Adaptive Restart Strategies for Conflict Driven SAT Solvers

Armin Biere

Institute for Formal Models and Verification
Johannes Kepler University
Linz, Austria

Eleventh International Conference on
Theory and Applications of Satisfiability Testing

SAT 2008

Guangzhou, P. R. China

Monday, May 12, 2008

Results of PicoSAT in SAT'07 Competition: Industrial Instances

Solver	Score	#SAT	#Uns
Rsat 2007-02-08	55452	63	76
picosat 535	51639	72	67
minisat SAT-2007	50398	53	79
TiniSatELite 2007-02-08	49560	55	75
CMUSAT 2007-02-08	41197	46	77
MXC 2007-02-08	38004	47	69
MiraXT v3	37700	52	74
SATzilla CRAFTED	31439	47	67
SAT7 2007-02-08	27607	51	63

Results of PicoSAT in SAT'07 Competition: Crafted Instances

Solver	Score	#SAT	#Uns
SATzilla CRAFTED	74469	27	67
minisat SAT-2007	63371	26	72
MXC 2007-02-08	39848	20	57
MiraXT v3	34236	24	54
CMUSAT 2007-02-08	26461	21	45
Rsat 2007-02-08	19532	15	40
picosat 535	19081	22	38

New Normalized VSIDS (NVSIDS) Reformulation of VSIDS

(consider only one variable)

feedback / punishment / I in PID: $0 < f < 1$

s old score s' new score

$$s' = \begin{cases} s \cdot f + (1-f) & \text{if variable is involved in current conflict} \\ s \cdot f & \text{if variable is NOT involved} \end{cases}$$

$$\boxed{0 \leq \underbrace{s \cdot f}_{\text{decay in any case}} \leq s' \leq s \overbrace{\cdot f}^{\text{decay in any case}} \underbrace{+ (1-f)}_{\text{increment if involved}} \leq f + (1-f) = 1}$$

MiniSAT, RSAT: $f = 0.95 \approx 1/1.05$ $(1-f) = 0.05$

PicoSAT: $f = 1/1.1 \approx 0.91$ $(1-f) = 0.09$

Exponential VSIDS (EVSIDS) as in MiniSAT

(consider only one variable)

$$\delta_k \quad = \quad \begin{cases} 1 & \text{if involved in } k\text{-th conflict} \\ 0 & \text{otherwise} \end{cases}$$

$$i_k \quad = \quad (1 - f) \cdot \delta_k$$

$$s_n = (\ldots(i_1 \cdot f + i_2) \cdot f + i_3) \cdot f \cdots) \cdot f + i_n = \sum_{k=1}^{n} i_k \cdot f^{n-k} = (1-f) \cdot \sum_{k=1}^{n} \delta_k \cdot f^{n-k} \qquad \text{(NVSIDS)}$$

$$S_n = \frac{f^{-n}}{(1-f)} \cdot s_n = \frac{f^{-n}}{(1-f)} \cdot (1-f) \cdot \sum_{k=1}^{n} \delta_k \cdot f^{n-k} = \sum_{k=1}^{n} \delta_k \cdot f^{-k} \qquad \text{(EVSIDS)}$$

Flips

- **phase saving** of assigned variables [RSAT]
  - initially pick phase according to number of occurrences
  - afterwards always pick last saved phase for decision variables
  - **rapid restarts** [TiniSAT] empirically [RSAT,PicoSAT] work nicely with phase saving
- **flipped assignment** or just **flip**
  - is a forced assignment (no decision)
  - with different previous saved phase, e.g.
    - last time x was assigned, it was assigned to 0
    - because of unit propagation x is now forced to be assigned 1

Average Number Recently Flipped Assignments (ANRFA)

(consider only one variable)

feedback / punishment / I in PID: $0 < g < 1$

a old agility a' new agility

$$a' = \begin{cases} a \cdot g + (1-g) & \text{if new forced assignment flips old value} \\ a \cdot g & \text{if saved phase is the same as in new assignment} \end{cases}$$

$$0 \leq a \underbrace{\cdot g}_{\text{decay in any case}} \leq a' \leq a \overbrace{\cdot g}^{\text{decay in any case}} \underbrace{+ (1-g)}_{\text{increment if flipped}} \leq g + (1-g) = 1$$

PicoSAT: $g = 0.9999 = 1 - 1/10000, \quad (1-g) = 1/10000$

Prohibiting Restarts

- high agility
  - measured in percentage of recent assignments that flipped variables
  - PicoSAT: at least 20% for inner restarts, at least 25% for outer restarts
  - then SAT solver is moving fast
  - with respect to Hamming distance
  - do not restart (actually do not backtrack but follow restart schedule)

- low agility
  - PicoSAT: less than 20% for inner restarts, less than 25% for outer restarts
  - SAT solver is somehow stuck
  - restart as usual

Experiments with SAT Competition'07 Benchmarks

time limit 900 seconds, memory limit 1.5 GB

			industrial			crafted	
		adaptive	*sat*	*unsat*	*solved*	*sat*	*unsat*	*solved*
MiniSAT	2.0	*no*	37	57	94	22	46	68
orig. RSAT	2.0	*no*	41	51	92	10	20	30
AAS-RSAT		*no*	45	48	93	11	21	32
AAS-RSAT	25%	**yes**	44	49	93	11	24	35
AAS-RSAT	30%	**yes**	48	48	96	12	23	35
PicoSAT	741	*no*	43	54	97	14	24	38
PicoSAT	741	**yes**	44	57	101	16	**36**	52
PicoPrepSAT	143	*no*	52	58	110	21	38	59
PicoPrepSAT	143	**yes**	52	64	116	21	**49**	70

Conclusion

- new metric ANRFA to measure agility / velocity of search
  - reformulated VSIDS as control problem (NVSIDS)
  - applied same “controller” to flipped assignments
  - came up with good magic constants: $g = 0.9999$ 20% 25%
- prohibit restarts if SAT solver is not moving fast enough
  - are there better metrics than ANRFA?
  - does ANRFA capture velocity with respect to Hamming distance good enough?
- sofar failed to extend these ideas to other magic constants
  - average recently used learned clauses can be calculated in the same way
  - goal is to control reduce schedule resp. garbage collection schedule