Reusing the Assignment Trail in CDCL Solvers

Peter van der Tak, Marijn Heule, and Antonio Ramos

Overview

- CDCL Solver & VSIDS
- Motivation
- ReusedTrail
- Results & Observations
- Conclusion

CDCL Solver

trail										
level										
score										
RSL										

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, x7, x12, x4, x10, x6, x8, x11

decided, implied, *unassigned*

CDCL Solver

trail	x5										
level	1										
score	93.5										
RSL											

VSIDS order:
x5, *x2*, *x13*, *x9*, *x3*, *x14*, *x1*, *x7*, *x12*, *x4*, *x10*, *x6*, *x8*, *x11*

decided, implied, *unassigned*

CDCL Solver

trail	x5	x2	x11								
level	1	2									
score	93.5	88.2	15.9								
RSL											

VSIDS order:
x5, x2, *x13*, *x9*, *x3*, *x14*, *x1*, *x7*, *x12*, *x4*, *x10*, *x6*, *x8*, x11

decided, implied, *unassigned*

CDCL Solver

trail	x5	x2	x11	x13	x9	x4	x3	x14	x8	x1	x12
level	1	2		3	4		5			6	
score	93.5	88.2	15.9	81.2	75.4	44.0	63.9	62.8	27.7	38.1	44.1
RSL									BJL		

conflict

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, *x7*, x12, x4, *x10*, *x6*, x8, x11

decided, implied, *unassigned*

VSIDS

When a conflict occurs:

- The solver analyses the conflicting clause and reason clauses
- Learns a conflict clause
- Increments scores of all variables in the conflict clause
- Multiplies (decays) variables by δ (VSIDS decay factor)
- Jumps to the highest level at which the conflict clause is unit backjump level (BJL)

CDCL Solver

trail	x5	x2	x11	x13	x9	x4	x3	x14	x8	x1	x12
level	1	2		3	4		5			6	
score	93.5	88.2	15.9	81.2	75.4	44.0	63.9	62.8	27.7	38.1	44.1
RSL											

conflict

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, *x7*, x12, x4, *x10*, *x6*, x8, x11

decided, implied, *unassigned*

CDCL Solver

trail										
level										
score										
RSL										

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, x7, x12, x4, x10, x8, x6, x11

decided, implied, *unassigned*

Motivation

- Frequent restarts lead to fewer conflicts
- But restarts are costly, solving time may increase

- VSIDS remains similar between frequent restarts
- Phase saving ensures that assignments will have equal values

Example

trail	x5	x2	x11	x9	x13	x4	x14	x8	x1	x6	x3
level	1	2		3	4		5		6	7	
score	93.5	88.2	15.9	75.4	81.2	44.0	62.8	27.7	53.6	38.1	63.9
RSL											BJL

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, *x7*, *x12*, x4, *x10*, x6, x8, x11

decided, implied, *unassigned*

Example (MatchingTrail)

trail	x5	x2	x11	x9	x13	x4	x14	x8	x1	x6	x3
level	1	2		3	4		5		6	7	
score	93.5	88.2	15.9	75.4	81.2	44.0	62.8	27.7	53.6	38.1	63.9
RSL			MTL								BJL

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, *x7*, *x12*, x4, *x10*, x6, x8, x11

decided, implied, *unassigned*

Example (PermutedTrail)

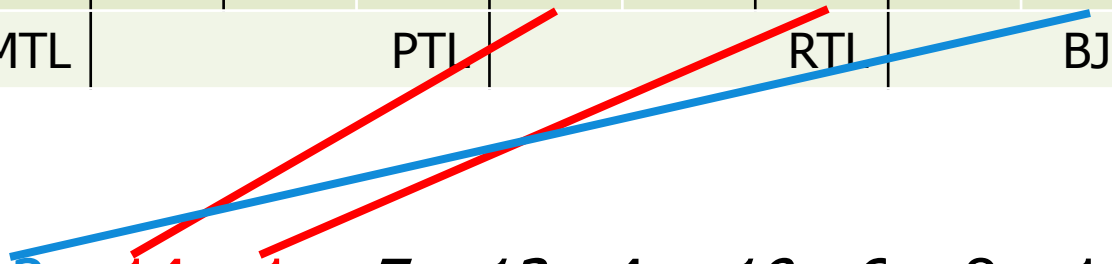
trail	x5	x2	x11	x9	x13	x4	x14	x8	x1	x6	x3
level	1	2		3	4		5		6	7	
score	93.5	88.2	15.9	75.4	81.2	44.0	62.8	27.7	53.6	38.1	63.9
RSL			MTL			PTL					BJL

VSIDS order:
x5, x2, x13, x9, x3, x14, x1, *x7*, *x12*, x4, *x10*, x6, x8, x11

decided, implied, *unassigned*

Example (ReusedTrail)

trail	x5	x2	x11	x9	x13	x4	x14	x8	x1	x6	x3
level	1	2		3	4		5		6	7	
score	93.5	88.2	15.9	75.4	81.2	44.0	62.8	27.7	53.6	38.1	63.9
RSL			MTL			PTL			RTL		BJL



VSIDS order:
x5, x2, x13, x9, x3, x14, x1, *x7*, *x12*, x4, *x10*, x6, x8, x11

decided, implied, *unassigned*

Algorithm

forever do
  if OrderHeap.empty() **then return** BackjumpLevel
  x_{next} := OrderHeap.remove()
  if AssignmentType[x_{next}] = Unassigned **then break**
OrderHeap.insert(x_{next})

for i := 1 **to** BackjumpLevel **do**
  if Activity[DecisionVar[i]] < Activity[x_{next}] **then return** i – 1
return BackjumpLevel

Approach

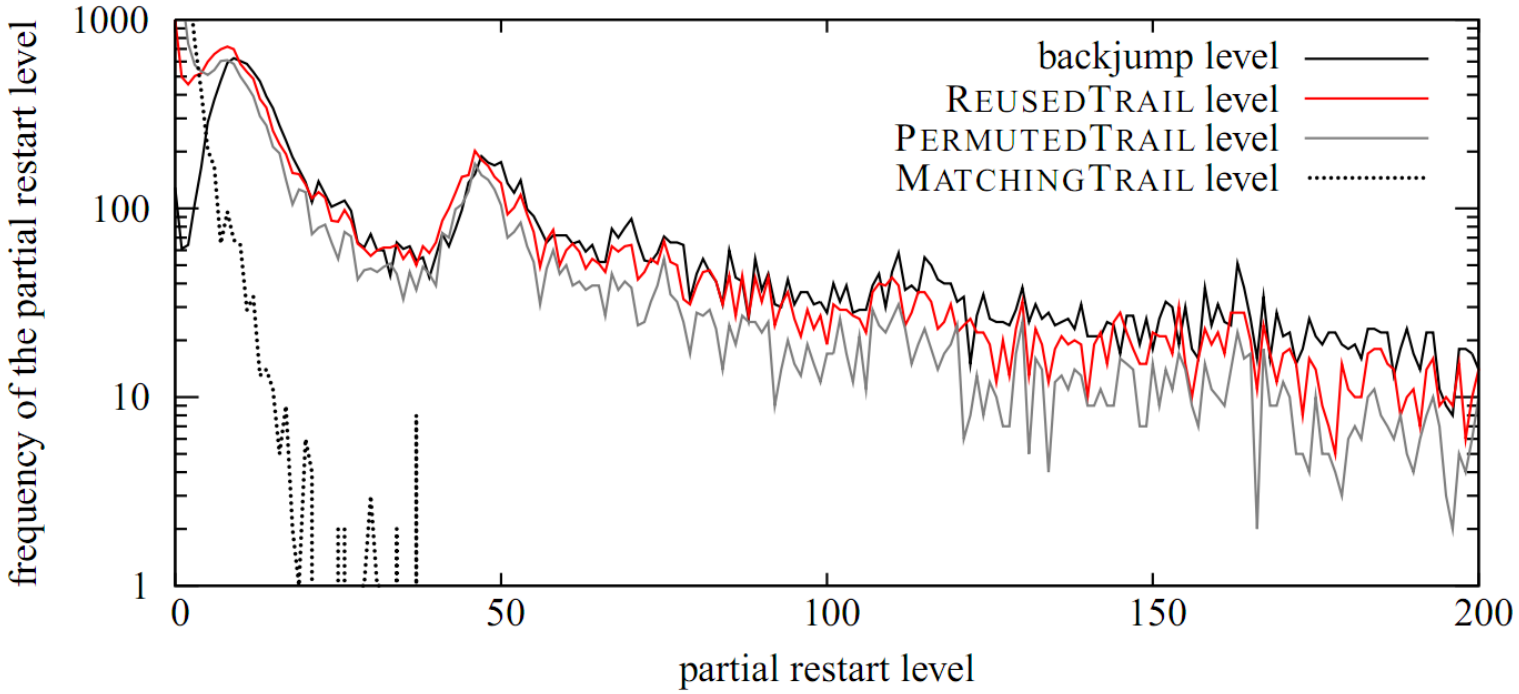
- Exploiting similarity before and after a restart
- Backtrack to RTL instead of RSL
- This retains the effect of a restart

Results

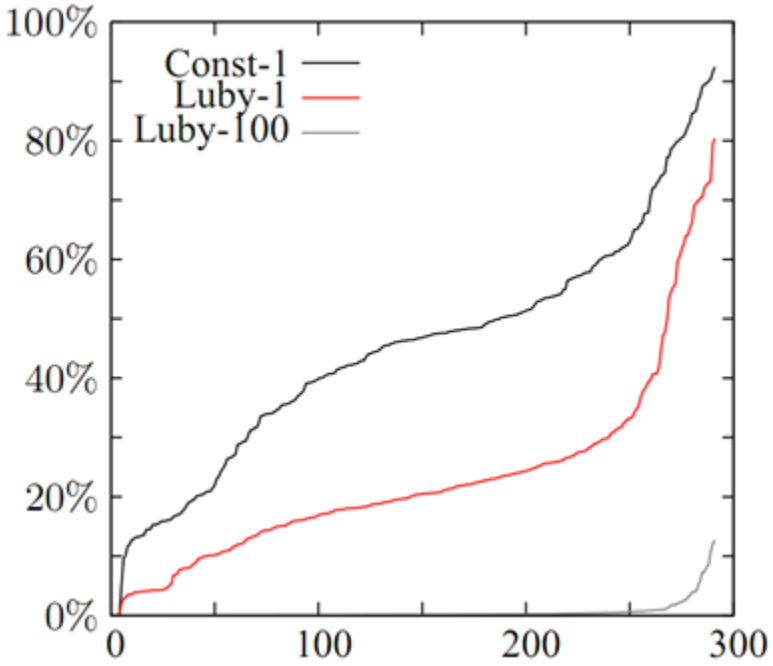
- Implemented in MiniSAT 2.2
- Consider Const-1, Luby-1, and Luby-100

- Determine the amount of work saved by ReusedTrail
- Count the number of extra instances solved

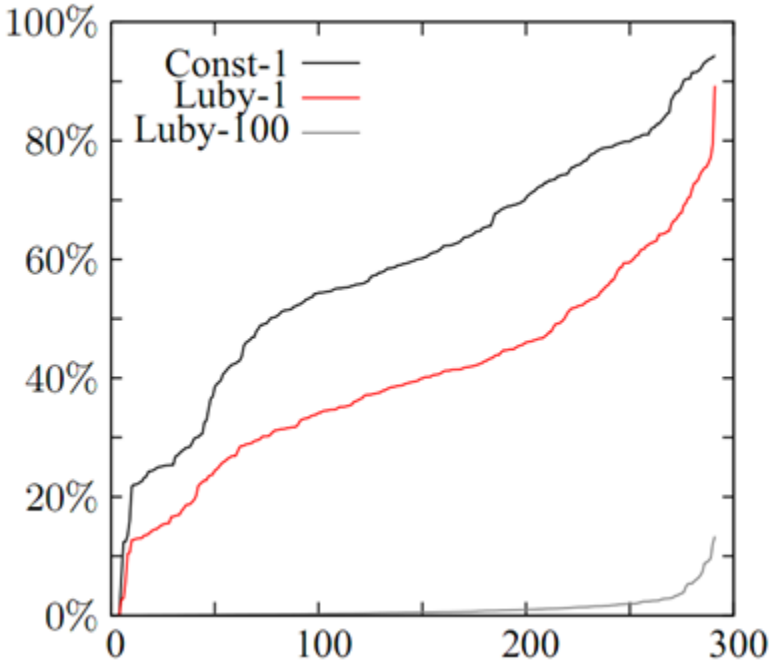
Results (Frequencies)



Results (Savings)



Propagations



Decisions

Results (Solved instances)

Number of instances solved within 1200 seconds
(out of all 292 application instances of the SAT 2009 competition)

	No RT	RT	RT
	δ = 0.95	δ = 0.95	δ = 0.75
Const-1	147	163	169
Luby-1	168	174	185
Luby-100	170	172	176

δ = VSIDS decay factor

Observations

- More radical restarts become more efficient
- The optimal VSIDS decay value becomes smaller
- Reason clauses may become different

Conclusion

- ReusedTrail significantly reduces restart costs
- This allows more radical restart strategies to perform better

- Easy to implement in most CDCL solvers

Thank you!

Reusing the Assignment Trail in CDCL Solvers

Peter van der Tak, Marijn Heule, and Antonio Ramos

Code available from: http://www.st.ewi.tudelft.nl/sat/download.php