2-CNF codes

Chris Calabro

May 9, 2005

Abstract

We consider here the problem of upper bounding the number of solutions to a $\phi \in$ 2-CNF whose solution space $\text{sol}(\phi)$ is a distance-d code. We do so by giving a randomized algorithm and lower bounding the probability that it finds a particular solution.

1 Definitions

We will suppose wlog that $0 \in \text{sol}(\phi)$. We will also blur the distinction between an assignment being a sequence of bits, a subset of the variables (that are considered to be assigned 1), and a map from the variables to 2. e.g. if $a, b \in 2^n$ are assignments, then $a \subseteq b$ means that every variable assigned 1 in a is also assigned 1 in b. We will also consider the solution space $\text{sol}(\phi)$ to be a lattice ordered by the subset relation and with 0 at the bottom. The *height* of a solution $\alpha \in \text{sol}(\phi)$ is the length of a longest path from 0 to α in the lattice.

2 The algorithm

First we consider a subroutine find_next that takes as input a formula ϕ, a solution $\alpha \in 2^n$, and a direction $i_0 \in \text{var}(\phi)$. If there is a $\subseteq$-minimal solution β above α in the direction i_0, then find_next will find it.

find_next($\phi \in$ 2-CNF$, a \in 2^n, i_0 \in \text{var}(\phi)$)
  flip a_{i_0}
  loop
    if $\phi(a) = 1$, return a
    $C \in \text{false}(\phi, a)$ // choose a false clause arbitrarily
    $\{i, j\} \leftarrow \text{var}(C)$
    wlog $a_j = 1$
    flip a_i

Lemma 1. *Let $0, \alpha, \beta \in \text{sol}(\phi)$, $\alpha \subset \beta$, $\forall \gamma \in \text{sol}(\phi)$ $(\alpha \subset \gamma \subseteq \beta \to \gamma = \beta)$. (i.e. β is $\subseteq$-minimal of those solutions properly containing α) Let $i_0 \in \beta - \alpha$. Then* find_next$(\phi, \alpha, i_0) = \beta$.

Proof. We prove the following loop invariant: at the beginning of loop iteration k, $|a| = |\alpha| + k$ and $\alpha \subset a \subseteq \beta$.

This is true for $k = 1$ since $\alpha \subset \beta$ and $i_0 \in \beta - \alpha$. Now suppose the invariant holds for iteration $k - 1$.

If $\phi(a) = 1$, then find_next returns a. By the induction hypothesis, $\alpha \subset a \subseteq \beta$ and so $a = \beta$, which was to be shown.

If $\phi(a) = 0$, then let C be any false clause at a. Let $\{i, j\} = \mathrm{var}(C)$. Since $0 \in \mathrm{sol}(\phi)$, $C(0) = 1$. But $C(a) = 0$. So $a_i = 1$ or $a_j = 1$. Wlog suppose $a_j = 1$.

We claim that $a_i = 0$. For suppose not. Then $\{i, j\} \subseteq a \subseteq \beta$, by the induction hypothesis, and that implies $C(\beta) = C(a) = 0$, contradicting that $\beta \in \mathrm{sol}(\phi)$. So $a_i = 0$. So flipping a_i increases the size of a by 1.

By the induction hypothesis, $j \in \beta$. If $i \notin \beta$, then we again have the contradiction $C(\beta) = 0$. So $i \in \beta$. So the invariant holds.

Since $|a| \leq n$, eventually the loop will terminate and output β. □

We now consider using find_next as a subroutine to a randomized algorithm find_sol that takes as input a $\phi \in$ 2-CNF and a height $h \in \{0, \ldots, n\}$. If there is a solution $\alpha \in \mathrm{sol}(\phi)$ at height h, then find_sol will find α with probability that we will lower bound later.

```
find_sol(φ ∈ 2-CNF, h ∈ {0, ..., n})
  a ← 0
  do h times
    i_0 ←$ var(φ) − a
    a ← find_next(φ, a, i_0)
  return a
```

Lemma 2. *Suppose* $\mathrm{sol}(\phi)$ *is a distance-*d *code containing* 0*. Let* $\alpha \in \mathrm{sol}(\phi)$ *be at height* h*. Then*

$$Pr(\text{find\_sol}(\phi, h) = \alpha) \geq \Big(\frac{d}{n}\Big)\Big(\frac{d}{n-d}\Big)\Big(\frac{d}{n-2d}\Big)\cdots\Big(\frac{d}{n-(h-1)d}\Big).$$

Proof. Let $0 = \beta_0 \subset \beta_1 \subset \cdots \subset \beta_h = \alpha$ be a longest path in the lattice $\mathrm{sol}(\phi)$ from 0 to α, so that each β_i is $\subseteq$-minimal of those solutions properly containing β_{i-1}.

Since $\mathrm{sol}(\phi)$ is a distance-d code, $|\beta_i| \geq |\beta_{i-1}| + d$. So $h \leq \frac{n}{d}$. By lemma 1, $\forall i_0 \in \beta_i - \beta_{i-1}$ $\text{find\_next}(\phi, \beta_{i-1}, i_0) = \beta_i$. Also $|\beta_i - \beta_{i-1}| \geq d$ and $|\mathrm{var}(\phi) - \beta_{i-1}| \leq n - (i-1)d$. So

$$\begin{aligned}
&Pr(\text{find_sol}(\phi, h) = \alpha) \\
&\geq \prod_{i=1}^{h} Pr_{i_0}(\text{find_next}(\phi, \beta_{i-1}, i_0) = \beta_i) \\
&\geq \prod_{i=1}^{h} \Big(\frac{d}{n-(i-1)d}\Big).
\end{aligned}$$

∎

Lemma 3. *Let X be a random variable and S be a set. If $\forall s \in S$ $Pr(X = s) \geq p$, then $|S| \leq \frac{1}{p}$.*

Proof.

$$1 \geq Pr(X \in S) = \sum_{s \in S} Pr(X = s) \geq |S|p.$$

∎

Theorem 4. *Suppose* $\mathrm{sol}(\phi)$ *is a distance-d code containing* 0. *Then* $|\mathrm{sol}(\phi)| \leq e(\frac{n}{d})!$.

Proof. Let S_h be the set of solutions of ϕ at height h. Then

$$|S| \leq \sum_{h=0}^{\frac{n}{d}} |S_h| \leq_{\text{lemmas 2,3}} \sum_{h=0}^{\frac{n}{d}} \left(\frac{n}{d}\right)\left(\frac{n}{d} - 1\right) \cdots \left(\frac{n}{d} - (h-1)\right)$$
$$\leq \left(\frac{n}{d}\right)!\left(1 + \frac{1}{2!} + \frac{1}{3!} + \cdots\right) = e\left(\frac{n}{d}\right)!.$$

∎