Switching Among Heuristics in Local Search for SAT

Wanxia Wei[1], Harry Zhang[1], and Chu Min Li[2]

[1] Faculty of Computer Science, University of New Brunswick, Fredericton, NB, Canada, E3B 5A3
{wanxia.wei,hzhang}@unb.ca
[2] LaRIA, Université de Picardie Jules Verne 33 Rue St. Leu, 80039 Amiens Cedex 01, France
chu-min.li@u-picardie.fr

1 Review of Algorithms $adaptG^2WSAT_P$ and VW

If promising decreasing variables exist, the local search algorithm $adaptG^2WSAT_P$ [3, 4] flips the promising decreasing variable with the largest computed promising score. Otherwise, it selects a variable to flip from a randomly chosen unsatisfied clause c using $Novelty$++$_P$, which is described as follows.

$Novelty$++$_P$(p, dp): With probability dp (diversification probability), flip a variable in c whose flip can falsify the least recently satisfied clause. With probability $1 - dp$, do as $Novelty$, but flip $second$ if $best$ is more recently flipped than $second$ and if $pscore(second) \geq pscore(best)$.

The local search algorithm VW [6] introduces variable weighting. This algorithm initializes the weight of a variable x, $variable\_weight[x]$, to 0 and updates and smoothes $variable\_weight[x]$ each time x is flipped, using the following equation:

$$variable\_weight[x] = (1 - s)(variable\_weight[x] + 1) + s \times t \quad (1)$$

where s is a parameter and $0 \leq s \leq 1$, and t denotes the time when x is flipped. VW uses a variable selection rule. We call this rule the *low variable weight favoring rule*. If a randomly selected unsatisfied clause c contains freebie variables,[3] VW randomly flips one of them. Otherwise, with probability p, it flips a variable chosen randomly from c, and with probability $1 - p$, it flips a variable in c according to this

Based on $adaptG^2WSAT_P$ and VW, we develop a new local search algorithm called FH (Four Heuristics). This algorithm can switch among four heuristics, two of which use variable weighting. Although several local search algorithms can switch between heuristics [1, 5, 2–4], none of the alternating or switching heuristics uses any weighting.

2 Algorithm FH

Assume that variable weights are updated using Equation 1. Assume that γ and ζ are two integers and that $1 < \gamma < \zeta$. If the maximum variable weight is at least γ times as high as the average variable weight, the distribution of variable weights is considered *uneven*. Otherwise, the distribution of variable weights is considered *even*. If the weight of the flipping variable is at least ζ times as high as the average variable weight, the flip is called *a flip with an extremely high variable weight*.

We propose the following two heuristics for choosing the best promising decreasing variable.

T: Choose the least recently flipped promising decreasing variable to flip.
W: Choose the promising decreasing variable with the lowest weight to flip.

Based on the algorithm VW, we design a heuristic called VW', which is described as follows.

$VW'(dp)$: With probability dp (diversification probability), flip the variable with the lowest variable weight in a randomly selected unsatisfied clause c. With probability $1 - dp$, randomly choose a freebie variable to flip from c if there are freebie variables in c, but select a variable to flip from c according to the low variable weight favoring rule otherwise.

[3] A freebie variable is a variable with a break of 0.

Algorithm: FH(SAT-formula $\mathcal{F}$)

$A\leftarrow$ randomly generated truth assignment;
for each variable x **do** initialize $flip\_time[x]$ and $variable\_weight[x]$ to 0;
initialize p, dp, max_weight, and ave_weight to 0; store promising decreasing variables in stack *DecVar*;
for *flip*=1 **to** *Maxsteps* **do**
  if A satisfies $\mathcal{F}$ **then return** A;
  $y\leftarrow -1$; /* variables are represented as non-negative numbers */
  if $max\_weight \geq \gamma \times ave\_weight$
  then
    if $|DecVar| > 0$
    then
      $y\leftarrow$a variable chosen by W; **if** $variable\_weight[y] \geq \zeta \times ave\_weight$ **then** $y\leftarrow -1$;
    if (y=-1) **then** $c\leftarrow$a randomly selected unsatisfied clause; $y\leftarrow VW'$(dp, c);
  else
    if $|DecVar| > 0$ **then** $y\leftarrow$a variable chosen by T;
    else $c\leftarrow$a randomly selected unsatisfied clause; $y\leftarrow Novelty$++$_P$(p, dp, c);
  $A\leftarrow A$ with y flipped; adapt p and dp;
  update $flip\_time[y]$, $variable\_weight[y]$, max_weight, ave_weight, and *DecVar*;
return Solution not found;

Fig. 1. Algorithm FH

We take the evenness or non-evenness of the distribution of variable weights as one criterion for switching among heuristics. We use the existence or non-existence of promising decreasing variables as the other criterion for switching among heuristics. This criterion is originally proposed in [2].

FH (Four Heuristics) is sketched in Fig. 1.

3 Contest Implementation

For the SAT 2007 competition, parameter γ in FH is set to 10.

References

1. E. A. Hirsch and A. Kojevnikov. UnitWalk: A New SAT Solver That Uses Local Search Guided by Unit Clause Elimination. *Ann. Math. Artif. Intell.*, 43(1):91–111, 2005.
2. C. M. Li and W. Q. Huang. Diversification and Determinism in Local Search for Satisfiability. In *Proceedings of SAT-2005*, pages 158–172. Springer, LNCS 3569, 2005.
3. C. M. Li, W. Wei, and H. Zhang. Combining Adaptive Noise and Look-Ahead in Local Search for SAT. In *Proceedings of LSCS-2006*, pages 2–16, 2006.
4. C. M. Li, W. Wei, and H. Zhang. Combining Adaptive Noise and Look-Ahead in Local Search for SAT. In F. Benhamou, N. Jussien, and B. O'Sullivan, editors, *Trends in Constraint Programming*, chapter 2. Hermes Science, 2007 (to appear).
5. X. Y. Li, M. F. M. Stallmann, and F. Brglez. A Local Search SAT Solver Using an Effective Switching Strategy and an Efficient Unit Propagation. In *Proceedings of SAT-2003*, pages 53–68. Springer, LNCS 2919, 2003.
6. S. Prestwich. Random Walk with Continuously Smoothed Variable Weights. In *Proceedings of SAT-2005*, pages 203–215. Springer, LNCS 3569, 2005.