GSAT with Adaptive Score Function

Vladimir Popov

Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
Vladimir.Popov@usu.ru

Copyright © 2013 Vladimir Popov. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

GSAT is a well-known satisfiability search algorithm. In this paper we consider a modification of GSAT. In particular, we consider an adaptive score function.

PACS: 02.70.Rr

Keywords: adaptive score function, GSAT, satisfiability

In this paper we consider a modification of GSAT. In particular, we consider an adaptive score function.

PACS: 02.70.Rr

Keywords: adaptive score function, GSAT, satisfiability
the score. Let $T_i$ be the sequence obtained from $T$ by inverting the value of $i$th variable $V_i$. The function $\text{hill-climb}(f, T)$ returns the set $\text{Poss-flips}$ of the variables $V_i$ which minimize $\text{score}(T_r, f)$. The function $\text{pick}(\text{Poss-flips})$ chooses randomly one of elements of $\text{Poss-flips}$. The function $\text{flip}(V, T)$ returns $T$ with $V$’s value inverted. The function $\text{UpdateScores}(f, V)$ updates score.

**procedure** GSAT($f$)

for $j := 1$ to $\text{Max-tries}$ do

$T := \text{initial}(f)$

for $k := 1$ to $\text{Max-flips}$ do

if $T \models f$ then return $T$

else $\text{Poss-flips} := \text{hill-climb}(f, T)$

$V := \text{pick}(\text{Poss-flips})$

$T := \text{flip}(V, T)$

$\text{UpdateScores}(f, V)$

end

end

return “no satisfying assignment found”.

Figure 1: A general schema for GSAT.

The function $\text{score}(T, f)$ is the number of clauses of $f$ which are falsified by $T$. In general, there are a number of different types of clauses. For instance, there is some difference between clauses $x$ and $y \lor z$. But, without additional information, we cannot give reasonable score of such difference. In particular, if $x = y = z = u = v = 0$,

\[
\begin{aligned}
&x \land (y \lor z) \land (\neg x \lor u) \land (\neg x \lor v) \land (y \lor u), \\
&x \land (y \lor z) \land (\neg x \lor u) \land (\neg y \lor u) \land (\neg y \lor v) \land (\neg z \lor u) \land (\neg z \lor v),
\end{aligned}
\tag{1}
\]

then we need modify two values of variables ($u$ and $v$) for $x$ and only one value of variable ($u$) for $y \lor z$ in formula (1), but we need modify two values of variables ($u$ and $v$) for $y \lor z$ and only one value of variable ($u$) for $x$ in formula (2). So, we can assume that the score of $x$ higher than the score of $y \lor z$, for formula (1), and the score of $x$ lower than the score of $y \lor z$, for formula (2).

Let $\#\text{occ}(k, f, z[i])$ be the number of positive occurrences of $z[i]$ in clauses of type $x_1 \lor \ldots \lor x_k$, $\#\text{occ}(k, f, \neg z[i])$ be the number of occurrences of $\neg z[i]$ in clauses of type $x_1 \lor \ldots \lor x_k$. Let $S(T, f)$ be the set of clauses of $f$ which are falsified by $T$. We assume that

\[
\text{score}(T, f) = \sum_{x_1 \lor \ldots \lor x_p \in S(T, f)} \sum_{1 \leq i \leq q, 1 \leq j \leq p} (\alpha(\#\text{occ}(i, f, x_j)) + \beta(\#\text{occ}(i, f, \neg x_j))).
\]
We use a genetic algorithm for prediction of values of $\alpha$ and $\beta$.

Now, we consider a special class of formulas 3-2-CNF. We assume that any clause of formula from this class belongs the set

$$\{z[i] \lor z[j] \lor z[l], \lnot z[i] \lor \lnot z[j] | i, j, l \in N\}. \quad (3)$$

**Theorem.** For any 3-CNF $f$, there is a formula $g$ such that any clause of $g$ belongs the set (3) and $f$ is satisfiable if and only if $g$ is satisfiable.

**Proof.** It is easy to see that $x_1 \lor x_2 \lor \lnot x_3$ is satisfiable if and only if $(x_1 \lor x_2 \lor x_4) \land (\lnot x_3 \lor \lnot x_4)$ is satisfiable. \hfill \square

In view of theorem, we can be sure that the class 3-2-CNF is sufficiently general. In our experiments, we consider GSAT with standard score function and GSAT with our score function (GSAT-ASF) for CNFs, 3-CNFs, and 3-2-CNFs. Selected experimental results are given in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>3-CNF</th>
<th>3-2-CNF</th>
<th>CNF</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSAT</td>
<td>86 %</td>
<td>81 %</td>
<td>79 %</td>
</tr>
<tr>
<td>GSAT-ASF, $G = 10^4$</td>
<td>91 %</td>
<td>89 %</td>
<td>81 %</td>
</tr>
<tr>
<td>GSAT-ASF, $G = 10^5$</td>
<td>92 %</td>
<td>94 %</td>
<td>85 %</td>
</tr>
<tr>
<td>GSAT-ASF, $G = 10^6$</td>
<td>93 %</td>
<td>95 %</td>
<td>88 %</td>
</tr>
</tbody>
</table>

Table 1: A number of solved formulas for GSAT and GSAT-ASF where $G$ is a number of generations of genetic algorithm.

**ACKNOWLEDGEMENTS.** The work was partially supported by Analytical Departmental Program “Developing the scientific potential of high school” 8.1616.2011.

**References**


Received: February 12, 2013