$.

Solutions

Exponential time hypothesis: There is no algorithm for CNF-SAT running in time $O^*(2^{o(n)})$.

The **current goal** is an $O^*(2^{\delta n})$-time algorithm for CNF-SAT for some $\delta < 1$ (*improved algorithm).*

Hypothesis 1: There are $k \leq 3, \epsilon > 0$ such that k-DomSet is solvable in time $O(n^{k-\epsilon})$.

Theorem 1: Hypothesis 1 implies an improved algorithm for CNF-SAT.

Lemma 1: If there are $k \leq 3$ and function f such that k-DomSet is decidable in time $O(n^{f(k)})$, then CNF-SAT can be decided in time $O((m + k2^{n/k})^{f(k)})$.

Hypothesis 1': There are $k \leq 2, \epsilon > 0$ such that k-SetCover with n sets over ground set of size poly(log(n)) is solvable in time $O(n^{k-\epsilon})$.

Theorem 1': Hypothesis 1' implies an improved algorithm for CNF-SAT.

Hypothesis 2: For some $\epsilon > 0$ and $m = n^{1+o(1)}$, 2-SAT+2Clauses is solvable in time $O(n^{2-\epsilon})$.

Theorem 2: Hypothesis 2 implies an improved algorithm for CNF-SAT.

Hypothesis 2': There are $k \leq 2, \epsilon > 0$ such that HornSAT+kClauses is decidable in time $O((n + m)^{k-\epsilon})$.

Theorem 2': Hypothesis 2' implies an improved algorithm for CNF-SAT.

Hypothesis 4: There is a $d < N^{0.99}$ such that d-Sum on N numbers of $O(d \log N)$ bits can be decided in time $N^{o(d)}$.

Theorem 4: Hypothesis 4 implies an algorithm for 3-SAT running in time $2^{o(n)}$.

Hypothesis 5: There is a deterministic protocol for 3-party set disjointness communicating $o(m)$ bits and running in time $2^{o(m)}$.

Theorem 5: For every k, hypothesis 5 implies an algorithm for k-SAT running in time $O(1.74^n)$.