ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 1, No. 1, 2006, *pp*. 03-10

New Lower Bound of Critical Function for (k, s)-SAT

Ping Gong [1], Daoyun Xu [2 +]

[1] Department of Mathematics, Science College, Guizhou University, Guiyang 550025, China

[2] Department of Computer Science, Computer College, Guizhou University, Guiyang 550025, China

(*Received Dec. 04, 2005, accepted Jan. 10, 2005*)

Abstract. $(k,s)-SAT$ is the propositional satisfiable problem restricted to instances where each clause has exactly k distinct literals and every variable occurs at most s times. It is known that there exits an exponential function f such that for $s \le f(k)$, all $(k,s)-SAT$ instances are satisfiable, but $(k, f(k)+1)-SAT$ is already NP-complete ($k \ge 3$). Therefore, we call the function $f(.)$ critical function. Exact values of $f(.)$ are only known for $k=3$ and $k=4$, and it's open whether $f(.)$ is computable. The best known lower and upper bounds on $f(k)$ are $\Omega(2^k/k)$ and $O(2^k/k^\alpha)$, where $\alpha = \log_3^4 - 1 \approx 0.26$, respectively. In this paper, analogous to the randomized algorithm for finding two coloring of $k-$ uniform hypergraph, we first present a similar randomize algorithm for outputting an assignment for a given formula. Based on it and by probabilistic method, we prove that, for every integer $k \ge 2$, each formula F in $(k,*)-CNF$ with at most $0.58\times\sqrt{\frac{k}{\ln k}}2^k$ clauses is satisfiable. In addition, by the Lovász Local lemma, we get a new lower bound of $f(k)$, $\Omega(\sqrt{\frac{k}{\ln k}}2^k/k)$, which improves the result $\Omega(2^k/k)$.

Keywords: $(k,s)-SAT$, NP-complete, randomize algorithm, probabilistic method.

1. Introduction

A literal is a propositional variable or a negated propositional variable. A clause C is a disjunction of literals, $C=(L_1,\cdots,L_m)$, or a set of literals, $\{L_1,\cdots,L_m\}$. A formula F in conjunctive normal form (CNF) is a conjunction of clauses, $F=(C_1\wedge\cdots\wedge C_n)$, or a set of clauses, $\{C_1,\cdots,C_n\}$. $|C|$ is the number of literal in the clause C. $\mathrm{var}(F)$ is the set of variables occurring in the formula F and $lit(F)$ is the set of literals over $\mathrm{var}(F)$. It was observed by Tovey [1] that all formulas in $(3,3)-CNF$ are satisfiable, and the satisfiability problem restricted to $(3,4)-CNF$ is already NP-complete. There was a generalization in Kratochvil's work, where it is shown that for each $k \ge 3$, there is some integer $s=f(k)$, such that

- all formulas in $(k,s)-CNF$ are satisfiable, and
- $(k,s+1)-SAT$, the satisfiability problem restricted to $(k,s+1)-CNF$, is already NP-complete.

Therefore the critical function $f(k)$ can be defined by the equation

$$f(k)=\max\{s:(k,s)-CNF\cap UNSAT=\emptyset\}.$$

From [1], it follows that $f(3)=3$ and $f(k)\ge k$ for $k>3$. However it is open whether f is computable. The upper and lower bounds for $f(k)$, $k=5,\cdots,9$, have been obtained in [2, 3, 4]. For larger values of k, the best known lower bound, a consequence of Lovász Local Lemma, is due to Kratochíl [5].

$$f(k)\ge\lfloor 2^k/ek\rfloor.$$

The best known upper bound, due to Savickỳ and Sgall [6], is given by

$$f(k)\le O(2^k/k^\alpha).$$

[+] Corresponding author. Tel.: +86-851-3627649; fax: +86-851-3621102.
E-mail address: dyxu@gzu.edu.cn.

where $\alpha = \log_3^4 - 1 \approx 0.26$.

The probabilistic method is not about probabilistic algorithms[7, 8], which give the right answer with high probability but not with certainty, nor about Monte Carlo methods, which are simulations relying on pseudo-randomness.

The probabilistic method is a non-constructive method primarily used in combinatorics and pioneered by Paul Erdős, for proving the existence of a prescribed kind of mathematical object. This method has now been applied to other areas of mathematics such as computer science, number theory, linear algebra, and real analysis. Common tools used in the probabilistic method include Markov's inequality, the Chernőff bound, and the Lovász Local lemma and so on.

In this paper, analogous to the randomized algorithm for finding two coloring of k -uniform hypergraph[3, 9, 10, 11, 12], we present one for outputting an assignment for formulas in $(k,*)-CNF$. And then, for each formula, we can create a probability space, the samples of which are the assignments derived randomly from the randomized algorithm. We show, in this kind of probability space, for each formula, the probability of formula with a truth assignment is positive. Therefore, by the probabilistic method, the formula is satisfiable by such assignment. Besides, based on the randomized algorithm, we get, by applying the Lovász Local lemma, the lower bound of $f(k)$, $\Omega(\sqrt{\frac{k}{\ln k}} 2^k / k)$, which improves the previous result $\Omega(2^k / k)$.

2. Basic notations

Let $F = \{C_1, \cdots, C_m\}$ be a CNF formula with variables set $\mathrm{var}(F) = \{x_1, \cdots, x_n\}$. An assignment to a formula F is a map $\tau : \mathrm{var}(F) \longrightarrow \{0,1\}$ We define $\tau(\neg x) := 0$ if $\tau(x) := 1$ and $\tau(\neg x) := 1$ otherwise. A variable x occurs in a clause C if $x \in C$ or $\neg x \in C$. Further, for $C \in F$ we define $\tau(C) := \max_{x \in C} \tau(x)$ and $\tau(F) := \min_{C \in F} \tau(C)$. A formula F is satisfied by an assignment τ if $\tau(F) = 1$. A formula F is satisfiable if there exits a truth assignment which satisfies F; otherwise F is called unsatisfiable. $ASS(F)$ is the set of all assignments of F on $\mathrm{var}(F)$;

For $\tau \in ASS(F)$, $TC_\tau(F) = \{C \in F : \tau(C) = 1\}$; $FC_\tau(F) = \{C \in F : \tau(C) = 0\}$.

For $C \in F$, $TV_\tau(C) = \{x \in \mathrm{var}(C) : \tau(x) = 1 \ \ if \ \ x \in C; \ \ \tau(\neg x) = 1 \ \ if \ \ \neg x \in C\}$; $FV_\tau(C) = \mathrm{var}(C) - TV_\tau(C)$.

For $x \in \mathrm{var}(F)$, $TC_\tau(x) = \{C \in TC_\tau(F) : x \in TV_\tau(C)\}$; $FC_\tau(x) = \{C \in FC_\tau(F) : x \in \mathrm{var}(C)\}$.

Besides, we use $F(C,\tau)$ to denote the event "clause C is unsatisfiable in assignment τ"; and $T(C,\tau)$ is reverse to $F(C,\tau)$.

We use SAT to denote the class of all satisfiable formulas and $UNSAT$ to the class of unsatisfiable formulas. We also use $k-CNF$ to denote the class of CNF formulas where the length of each clauses is no more than k. $(k,s)-CNF$ is a class of the conjunctive normal form formula where each clause' length is exactly k and the occurrence number of each variable is at most s. For conveniency, we define $(k,*)-CNF = \bigcup_{i=1}^{+\infty} (k,i)-CNF$.

3. The randomize algorithm

In this section, a useful tool, which will be applied following, is the probabilistic method. Roughly speaking, the method works as follows: Trying to prove that a structure with certain desired properties exists, one defines an appropriate probability space of structures and then shows that the probability of an object selected uniformly from the space satisfying the desired properties is positive or falsifying them is less than 1.

For a $(k,*)-CNF$ formula F with variables set $\mathrm{var}(F) = \{x_1, \cdots, x_n\}$, we first define two following functions.

The function $ord : \mathrm{var}(F) \longrightarrow [0,1]$. For each $x \in \mathrm{var}(F)$, the value of $ord(x)$ is randomly picked from $[0,1]$ independently. The purpose of function ord is to give a random order among variables set $\mathrm{var}(F)$ (please note that with probability 1, no two variables were assigned same values).

The function $b : \mathrm{var}(F) \longrightarrow \{0,1\}$. For each $x \in \mathrm{var}(F)$, $b(x) = 1$ with probability p and $b(x) = 0$ with probability $1-p$. p is a parameter the value of which will be presented properly later.

The algorithm:

Input: a $(k,*)-CNF$ formula F with variables set $\mathrm{var}(F)=\{x_1,\cdots,x_n\}$.

Output: an assignment τ^* for formula F .

Step 1. Generate a random assignment τ_0 by choosing $\tau_0(x)$ to be 0 or 1 with probability 1/2, independently for each variable $x\in \mathrm{var}(F)$.

Step 2. For each $x\in \mathrm{var}(F)$, we get the values of $ord(x)$ and $b(x)$ independently. Let $x_1,\cdots,x_n$ be an increasing variables sequence ordered in values of $ord(.)$. Next $n=|\mathrm{var}(F)|$ steps are reassignment steps based on the values of $ord(.)$ and $b(.)$.

Step 3. If $FC_{\tau_0}(x_1)\neq\varnothing$ and $b(x_1)=1$, then flip the value of x_1 . Otherwise, go to next step. Let the resulting assignment be τ_1 . (Please note if the value of x_i was not flipped, then $\tau_i=\tau_{i-1}$ for $i=1,\cdots,n$.)

Step 4. If $FC_{\tau_0}(x_2)\cap FC_{\tau_1}(F)\neq\varnothing$ and $b(x_2)=1$, then flip the value of x_2 . Otherwise, go to next step. Let the resulting assignment be τ_2 .

... …

Step n+2. If $FC_{\tau_0}(x_n)\cap FC_{\tau_{n-1}}(F)\neq\varnothing$ and $b(x_n)=1$, then flip the value of x_n . Let the resulting assignment be τ^* , output τ^* and stop the algorithm.

Remark. The purpose of defining the functions of $ord(.)$ and $b(.)$ is to control the reassignment steps. More precisely, they can avoid the situation of some previously-processed variables reassigning their values again. The notation $FC_{\tau_0}(x_i)\cap FC_{\tau_{i-1}}(F)\neq\varnothing$ means there exit some clauses, which contain the variable x_i making their falseness in the assignment τ_0 , are still false in the new assignment τ_{i-1} .

For $F\in(k,*)-CNF$, a random assignment $\tau_0\in ASS(F)$ is generated in Step 1, and by values of $ord(.)$ and $b(.)$, the algorithm processes reassignments from Step 3 to Step n+2. At the final step, the result assignment τ^* is outputted concerned with the values of $ord(.)$ and $b(.)$.

Thus for formula F , we define a dual structure (F,S_F) . where

$$S_F=\{\tau^*:\tau^*\ \textit{is derived randomly from the algorithm}\}\ .$$

For the structure, we want to know, whether there is an assignment $\tau^*\in S_F$ satisfying F . To solve this question, Based on the probabilistic method, a proper probability space $\Omega_F=(S_F,\Re,P)$ is defined firstly, where $\Re$ is a σ-algebra on S_F and P is a measure on $\Re$ the values of which are concerned with the values of $ord(.)$ and $b(.)$. To prove there exits an assignment satisfying formula, we just need to prove, in the probability space Ω_F , for an assignment τ^* picked uniformly from S_F , the probability of the assignment τ^* failing satisfying formula F is less than 1. Formally,

$$\Pr[\tau^*(F)=0]=\Pr[FC_{\tau^*}(F)\neq\varnothing]<1\ .$$

Based on above discussions, we begin estimating the probability of the event that there exits a clause $C\in F$ which is false in the random assignment $\tau^*\in S_F$. We have two following cases based on whether or not at least one variable whose value was reassigned during the reassignment steps.

Case 1. C is false in both τ_0 and τ^* , that is the value of all the variables in C are not flipped during whole reassignment steps. We say that event $A(C)$ takes place. Formally,

$$A(C)=F(C,\tau_0)\wedge F(C,\tau^*)\ .$$

In fact, it is the event of $b(x)=0$ for each $x\in \mathrm{var}(C)$ which triggers $A(C)$.

Case 2. C is true in τ_0 , but becomes false during the reassignment steps. That is, in some reassignment steps, every true literal of C has been changed false. Let x be the last variable, the literal of which is true in C in assignment τ' , to change its value. There must be at least one clause $C'\neq C$ such that $x\in \mathrm{var}(C')\cap \mathrm{var}(C)$, $C'\in FC_{\tau_0}(x)$ and C' was continued being false until x was considered and $b(x)=1$, And then C become false in the result assignment after flipping the value of x because of C' . we denote by $B(C,C')$ the event of C' making C false. Formally,

$B(C,C') = (\exists x \in \text{var}(C) \cap \text{var}(C') : x \in TV_{\tau'}(C) \wedge b(x) = 1) \wedge (\forall y \in \text{var}(C) \setminus \{x\} : y \notin TV_{\tau'}(C)) \wedge (C' \in FC_{\tau'}(F) \cap FC_{\tau_0}(x))$.

We have following lemma.

Lemma 1. If $C \in F$ is false in τ^*, then at least one of $A(C)$ or $B(C,C')$ takes place for some $C' \in F \setminus \{C\}$.

Thus, to bound the probability that there are some false clauses in τ^*, it is enough to bound the probabilities of the events: $\exists C \in F : A(C)$ and $\exists C, C' \in F : B(C,C')$. The following three claims will help us estimate the probabilities of these events.

Claim 1. $\Pr[A(C)] = 2^{-k}(1-p)^k$.

Proof: Since C is false in both τ_0 and τ^*, that is the value of all the variables in C are not flipped during whole reassignment steps, we have

$$A(C) = F(C,\tau_0) \wedge (\forall x \in \text{var}(C) : b(x) = 0).$$

Therefore, $\Pr[A(C)] = 2^{-k}(1-p)^k$ is correct.

We also have following claim.

Claim 2. If $| TV_{\tau_0}(C) \cap FV_{\tau_0}(C') | > 1$, then $\Pr[B(C,C')] = 0$.

Proof: Suppose $TV_{\tau_0}(C) \cap FV_{\tau_0}(C') = \{x, x'\}$ and $ord(x) > ord(x')$. Then the value of x' will be flipped before the value of x will be done. Let the result assignment be τ' after flipping the value of x'. As a result, $C' \in FC_{\tau_0}(x)$, but $C' \notin FC_{\tau'}(F)$. Therefore, C' can not make C false by the algorithm.

Supposing $TV_{\tau_0}(C) \cap FV_{\tau_0}(C') = \{x\}$, by the definition of $B(C,C')$, we have two following conditional events which trigger $B(C,C')$:

- $\Phi^1(S,C,C') \equiv T(C,\tau_0) \wedge F(C - S - \{x, \neg x\}, \tau_0) \wedge T(S,\tau_0) \wedge (\forall x \in (\text{var}(S) \cup \{x\}) : b(x) = 1)$.

Where $S \subseteq C - \{x, \neg x\}$. The event means, C and S, some sub-clause of C, are true in assignment τ_0 but not for $C - S - \{x, \neg x\}$ and each variable in $\text{var}(S) \cup \{x\}$ owns qualification for reassignment. Obviously, if $B(C,C')$ holds, then the event $\Phi^1(S,C,C')$ must hold for some sub-clause $S \subseteq C - \{x, \neg x\}$.

- $\Phi^2(S,C,C') \equiv (\forall x' \in \text{var}(S), ord(x') < ord(x)) \wedge (\forall x' \in \text{var}(C') - \{x\}) : (ord(x') > ord(x) \vee (b(x') = 0))$.

The event means, the variable x is the last one in $\text{var}(C)$ performing reassignment and every variable in $\text{var}(C') - \{x\}$ processes reassignment after x does, or doesn't own qualification at all.

Thus, if we define $B^*(C,C') = \Phi^1(S,C,C') \wedge \Phi^2(S,C,C')$, then the event $B^*(C,C')$ triggers $B(C,C')$ happening. Thus, to bound the probability of the event $B(C,C')$, we just only need to do it for the event $B^*(C,C')$.

Claim 3. If $TV_{\tau_0}(C) \cap FV_{\tau_0}(C') = 1$, then $\Pr[B^*(C,C')] \le 2^{-2k-1} p$.

Proof: Suppose $TV_{\tau_0}(C) \cap FV_{\tau_0}(C') = \{x\}$ and $ord(x) = w$. $\Pr[F(C',\tau_0)] = 2^{-k}$;

$$\Pr[F(C - S - \{x,\neg x\}, \tau_0) \cap T(S,\tau_0)] = 2^{1-k}; \quad \Pr[\forall x' \in \text{var}(S) \cup \{x\} : b(x') = 1] = p^{|S|-1};$$

$$\Pr[\forall x' \in \text{var}(S) : ord(x') \le ord(x)\} = w^{|S|}; \quad \Pr[\forall x' \in \text{var}(C') - \{x\} : ord(x') \le ord(x) \vee b(x') = 0] = (1-wp)^{k-1}.$$

Thus, by the definition, $\Pr[B^*(C,C')] = 2^{-2k+1} p^{|S|-1} w^{|S|} (1-wp)^{k-1}$. On integrating over w and summing over all S, we obtain

$$\begin{aligned}
\Pr[B(C,C')] &\le 2^{-2k-1} \sum_{l=0}^{k-1} \binom{k-l}{l} p^{l+1} \int_0^1 w^l (1-wp)^{k-1} dw \\
&= 2^{-2k-1} p \int_0^1 (1-wp)^{k-1} [\sum_{l=0}^{k-1} \binom{k-l}{l} p^l w^l] dw \\
&= 2^{-2k-1} p \int_0^1 (1-wp)^{k-1} (1+wp)^{k-1} dw \\
&= 2^{-2k-1} p \int_0^1 (1-(wp)^2)^{k-1} dw \\
&\le 2^{-2k-1} p \int_0^1 1 dw \\
&= 2^{-2k-1} p .
\end{aligned}$$

From Claim 1, we have

$$\Pr[\exists C \in F : A(C)] \le | F | \times 2^{-k} (1-p)^k .$$

From Claim 2, 3, we have

$$\Pr[\exists C, C' \in F : B(C, C')] \le |F|^2 \times 2^{-2k+1} p .$$

By lemma 1, for the random assignment $\tau^* \in S_F$, we have

$$\Pr[FC_{\tau^*}(F) \ne \varnothing] \le |F| \times 2^{-k}(1-p)^k + |F|^2 \times 2^{-2k+1} p . \qquad (1)$$

Now we just need to search some conditions to satisfy above inequality less than 1. Then the conclusion of the formula F owning a true assignment can be gotten. Thus we have following theorem.

Theorem 1. Let F is a $(k,*)-CNF$ formula with at most $0.58 \times \sqrt{\frac{k}{\ln k}} 2^k$ clauses, then for all $k \ge 2$, F is a satisfiable formula.

Proof: Let $|F| = l2^k$. Then the inequality of (1) becomes

$$\Pr[FC_{\tau^*}(F) \ne \varnothing] \le l(1-p)^k + 2l^2 p .$$

For $0 < \varepsilon < 1$, set $l = (1-\varepsilon)\sqrt{\frac{k}{\ln k}}$, $p = (1/2)\ln k / k$. And then we have

$$\begin{aligned}\Pr[FC_{\tau^*}(F) \ne \varnothing] &\le l(1-p)^k + 2l^2 p \\ &= (1-\varepsilon)\sqrt{\frac{k}{\ln k}}(1-\frac{\ln k}{2k})^k + (1-\varepsilon)^2 \\ &= (1-\varepsilon)[1 + (\sqrt{\frac{k}{\ln k}}(1-\frac{\ln k}{2k})^k - \varepsilon)] \qquad (2)\end{aligned}$$

Set $g(k) = \sqrt{\frac{k}{\ln k}}(1-\frac{\ln k}{2k})^k$. $g(k)$ is a decreasing function on k. Since $g(2) < 1.15$, $g(k) < 1.15$ is correct for all $k \ge 2$. By analysis (2), set $\varepsilon = 0.42$ which is the minimal number satisfying the inequality of (2)<1 forany $k \ge 2$. Therefore $l = (1-\varepsilon)\sqrt{\frac{k}{\ln k}} = 0.58 \times \sqrt{\frac{k}{\ln k}}$ is the maximal number satisfying the inequality of (2)<1 for any $k \ge 2$. By the probabilistic method, we have proven the theorem.

4. The lower bound of f(k)

Let F be $(k,*)-CNF$ formula, if the parameter s, the maximal occurring number of variables in F, is not more than f(k), then each formula in (k, s)−CNF is satisfiable. To bound $f(.)$, we introduce a useful parameter of F : *overlap*.

For each clause $C \in F$, the overlap of C, denoted by d_c, is defined by $d_c = |\{C' \in F \setminus \{C\} : \mathrm{var}(C) \cap \mathrm{var}(C') \ne \varnothing\}|$. The overlap of F is the maximal d_c for $C \in F$, denoted by d. We first present the upper bound of d within which every $(k,*)-CNF$ formula is satisfiable. Then we conclude the lower bound of $f(k)$ based on the relation between parameters s and d.

We will apply a special case of Lovász Local lemma, which shows a useful sufficient condition for simultaneously avoiding a set $A_1, A_2, \cdots, A_N$ of "bad" events:

Theorem 2. Suppose events $A_1, A_2, \cdots, A_N$ are given. Let $S_1, S_2, \cdots, S_N$ be subsets of $[N] = \{1, 2, \cdots, N\}$ such that for each i, A_i is independent of the events $\{A_j : j \in ([N] - S_i)\}$. Suppose that $\forall i \in [N]$: (1) $\Pr[A_i] < 1/2$, and (2) $\sum_{j \in S_i} \Pr[A_j] \le 1/4$. Then $\Pr[\bigwedge_{i \in [N]} (\neg A_i)] > 0$.

Remark. Often, each $i \in N$ will be an element of at least one of the sets S_j ; Therefore, $\Pr[A_i] \le \sum_{i \in S_j} \Pr[A_j]$.

Thus, it clearly suffices to only verify condition (2) of Theorem.

Suppose F is a $(k,*)-CNF$ formula with overlap $d = \lambda 2^k$. Let τ^* be the random assignment obtained by above algorithm. By lemma 1, if we can simultaneously avoid the following events, then τ^* will be a valid truth assignment of F :

$$\{A(C) : C \in F\} \cup \{B(C, C') : C, C' \in F\} .$$

In Claim 2 and 3, we observed that the event $B(C,C')$ holds only if $|TV_{\tau_0}(C)\cap \mathrm{var}(C')|=1$ and the event $B^*(C,C')$ holds. Thus, it is enough if we can simultaneously avoid the following two types of events:

(a) Type 1 events: $\{A(C): C\in F\}$.

(b) Type 2 events: $\{B(C,C'): C,C'\in F \wedge |TV_{\tau_0}(C)\cap FV_{\tau_0}(C')|=1\}$.

We call above two types events *bad* events.

For a bad event B. Let $S(B)$ be the set of all bad events at least one of whose argument has a non-empty intersection with at least one argument of B. Formally, $S(B)=\{A(C'): C'\in F \wedge \mathrm{var}(C')\cap \mathrm{var}(C)\neq\varnothing\}$ if $B=A(C)$; $S(B)=\{B(C_0,C_0'): C_0,C_0'\in F\wedge((\mathrm{var}(C_0)\cup \mathrm{var}(C_0'))\cap(\mathrm{var}(C)\cup\mathrm{var}(C'))\neq\varnothing)\}$ if $B=B(C,C')$. Thus, as discussed above, B is independent of any events outside $S(B)$. Thus to apply Theorem 2, we need to bound the sum of probabilities of events in $S(B)$. To do these, we will first bound the number of events of each type in $S(B)$, then we will use Claim 1 and Claim 3 to bound their probabilities.

Claim 4. For all bad events B, $S(B)$ has at most $2d$ events of type 1 and at most $4d^2$ events of type 2.

***Proof*:** Suppose C and C' are the arguments of B (we will take $C = C'$ if B is a type 1 event). The only events of type 1 that are in $S(B)$ correspond to clauses that intersect either C or C'. There are at most $2d$ such clauses by the definition of overlap;

For a type 2 events with arguments (C_0,C_0') to be in $S(B)$, at least one of C_0 and C_0' must intersect at least one of C and C'; furthermore, C_0 and C_0' must themselves intersect. It follows that there are at most $4d^2$ possibilities for (C_0,C_0').

Now, we can apply Theorem 2 to these bad events.

Claim 5. Suppose $d=\lambda 2^k$, where $\lambda\le 0.1\sqrt{\frac{k}{\ln k}}$ and $k\ge 2$, then for any bad events B, $\sum_{B'\in S(B)}\Pr[B']\le 1/4$.

***Proof*:**

$$\sum_{B'\in S(B)}\Pr[B']\le 2d\times 2^{-k}(1-p)^k+4d^2p\ =2\lambda(1-p)^k+8\lambda^2 p\ . \tag{3}$$

If $p=(1/2)(\ln k)/k$, $\varepsilon>0$ and $\lambda=1/4(1-\varepsilon)\sqrt{\frac{k}{\ln k}}$, the equation

$$(3)=1/4(1-\varepsilon)(2g(k)+(1-\varepsilon))$$

Please note that $g(k)=\sqrt{\frac{k}{\ln k}}(1-\frac{\ln k}{2k})^k$ and it is a decreasing function. It is enough to just choose a proper ε to make

$$(1-\varepsilon)(2g(k)+(1-\varepsilon))<1\,. \tag{4}$$

We choose $(1-\varepsilon)=0.37$ which is maximum number satisfying the inequality of (3) when $k\ge 2$. Since $g(k)$ is a decreasing function, for any $k\ge 2$, Therefore, when

$$\lambda=1/4(1-\varepsilon)\sqrt{\frac{k}{\ln k}}\le 1/4\times 0.37\sqrt{\frac{k}{\ln k}}\le 0.1\times\sqrt{\frac{k}{\ln k}}\ ,$$

The inequality of (4) < 1/4 is always correct. Thus, the claim is correct.

We have thus established that condition (2) of Theorem 2 holds if d is chosen suitably. As remarked before, this implies that condition (1) holds as well. Thus, by the theorem 2, we get following theorem.

Theorem 3. For a formula F in $(k,*)-CNF$, if the overlap of it is at most $0.1\times\sqrt{\frac{k}{\ln k}}\times 2^k$, then F is a satisfiable formula.

Now, we study the connection between the maximal occurring number of variable of F and the overlap of F by following lemma.

Lemma 2. For a formula $F\in(k,*)-CNF$, suppose the parameters of s and d are the maximal occurrence number of variable in F and the overlap of F, respectively. Then $s\ge d/k+1$.

***Proof*:** Let a clause $C_0\in F$, $|C_0|=k$. For any variable x in $\mathrm{var}(C_0)$, $C(x)=\{C\in F: x\in\mathrm{var}(C)\}$. Obviously,

$C(x) \subseteq \{C \in F : \text{var}(C) \cap \text{var}(C_0) \neq \emptyset\} \cup \{C_0\}$ and $\bigcup_{x \in \text{var}(C_0)} C(x) = \{C \in F : \text{var}(C) \cap \text{var}(C_0) \neq \emptyset\} \cup \{C_0\}$.

Thus

$$|\bigcup_{x \in \text{var}(C_0)} C(x)| = |\{C \in F : \text{var}(C) \cap \text{var}(C_0) \neq \emptyset\} \cup \{C_0\}| .$$

That is,

$$d = |\bigcup_{x \in \text{var}(C_0)} C(x)| - 1 . \quad (5)$$

For each two variables $x, x' \in \text{var}(C_0)$, $\{C_0\} \subseteq C(x) \cap C(x_0)$. By the principle of inclusion and exclusion,

$$|\bigcup_{x \in \text{var}(C_0)} C(x)| = \sum_{x \in \text{var}(C_0)} |C(x)| - \sum_{i=2}^{k} (-1)^i \sum_{2 \le i_1 \le \cdots \le i_i \le k} |C(x_{i_1}) \cap C(x_{i_2}) \cdots \cap C(x_{i_i})| . \quad (6)$$

Since

$$\begin{aligned}(6) &\le \sum_{x \in \text{var}(C_0)} |C(x)| - \sum_{i=2}^{k} (-1)^i \sum_{2 \le i_1 \le \cdots \le i_i \le k} 1 \\ &= \sum_{x \in \text{var}(C_0)} |C(x)| - \sum_{i=2}^{k} (-1)^i \binom{k}{i} \\ &= \sum_{x \in \text{var}(C_0)} |C(x)| - (k-1) \\ &\le k \cdot s - (k-1) .\end{aligned}$$

By (5), we have $d \le k \cdot s - (k-1) - 1 = k \cdot s - k$. Therefore, $s \ge d / k + 1$ is correct.

By Theorem 3 and lemma 2, we have following theorem.

Theorem 4. $f(k) = \Omega(\sqrt{\frac{k}{\ln k}} 2^k / k)$.

5. Conclusion

The key gadgets applied in this paper are probabilistic method and Lovász Local lemma. Based on them, in section 3, we first got the result about the $(k,*)-CNF$ formulas' satisfiability and the number of clauses. In section 4, we first study the close connection between $(k,*)-CNF$ formulas' satisfiability and the parameter of overlap. And then, by the lemma 2, we got the new lower bound of critical function $f(k)$ for $(k,s)-SAT$.

6. References

[1] C. A. Tovey, A Simplified NP-complete Satisfiability Problem, Discr. Appl. Math., 8(1984)1, 85-89.

[2] O. Dubois, On the r, s-SAT Satisfiability Problem and a Conjecture of Tovey, Discr. Appl. Math., 26(1990)1, 50-60.

[3] P. Berman, M. Karpinski and A. D. Scott, Approximation Hardness and Satisfiability of Bounded Occurrence instances of SAT, Technical Report TR03-022, Electronic Colloquium on Computational Complexity (ECCC), 2003.

[4] J. Kratochvíl, P. Savickỳ, and Z. Tuza, One More Occurrence of Variables Make Satisfiability Jmup from Trivial to NP-complete. Acta Informatica, 30(1993), 397-403.

[5] P. Savickỳ and J. Sgall, DNF Tautologies with a Limited Number of Occurrences of Every Vairiable, Theoret. Comput. Sci,, 238(1-2)(2000), 495-498.

[6] N. Alon and J. H. Spencer, The Probabilistic Method, Wiley-Interscience Series, John Wiley & Sons. Inc., NewYork, 1992.

[7] J. H. Spencer, Ten Lectures on the Probabilistic Method, SIAM, Philadephia, 1987. (Second edition published in 1994.)

[8] C. Mcdiarmid, A Random Recolouring Method for Graphs and Hypergraphs, Combinatorics, Probability and Computing, 2(1973), 363-365.

[9] C. Mcdiarmid, Hypergraph colouring and the Lovász Local Lemmma, Discrete Mathematics, 167/168(1997), 481-486.

[10] J. Beck, On 3-chomatic Hypergraphs, Discrete Mathematics, 24(1978), 127-137.

[11] J. Radhakrishnan and A. Srinivasan, Improved Bounds and Algorithms for Hypergraph2-colouring, Rand, Structures and Algorithms, 16(2000), 4-32.