# Majority-3SAT (and Related Problems) in Polynomial Time

Shyan Akmal and Ryan Williams

## Definitions

Let $F$ be a Boolean formula on variables $\vec{x} = x_1, \ldots, x_n$. We denote by $\#\mathrm{SAT}(F)$ the number of assignments of $\vec{x}$ that satisfy $F$.

## Problems

- ThresholdSAT: for a constant $\varrho \in (0,1)$, is it true that $\frac{\#\mathrm{SAT}(F)}{2^n} \geq \varrho$?
- MajoritySat: is ThresholdSAT with $\varrho = \frac{1}{2}$.
- GtThresholdSAT: is ThresholdSAT with strict inequality.

We will usually consider the special case of these problems where each clause can contain at most $k$ literals. Then we call these problems Threshold-kSAT, Majority-kSat, and GtThreshold-kSAT respectively.

## Results

- **Theorem 1.1** (Main result). For every constant rational $\varrho \in (0,1)$ and every constant $k \geq 2$, there is a deterministic linear-time algorithm that given a $k$-CNF $F$ determines whether or not $\#\mathrm{SAT}(F) \geq \varrho \cdot 2^n$.
- **Theorem 1.4.** For all $k \leq 3$, GtMajority-kSAT is in P.
- **Theorem 1.5.** For all $k \geq 4$, GtMajority-kSAT is NP-complete.
- **Theorem 1.6.** Deciding MajoritySAT over $k$-CNFs with one extra clause of arbitrary width is in P for $k = 2$, NP-hard for $k = 3$, and PP-complete for $k \geq 4$.
- **Proposition 2.4.** Let $F$ be a CNF formula on $n$ variables, construed as a set of clauses. Suppose there is a $\varrho \in (0,1)$ and a subset $F'$ of the clauses of $F$ such that $F'$ contains $r \leq n$ variables and $\#\mathrm{SAT}(F') \leq \varrho \cdot 2^r$. Then $\#\mathrm{SAT}(F) \leq \varrho \cdot 2^n$.
- **Proposition 2.5.** Given a 1-CNF formula $F$ (i.e. $F$ is a conjunction of literals), the number of satisfying assignments to $F$ can be computed in linear time.
- **Theorem 3.1.** For every rational $\alpha \in (0,1)$, there is an $m \cdot \mathrm{poly}(1/\alpha)$-time algorithm that, given any 2-CNF formula $F$ on $n$ variables and $m$ clauses, decides whether $\#\mathrm{SAT}(F) \geq \alpha \cdot 2^n$ or not. Furthermore, when $\#\mathrm{SAT}(F) \geq \alpha \cdot 2^n$ is true, the algorithm outputs $\#\mathrm{SAT}(F)$, along with a decision tree representation for $F$ of $\mathrm{poly}(1/\alpha)$ size. The internal nodes are labeled by variables and leaves are labeled by 1-CNFs.
- **Theorem 4.1.** For every constant $\varrho \in [1/2, 1]$, we can decide in polynomial time if a given 3-CNF on $n$ variables has at least $\varrho \cdot 2^n$ satisfying assignments.
- **Theorem 4.3.** For every $\varepsilon \in (0, 1/2]$, we can decide in $\mathrm{poly}(1/\varepsilon, n)$ time if a given 3-CNF on $n$ variables has at least $(1/2 + \varepsilon) \cdot 2^n$ satisfying assignments. Moreover, given any 3-CNF with at least $(1/2 + \varepsilon) \cdot 2^n$ satisfying assignments, we can report the exact number of satisfying assignments.
- **Lemma 4.4.** Let $\varrho > 1/2$, and let $S$ be a maximal disjoint set of $k$-clauses in a $k$-CNF $F$. Suppose $F$ has at least $\varrho \cdot 2^n$ satisfying assignments. For all possible assignments $A$ to the variables of $S$, and for every induced 2-CNF $F_A$ obtained by assigning $A$ to $S$, $F_A$ must contain a maximal disjoint set of $(k-1)$-clauses of size less than $2^k |S| \ln(1/(\varrho - 1/2))$.
- **Lemma 4.5.** Let $\ell \in \{x, \neg x\}$ be a literal, and let

  $$S = \{(\ell \vee a_1 \vee b_1), \ldots, (\ell \vee a_t \vee b_t), (u \vee v \vee w)\}$$

  be a set of clauses with the following properties:

  - For all $i, j \in [t]$, $a_i$ and $b_j$ are literals from $2t$ distinct variables, all of which are different from $x$.
  - The literal $\ell$ does not appear in $(u \vee v \vee w)$ (however, $\neg \ell$ may appear in $(u \vee v \vee w)$).

  Then for all $t \geq 8$, $S$ has less than $2^{r-1}$ satisfying assignments, where $r$ is the total number of variables occurring in $S$.