A satisfiability algorithm using LLL

Input: A k-SAT formula $\bigvee_{i=1}^{m} C_i$ (k-CNF) (ith clause) with l input Boolean variable $x_1, \dots, x_l$ s.t. k is an even constant and each variable appears in no more than $2^{\alpha k}$ clauses for a sufficiently small constant $\alpha > 0$.

Goal: Design an algorithm that finds a satisfying assignment for $\bigvee_{i=1}^{m} C_i$ in expected time that is polynomial in m.

Ideas: Consider a two-phase algorithm by using LLL, where Phase I breaks the original problem into smaller subproblems. Then phase II solves the subproblems independently by an exhaustive search.

The first pass

- $Fi(C_i) \stackrel{def}{=}$ the variables in C_i which have been assigned values (at this moment).
- A clause C_i is called dangeous if
  1. $|Fi(C_i)| = k/2$. 2. C_i is Not yet satisfied by variables in $Fi(C_i)$.

- **Phase I:** Initially, all variables $x_1, \dots, x_\ell$ are not fixed.

  For $i=1$ **to** ℓ **do If** $x_i \in$ a dangeous clause **then** do nothing.
  else toss a fair coin Y_i to assign x_i value in $\{0,1\}$.

- A clause C_i is called surviving if C_i is not satisfied by the variables in $Fi(C_i)$.

- Note that if C_i is surviving then $|Fi(C_i)| \le \frac{k}{2}$.
- Note that if C_i is dangeous then $|Fi(C_i)| = \frac{k}{2}$.
- If C_i is dangeous then C_i is surviving.

The Second pass

- Variables in $\{x_1,\dots,x_\ell\}\setminus\bigcup_{i=1}^{m} F(C_i)$ are called **deferred**.
- **Phase II**: Using exhaustive search to find an assignment of the values to the deferred variables s.t. all the surviving clauses are satisfied.

Lemma: Phase II is doable.

Pf: To show the random partial assignment fixed in Phase I can be extended to a full assignment of the problem by tossing a fair coin Y_i to each deferred variable x_i. Let $V' = \{ i : C_i \text{ is a surviving clause}\}$ and $D = \{ j : x_j \text{ is a deferred variable}\}$.

Consider the prob. space defined by the rvs in $\{Y_i\}_{i\in D}$.

For each $i \in V'$, let event $A_i \overset{\text{def}}{=} \{C_i \text{ is not satisfied by rvs in } \{Y_i\}_{i\in D}\}$.

Consider dependency digraph for the events $\{A_i\}_{i\in V'}$.

For each $i\in V'$, $\#\{j\in V' : j\neq i,\ A_j\cap A_i \text{ contains a deferred variable}\} \le k\cdot 2^{\alpha k}$

$P(A_i)4(k\cdot 2^{\alpha k}) \le \left(\frac{1}{2}\right)^{\frac{k}{2}} 4k2^{\alpha k} \le 1$ as $\alpha \le \frac{1}{2} - \frac{2+\log_2 k}{k}$ for even constant $k \ge 12$.

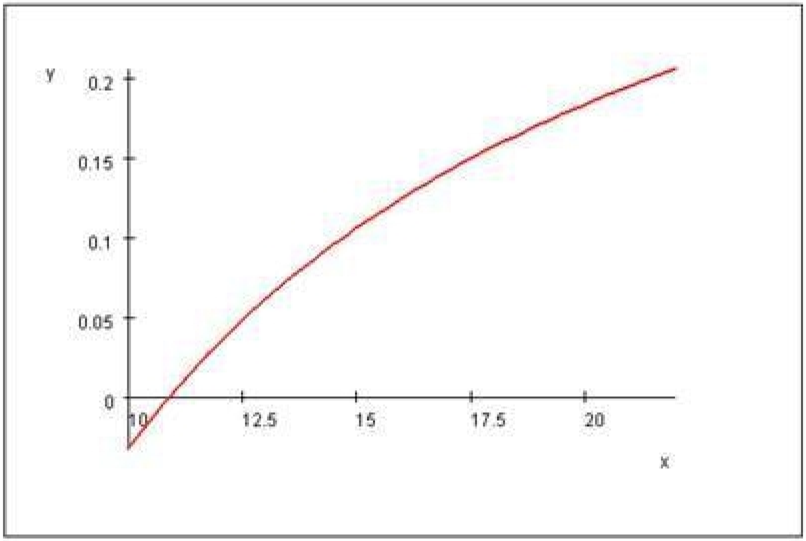
Note that $|C_i \cap D| \ge \frac{k}{2}$ for $\forall i\in V'$. By LLL, $\Pr\left(\bigcap_{i\in V'} \overline{A_i}\right) > 0$.

QED

Scientific WorkPlace - [C:\Documents and Settings\Deepthought\桌面\Untitled1.tex]

File Edit Insert View Go Tag Typeset Tools Compute Window Help

We need $\alpha \le \frac{1}{2} - \frac{2 + \frac{\ln x}{\ln 2}}{x}$



$$f(x) = \frac{1}{2} - \frac{2+\frac{\ln x}{\ln 2}}{x}$$

Body Text

What we need before an exhaustive Search?

The best scenario: The assignment of values in Phase I partitions the original formula into $\leq m$ subformula, so that each deferred variable appears in only one subformula.

And each subformula has the following form:

1. it is a CNF
2. it has $O(\log m)$ clauses.
3. each clause of a subformula has $\leq k$ literals.

Notation: $G = (V, E)$ and $G' = (V', E')$ where

$V = \{C_1, \cdots, C_m\}$, and $C_i \sim C_j$ in $E \iff C_i \cap C_j \neq \phi$. Note that $d_G(C) \leq k2^{ak}$ for $\forall C \in V$.

V' = all surviving clauses, and $C_i \sim C_j$ in $E' \iff C_i \cap C_j$ contains a deferred variable.

4-tree (I)

Ideas: Let R be a connected component of G.
Try to identify a vertex subset T of R such that the events of each of the clause in T are mutually independent.

4-tree: T is called a 4-tree if it satisfies

1. $i, j \in T \Rightarrow d_G(i,j) \geq 4$,
2. $E_T \overset{\text{def}}{=} \{ \overline{ij} : i, j \in T \text{ and } d_G(i,j) = 4 \} \Rightarrow G_T = (T, E_T)$ is a connected graph.

4-tree (II)

ClaimA Let event $A_c \overset{\text{def}}{=} \{\text{clause } C \text{ survives}\}$, T a 4-tree.
Then events in $\{A_c\}_{C \in T}$ are mutually indep..

Pf: (sketch) For $C, C' \in T$, we have

$$A_c \subseteq \bigcup_{\hat{C} \in N_G[C]} \{\hat{C} \text{ is dangeous}\} \quad \text{and}$$

$$A_{c'} \subseteq \bigcup_{\hat{C}' \in N_G[C']} \{\hat{C}' \text{ is dangeous}\}.$$

$d_G(C, C') \geq 4 \Longrightarrow d_G(\hat{C}, \hat{C}') \geq 2 \Longrightarrow \hat{C} \cap \hat{C}' = \phi$ done. QED

4-tree (III)

Claim B Let T be a 4-tree of a connected component R of G with the largest number of vertices. Then $|T| \geq |R|/d^3$, where $d = k2^{\alpha k}$.

pf:

$$|R| \overset{(\because T \text{ is a maximal 4-tree})}{=} \#\{C \in R : d_G(C,T) \leq 3\}$$

$$\leq \sum_{C' \in T} \#\{C \in G : d_G(C, C') \leq 3\}$$

$$\leq |T|\,(d + d(d-1) + d(d-1)^2) \leq |T|d^3.$$

QED

Count 4-trees

Thm The number of 4-trees of size s in G is bounded by $m\,(4d^4)^s$, where $d = k2^{\alpha k}$.

pf: By the definition of a 4-tree T, $G_T = (T, E_T)$ is connected and thus it must contain a spanning tree.

Cayley's Thm says, $\exists$ at most $\dfrac{|T|^{|T|-2}}{|T|!} < 4^{|T|}$ spanning trees on $|T|$ vertices (up to isomorphism). Also, for a specific spanning tree on $|T|$ vertices, the # of 4-trees containing this tree is bounded by $m\,[d\,(d-1)^3]^{|T|-1} \leq m\,d^{4|T|}$. Done!

QED

Key Observations

Thm $\Pr\{\text{all components of } G' \text{ have size} < c\log_2 m\} \geq \frac{1}{2}$, for a suitably large constant c and a sufficiently small constant α, and $k \geq 70$.

Pf: $\Pr\{G' \text{ has a component of size} \geq r\}$ (where $r \stackrel{\text{def}}{=} c\log_2 m$)

$\underset{\text{claim B}}{\leq} \Pr\{\exists \text{ a 4-tree } T \text{ of size } \frac{r}{d^3} \text{ in } G \text{ s.t. all nodes in } T \text{ survive}\}$ (where $d = k2^{\alpha k}$)

$\leq \sum_{\substack{T \text{ is a 4-tree} \\ \text{of size } r/d^3}} \Pr\{\text{all nodes in } T \text{ survive}\}$

$\underset{\text{claim A}}{=} \sum_{T} \prod_{C \in T} \Pr\{\text{clause } C \text{ survives}\}$

$= \sum_{T} \prod_{C \in T} \Pr\{\bigcup_{\hat{C} \in N_G[C]} \{\hat{C} \text{ is dangeous}\}\}$

pf (continued)

$$\leq \sum_{\substack{T \text{ is a 4-tree} \\ \text{of size } r/d^3}} \left[(d+1) \left(\tfrac{1}{2}\right)^{\frac{k}{2}} \right]^{|T|}$$

$$\leq m (4d^4)^{\frac{r}{d^3}} \left[(d+1) \left(\tfrac{1}{2}\right)^{\frac{k}{2}} \right]^{\frac{r}{d^3}}$$

count 4-tree

$$\leq m \left(\left[8k^5 2^{(5\alpha - \frac{1}{2})k} \right]^{\frac{c}{2d^3}} \right)^{2\log_2 m}$$

$$\leq m \left(\tfrac{1}{2}\right)^{2\log_2 m} \text{(for some } \alpha, c\text{)}$$

$$\leq \frac{1}{2}$$

QED

Explicit construction using the LLL

Thm The above algorithm finds a satisfying truth assignment for any instance of K-SAT containing m clauses in which each variable is contained in at most $2^{\alpha k}$ clauses for a sufficiently small constant $\alpha > 0$, in expected time polynomial in m.

Pf: By key observations and an exhaustive search!

QED