Local Search For SMT on Linear Integer Arithmetic

CAV 2022

Shaowei Cai, Bohan Li, Xindi Zhang
{caisw, libh, zhangxd}@ios.ac.cn

Arithmetic atomic formulae: $\sum_i a_i x_i + c \bowtie 0, \bowtie \in \{=, \leq\}, c\ and\ a_i\ are\ rational\ numbers\ and\ x_i\ are\ interger\ variables$

SMT(LIA):

Test the satisfiability of the Boolean combination of arithmetic atomic formulae and propositional variables

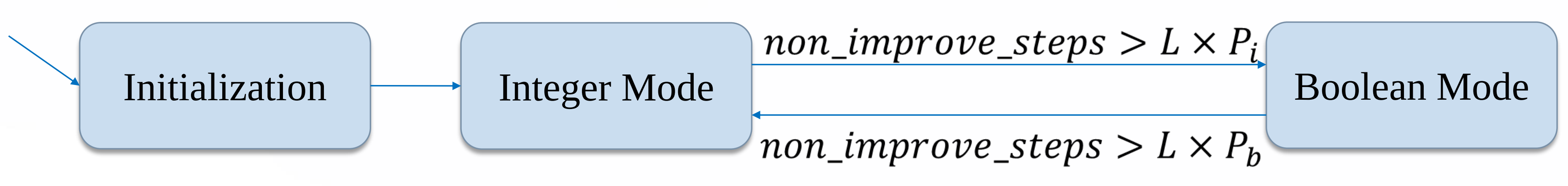
e.g. $\phi = (x_1 - x_2 \leq 13 \vee x_2 \neq x_3) \wedge (x_2 = x_3 \rightarrow x_4 > x_5) \wedge A \wedge \neg B$

SMT on Integer Difference Logic(IDL) :

a fragment of SMT(LIA) where arithmetic atomic formulae in the form of $x_i - x_j \leq$ k

Application: Automated termination analysis, Sequential equivalence checking, State reachability checking, Job shop scheduling, e.t.c

Two mode Local Search Framework



P_b , P_i : the proportion of Boolean and integer literals to all literals in falsified clauses

L : parameter

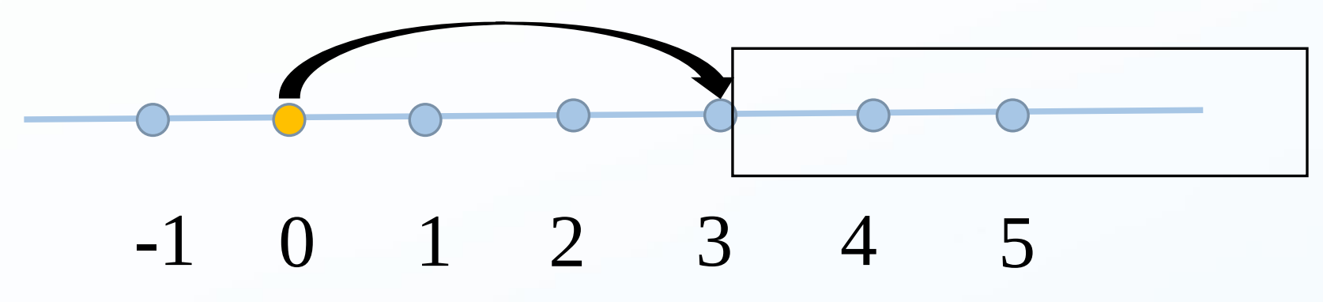
$non\_improve\_steps$: non-improving steps of the current mode

Literal-level operator: critical move

The critical move operator, $cm(x, \ell)$, assigns an integer variable x to the threshold value making literal ℓ true, where ℓ is a falsified literal containing x.

Example:

- given two falsified literals l_1: $(2b - a \leq -3)$ and l_2: $(5c - d + 3a = 5)$ where the assignment is $\{a = b = c = d = 0\}$
- $cm(a, l_1), cm(b, l_1), cm(c, l_2), cm(d, l_2)$ refers to assigning a, b, c, d to $3, -2, 1, -5$, respectively.



$cm(a, l_1)$ assigns a to 3

Two-level picking heuristic

Candidate set of decreasing operation

$D = \{cm(x, \ell) | \ell\ \ is\ a\ false\ literal\ and\ x\ appears\ in\ \ell\}$

A special subset $S \subseteq D$

$S = \{cm(x, \ell) | \ell\ appears\ in\ at\ least\ one\ falsified\ clause\}$

Level 1
- Search for a decreasing cm operation from S

Level 2
- Search for decreasing cm operation from D\S

Fine grained scoring function: distance score

Distance to Truth of literal: *dtt*

Given an assignment α and an literal $\sum_i a_i x_i \leq k$.

$$dtt(\ell, \alpha) = max\{\sum_i a_i \alpha(x_i) - k, 0\}$$

Extend to Clause

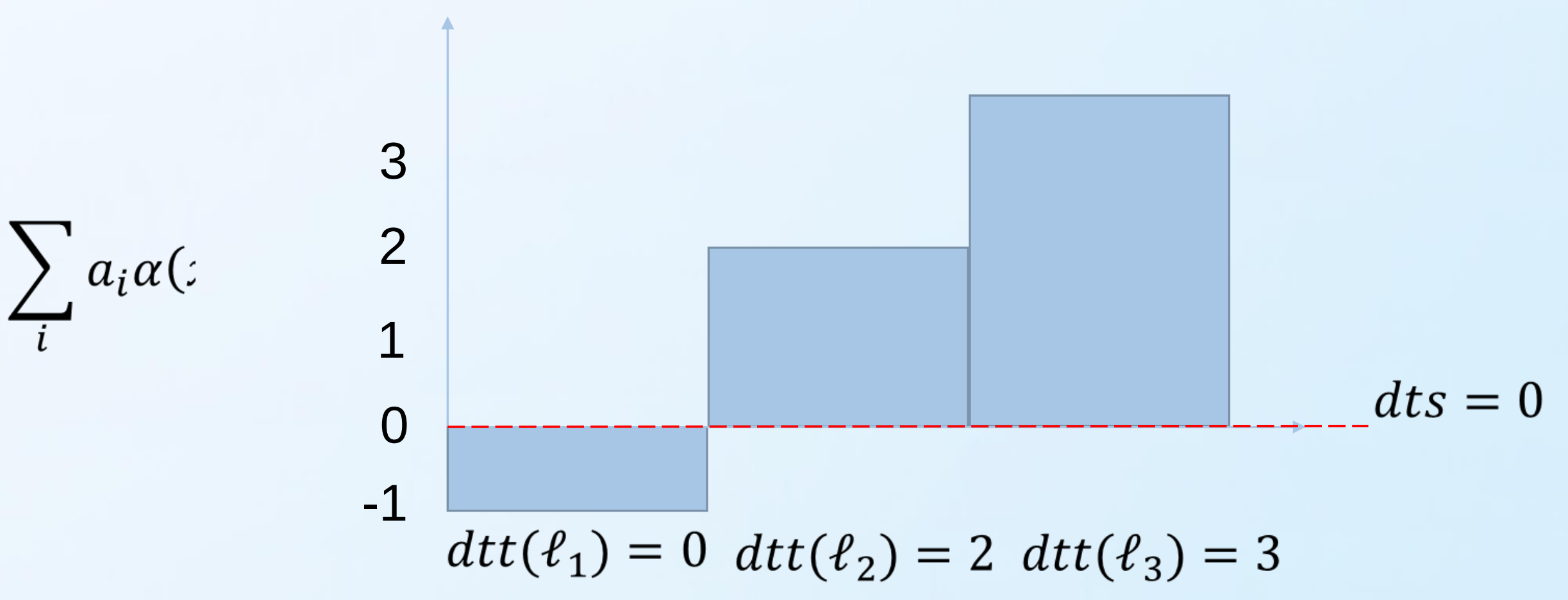
Distance to satisfaction of clauses: *dts*

Given an assignment α and a clause C

$$dts(C, \alpha) = \min_{l \in C}\{dtt(\ell, \alpha)\}$$

Property: $dts(\ell, \alpha) = 0$ when C is satisfied

$dts(\ell, \alpha) > 0$ when C is falsified

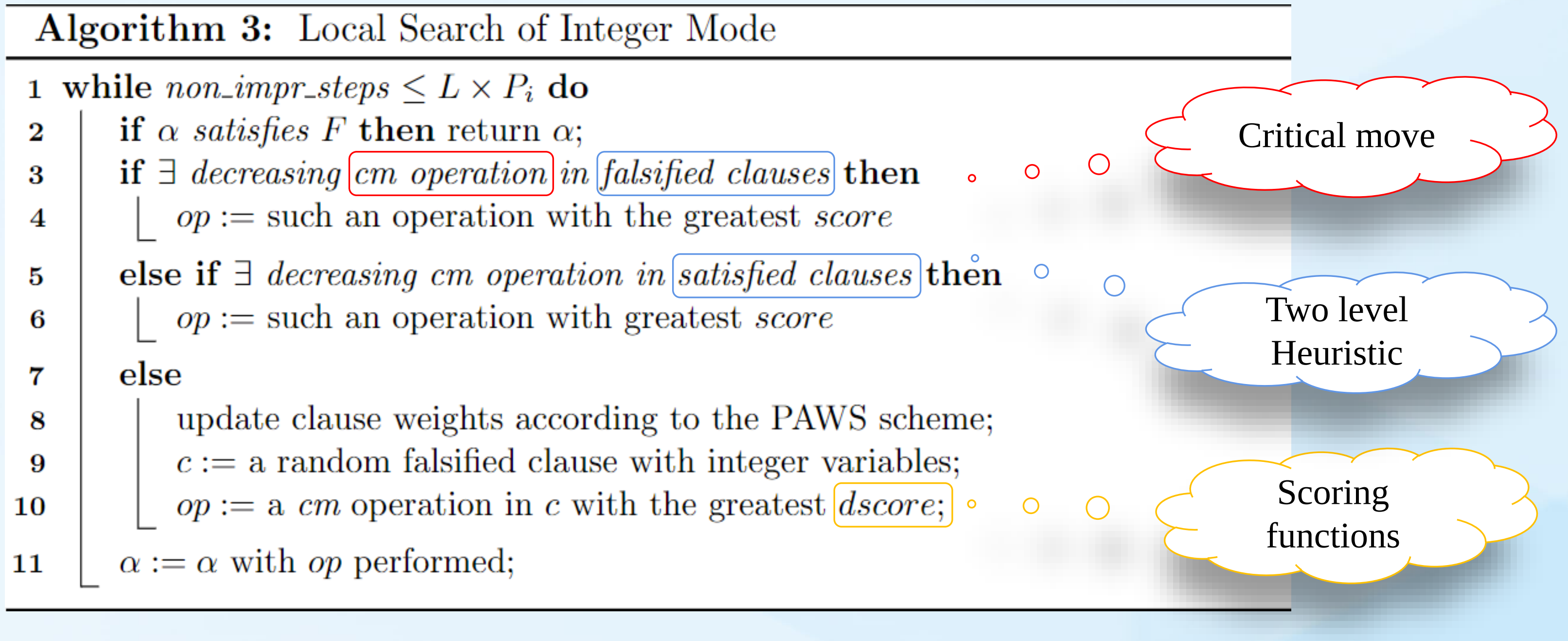


$$C = \ell_1 \vee l_2 \vee l_3 = (a - b \leq 1) \vee (b \leq -2) \vee (c \leq -3)$$
$$\alpha = \{a = b = c = 0\}$$

$$dscore(op) = \sum_{c \in F}(dts(c, \alpha) - dts(c, \alpha'))$$

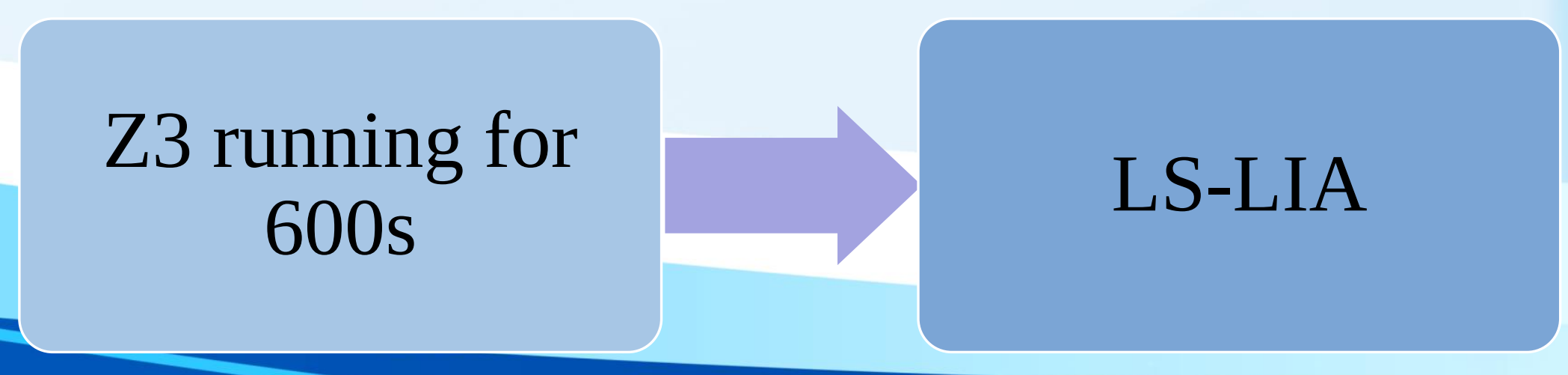
where α, α' denotes the assignment before and after performing op

Main Algorithm



```
Algorithm 3: Local Search of Integer Mode
while $non\_impr\_steps \leq L \times P_i$ do
    if $\alpha$ satisfies $F$ then return $\alpha$;
    if $\exists$ decreasing cm operation in falsified clauses then
        $op :=$ such an operation with the greatest score
    else if $\exists$ decreasing cm operation in satisfied clauses then
        $op :=$ such an operation with greatest score
    else
        update clause weights according to the PAWS scheme;
        $c :=$ a random falsified clause with integer variables;
        $op :=$ a cm operation in $c$ with the greatest dscore;
    $\alpha := \alpha$ with $op$ performed;
```

Combining with Z3: Z3+LS



Experiments

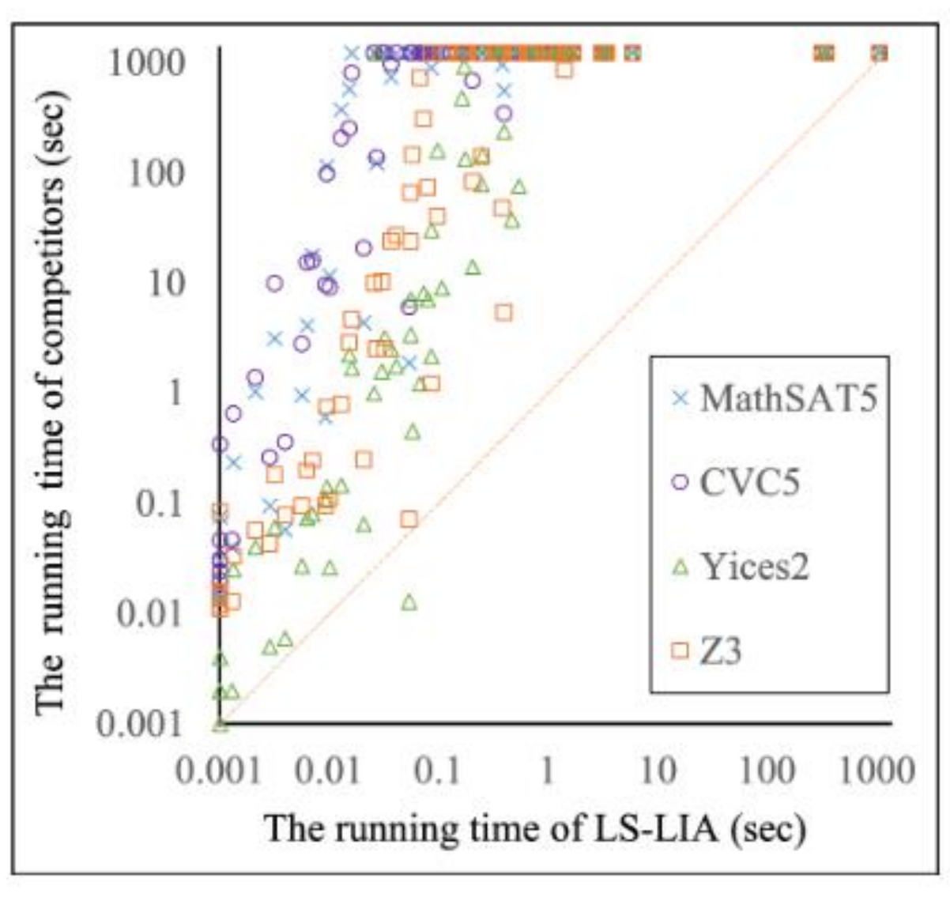
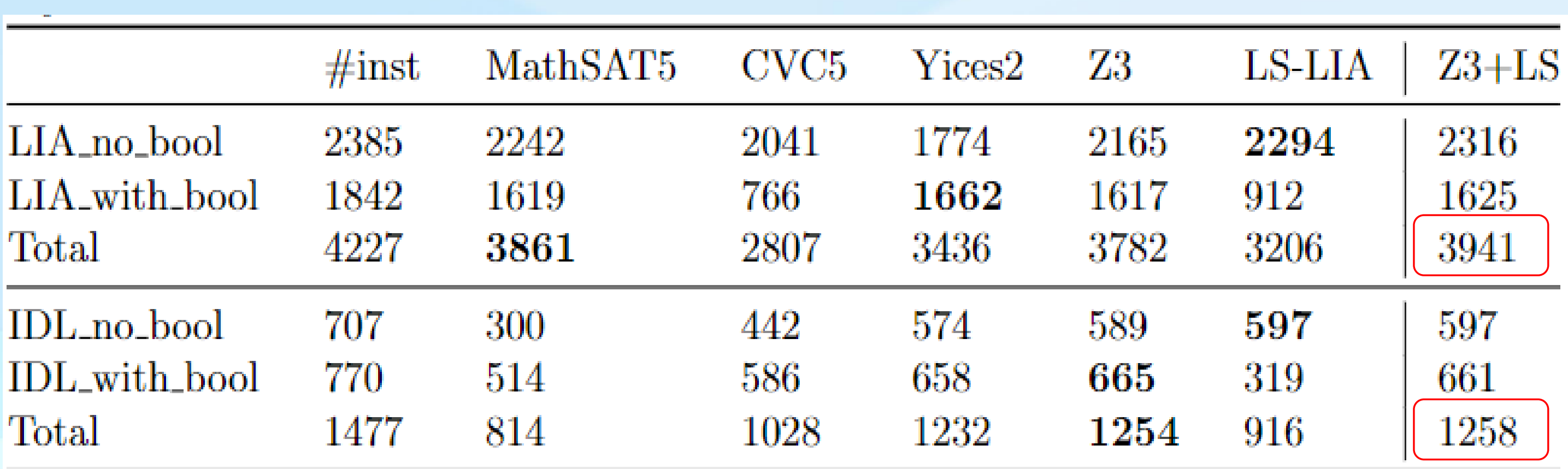


Fig. 4: Run time comparison on job shop scheduling instances.



	#inst	MathSAT5	CVC5	Yices2	Z3	LS-LIA	Z3+LS
LIA_no_bool	2385	2242	2041	1774	2165	**2294**	2316
LIA_with_bool	1842	1619	766	**1662**	1617	912	1625
Total	4227	**3861**	2807	3436	3782	3206	3941
IDL_no_bool	707	300	442	574	589	**597**	597
IDL_with_bool	770	514	586	658	**665**	319	661
Total	1477	814	1028	1232	**1254**	916	1258