On Reuse in Multi-Goal Test-Suite Generation for Software Product Lines

FOSD Meeting 2015
2015, May 13-16 – Traunkirchen, Austria

Johannes Bürdek,
Malte Lochau,
Stefan Bauregger

Andreas Holzer

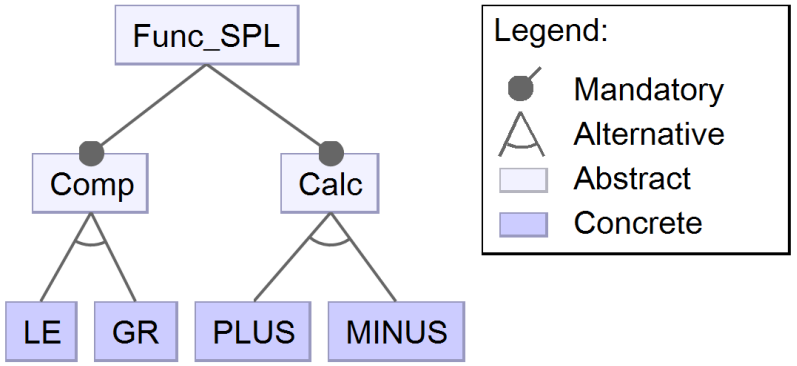
Alexander von Rhein,
Sven Apel

Dirk Beyer

Design For FUTURE IMOTEP

Software Product Line

```
1  int func-spl(int x, int y, int z) {
2      int a;
3      #ifdef LE
4      if (x < y)
5        a = x;
6      else
7        a = y;
8      #elsif GR
9      if (x > y)
10       a = x;
11     else
12       a = y;
13     #endif
14     #ifdef PLUS
15     z = z + a;
16     #elsif MINUS
17     z = z - a;
18     #endif
19     return z;
20 }
```



[*Conguration Lifting: Verication Meets Software Conguration.* Hendrick Post, Carsten Sinz: In: ASE. pp. 347-350. IEEE, 2008]

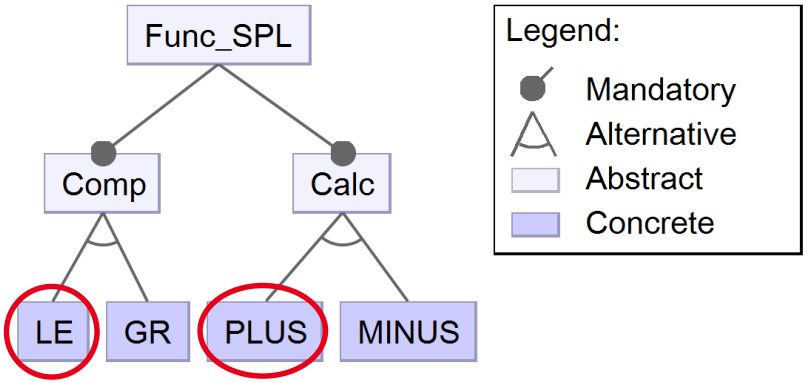
Software Product Line

```
1  int func-spl(int x, int y, int z) {
2      int a;

4      if (x < y)
5        a = x;
6      else
7        a = y;

15     z = z + a;

19     return z;
20 }
```



[*Conguration Lifting: Verication Meets Software Conguration.* Hendrick Post, Carsten Sinz: In: ASE. pp. 347-350. IEEE, 2008]

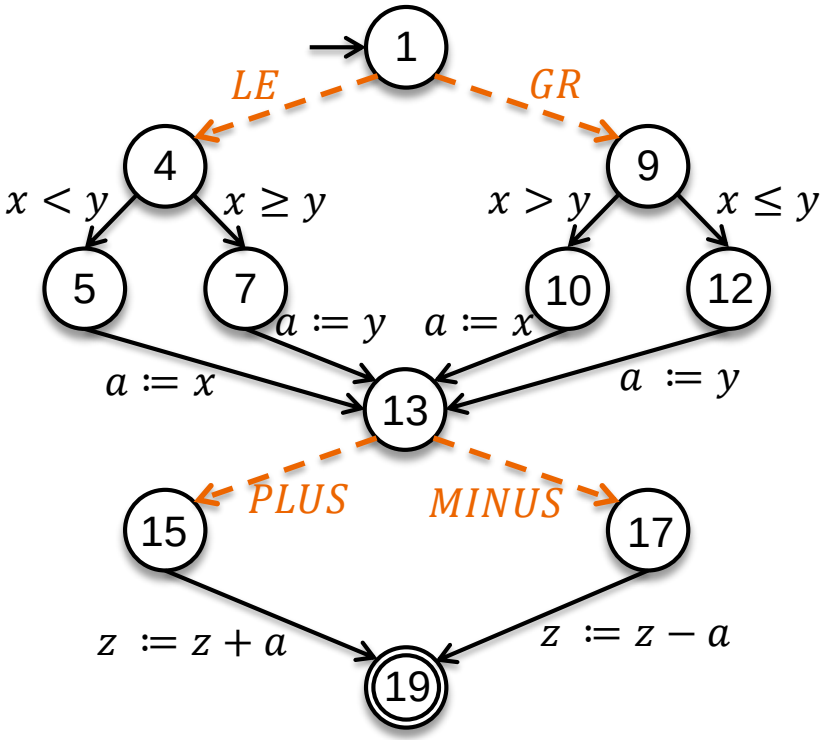
Test-Goal Specification

```
1  int func-spl(int x, int y, int z) {
2      int a;
3      #ifdef LE
4      if (x < y)
5        a = x;
6      else
7        a = y;
8      #elsif GR
9      if (x > y)
10       a = x;
11     else
12       a = y;
13     #endif
14     #ifdef PLUS
15     z = z + a;
16     #elsif MINUS
17     z = z - a;
18     #endif
19     return z;
20 }
```

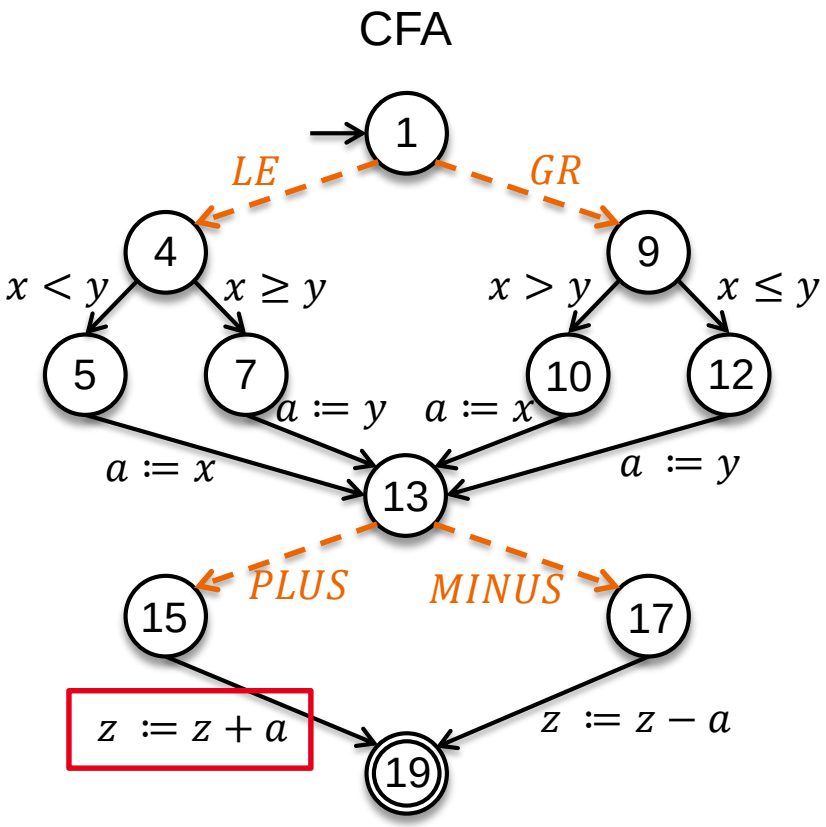
- Statement Coverage
  - Cover each statement with a test case
- Test case consists of
  - Input vector
  - Test oracle
- SPL test-goal coverage:
  - Each test goal has to be covered by at least one test case in every product where it is reachable

CFA for func-spl

```
1  int func-spl(int x, int y, int z) {
2      int a;
3      #ifdef LE
4      if (x < y)
5        a = x;
6      else
7        a = y;
8      #elsif GR
9      if (x > y)
10       a = x;
11     else
12       a = y;
13     #endif
14     #ifdef PLUS
15     z = z + a;
16     #elsif MINUS
17     z = z - a;
18     #endif
19     return z;
20 }
```

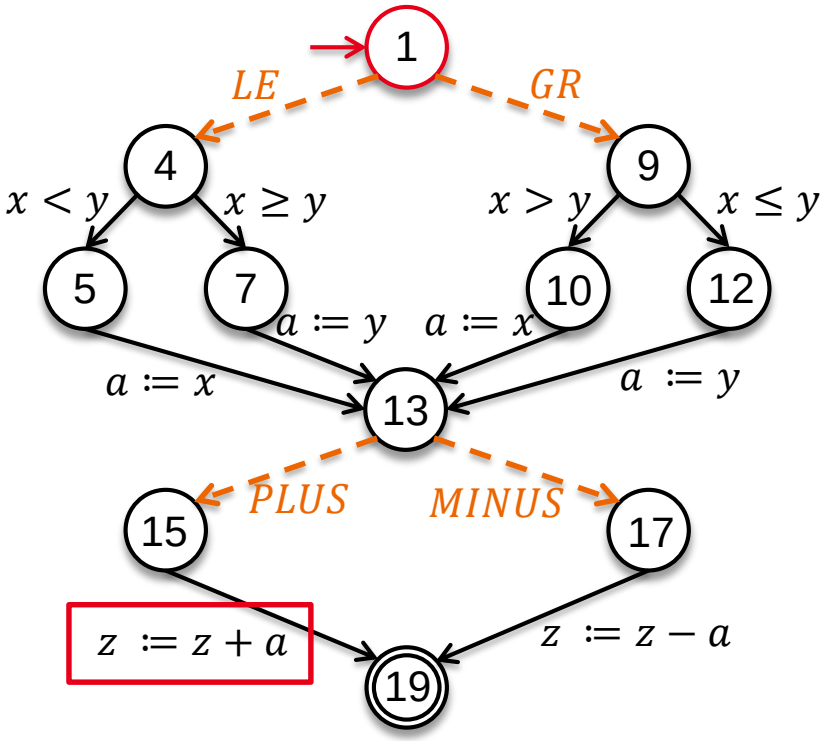


SPL Test-Case Generation with Reuse of Reachability Informations among Products

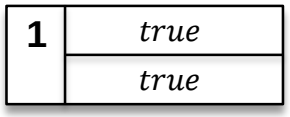


SPL Test-Case Generation with Reuse of Reachability Informations among Products

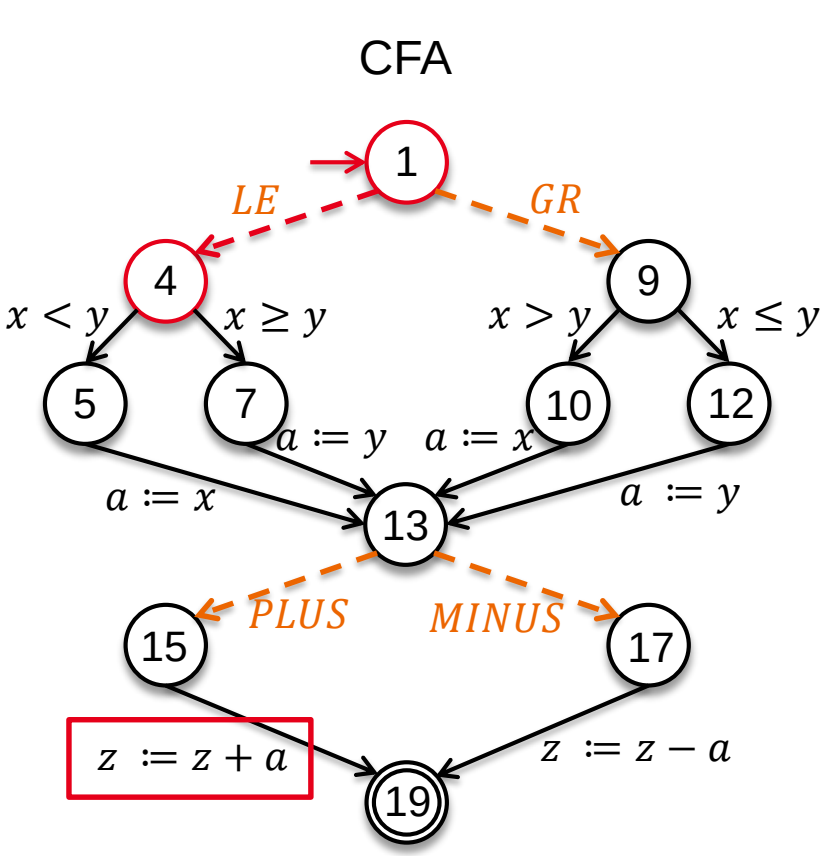
CFA

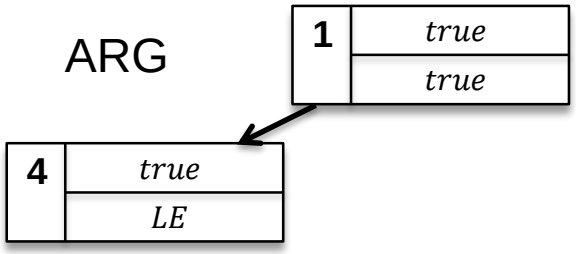


ARG

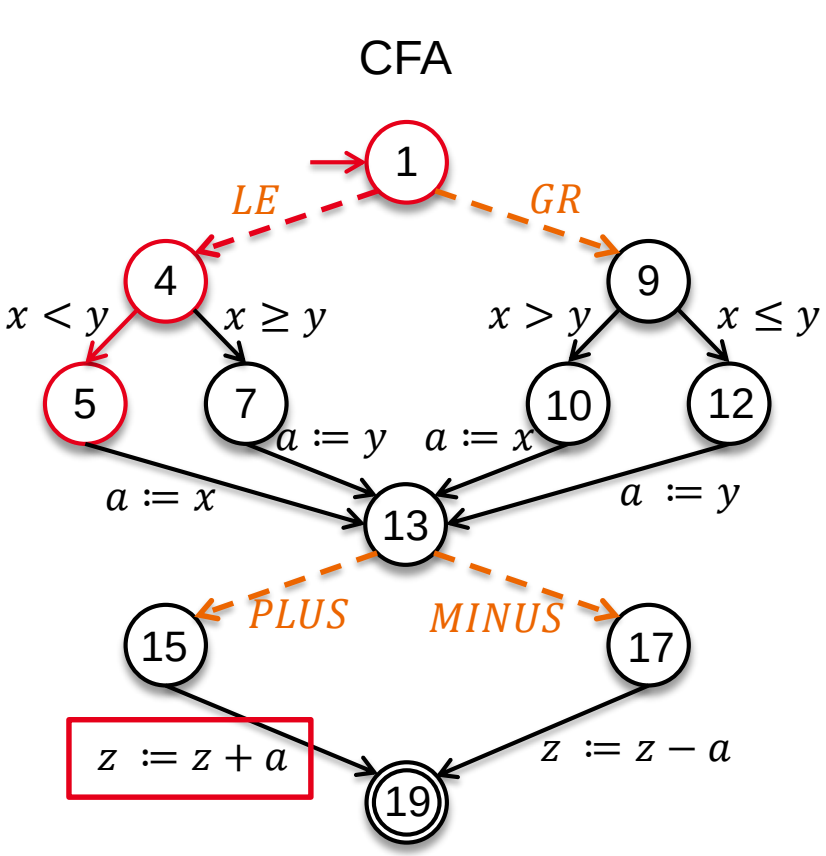


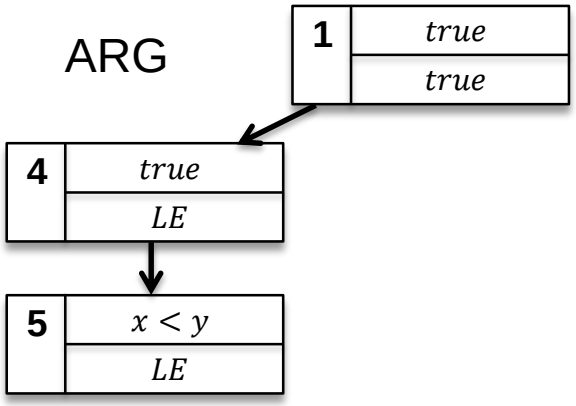
SPL Test-Case Generation with Reuse of Reachability Informations among Products



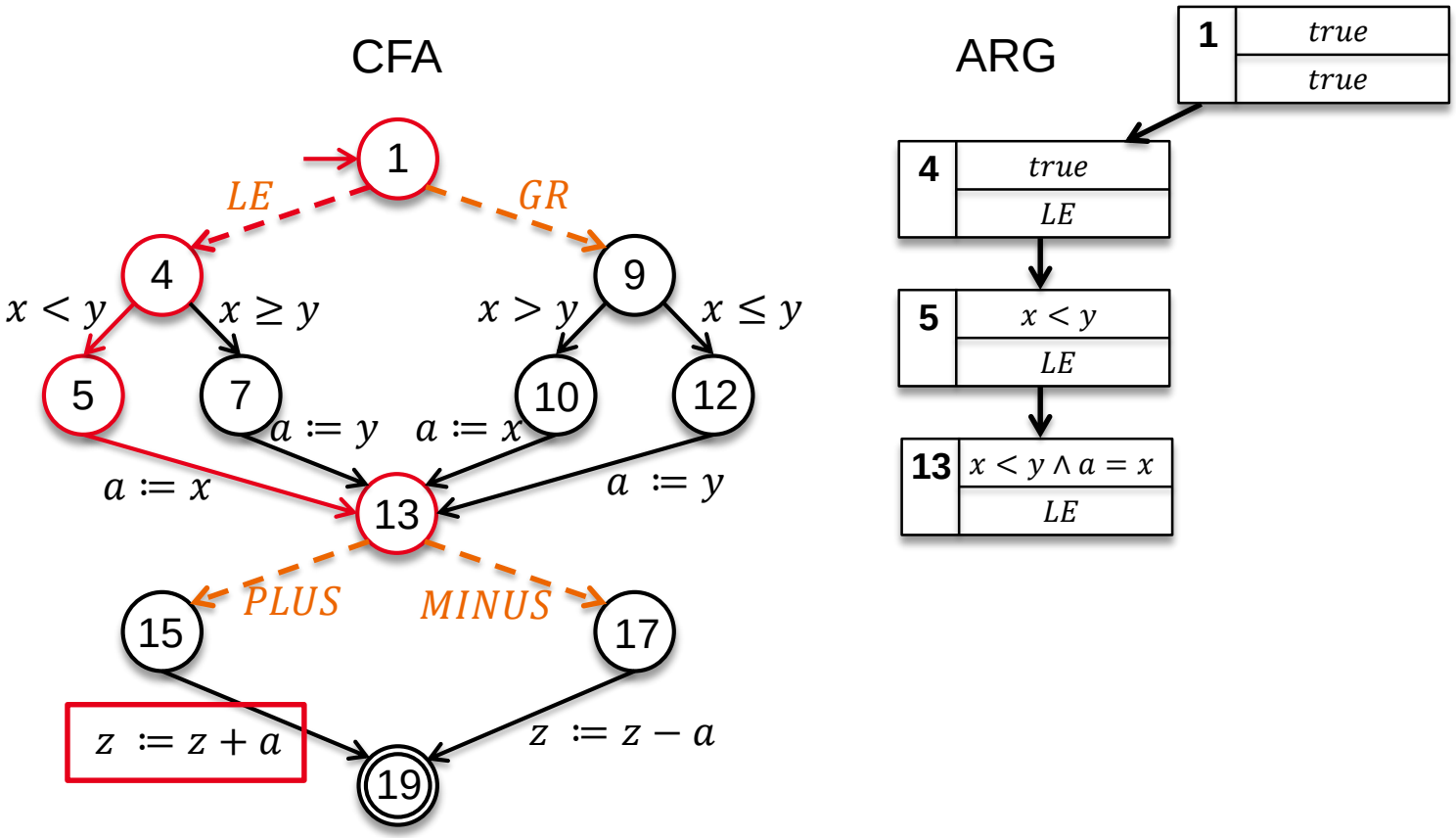


SPL Test-Case Generation with Reuse of Reachability Informations among Products

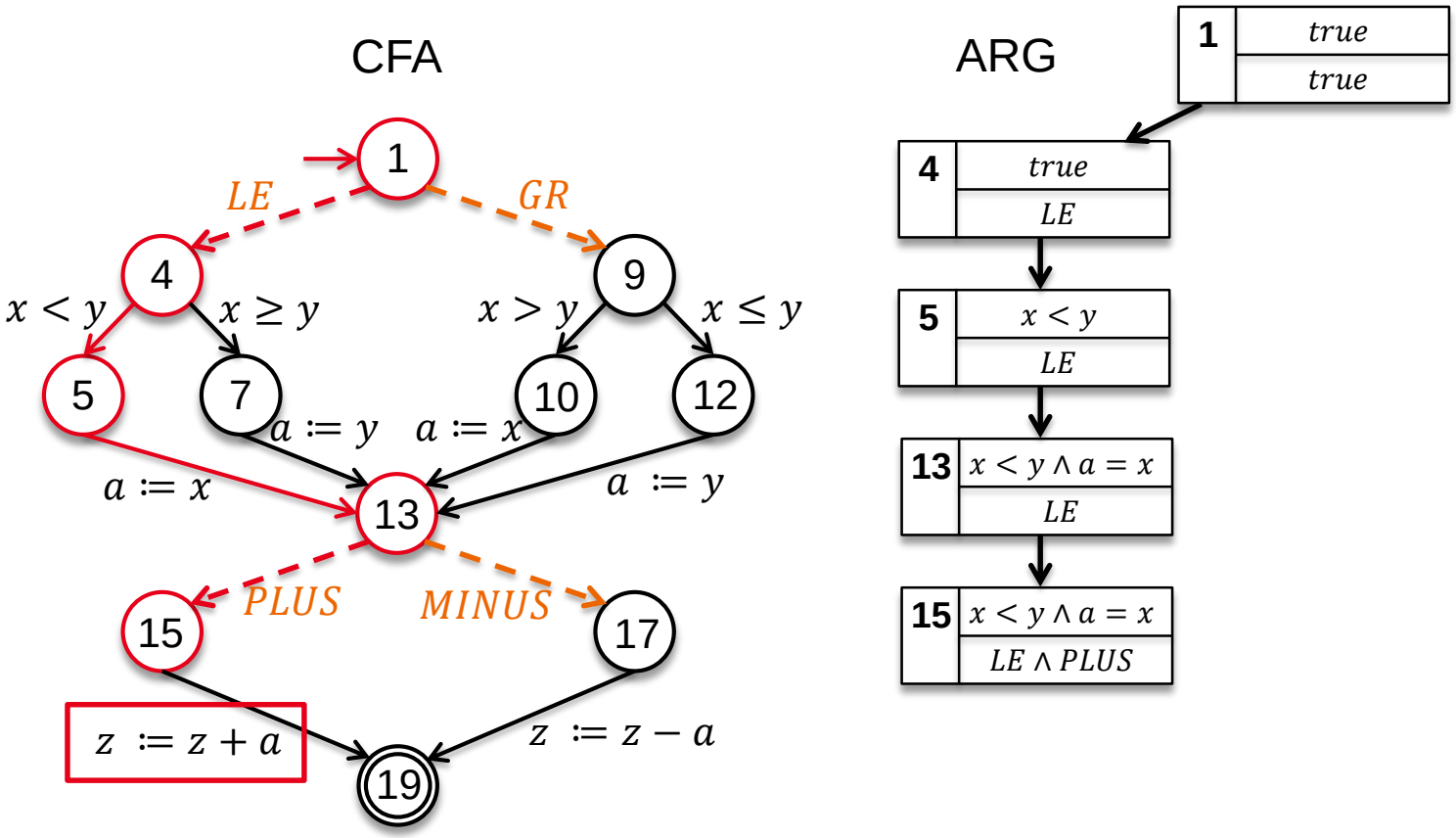




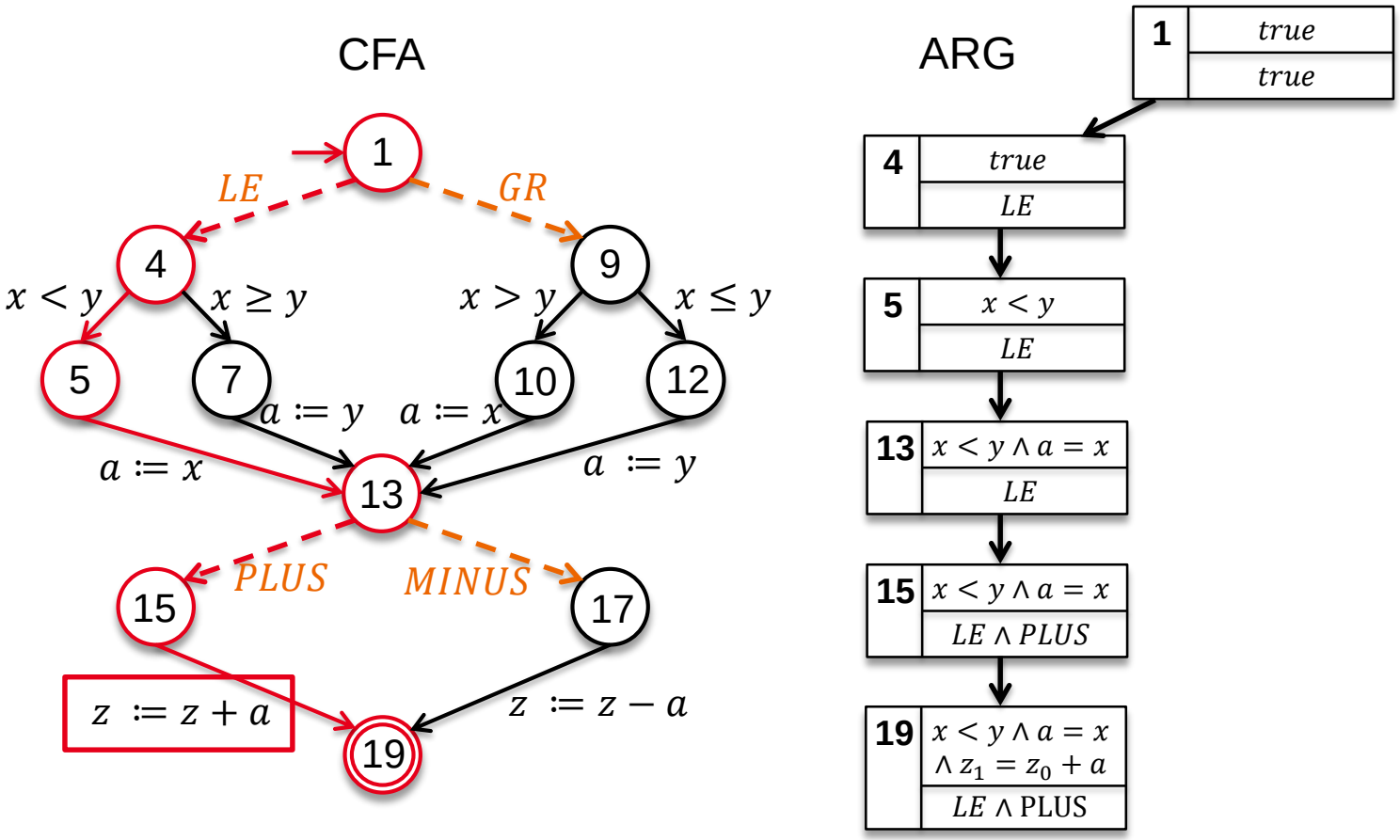
SPL Test-Case Generation with Reuse of Reachability Informations among Products



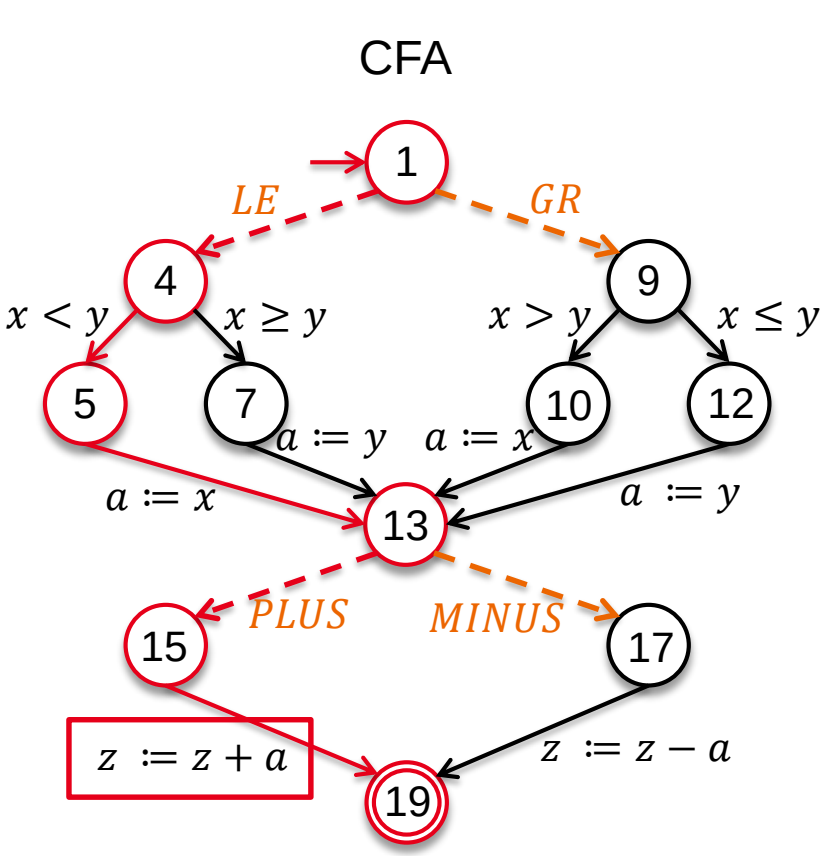
SPL Test-Case Generation with Reuse of Reachability Informations among Products

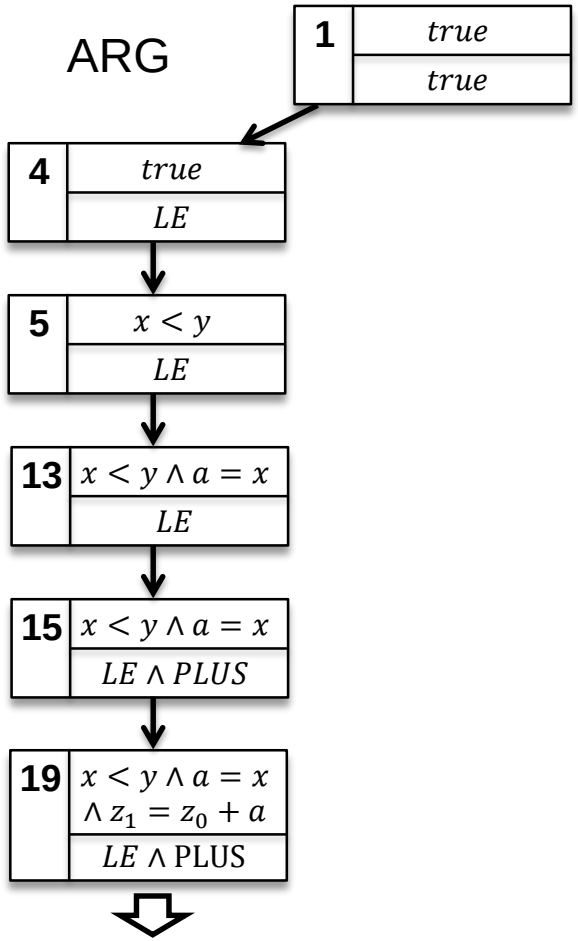


SPL Test-Case Generation with Reuse of Reachability Informations among Products



SPL Test-Case Generation with Reuse of Reachability Informations among Products

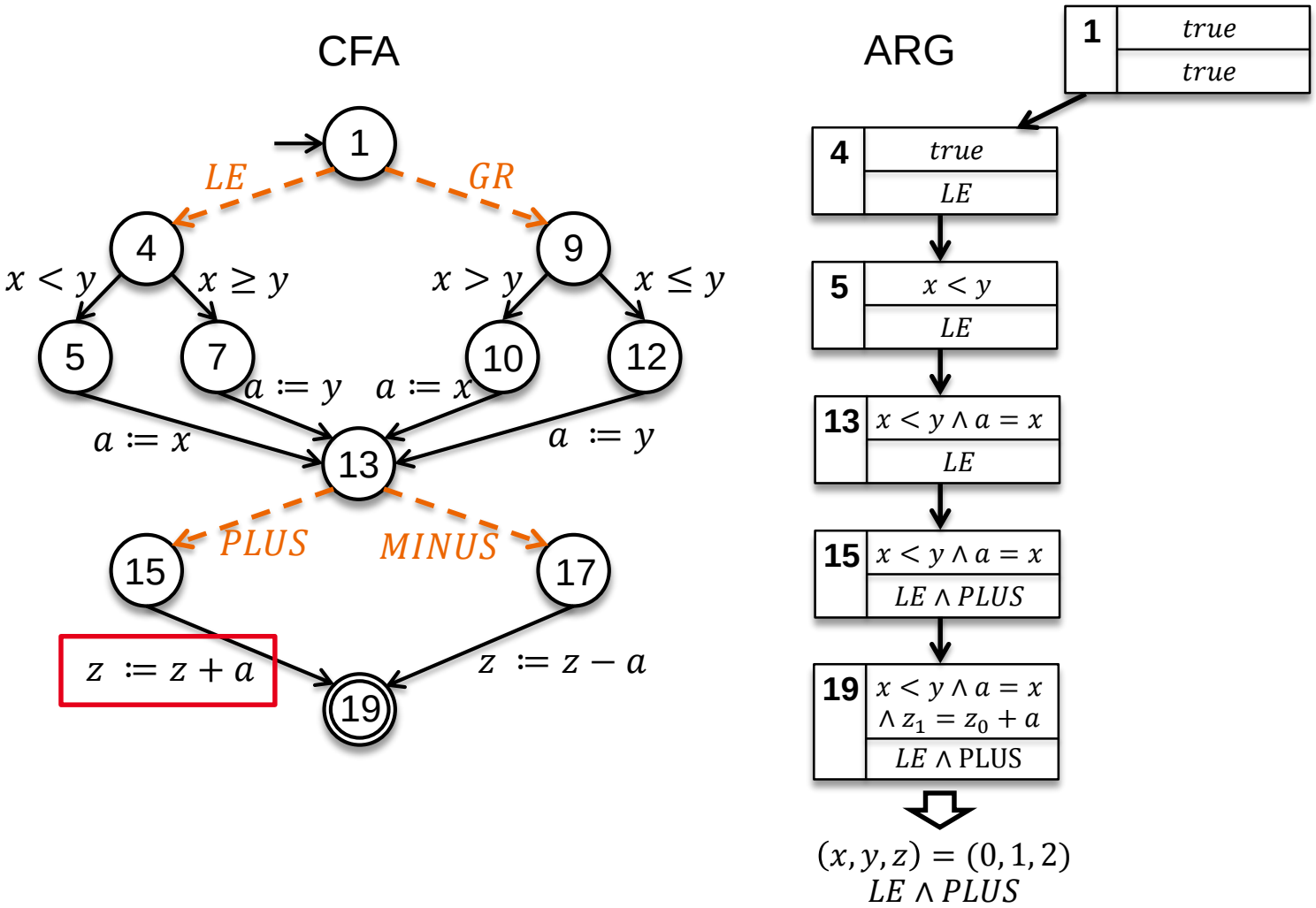




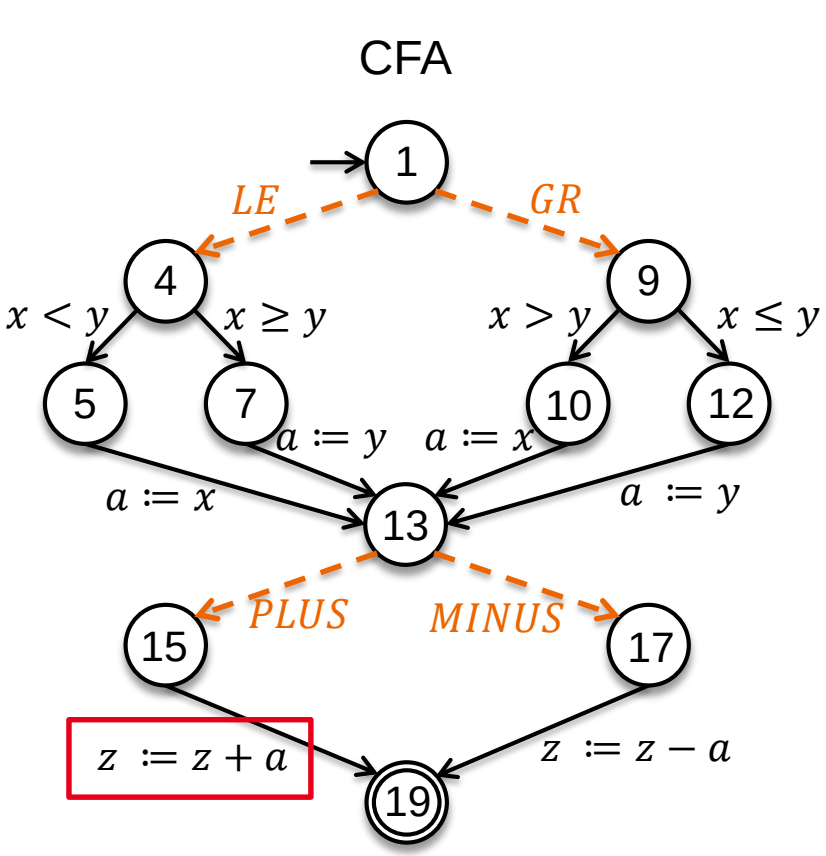
Input Vector → $(x, y, z) = (0, 1, 2)$

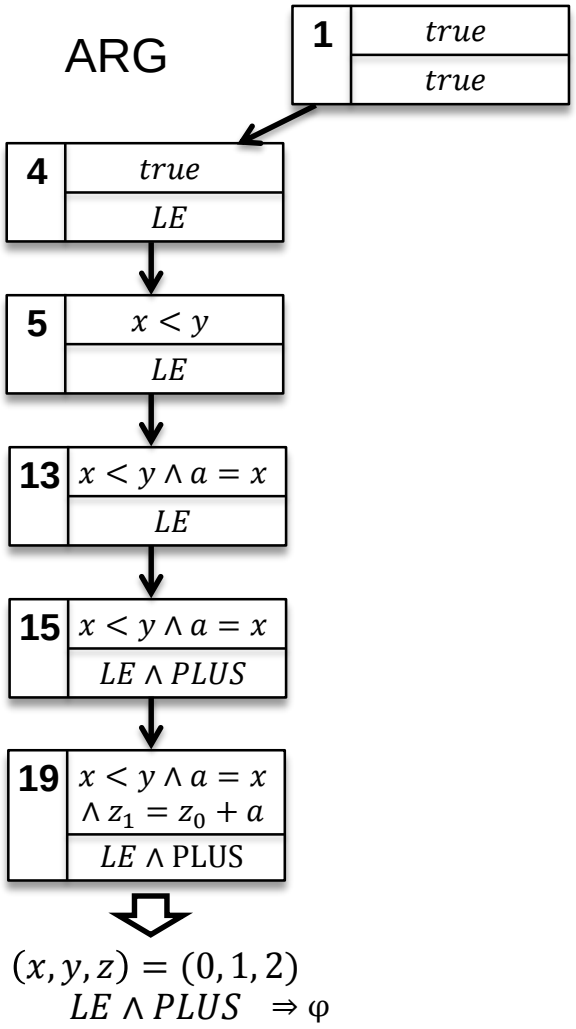
Presence Condition → $LE \wedge PLUS$

SPL Test-Case Generation with Reuse of Reachability Informations among Products

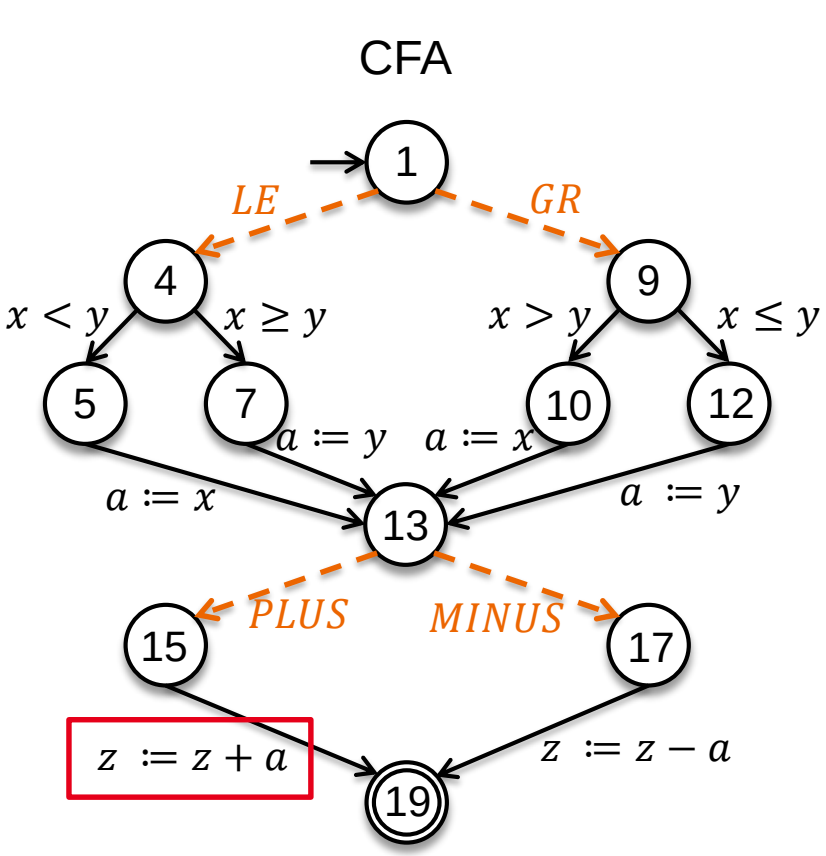


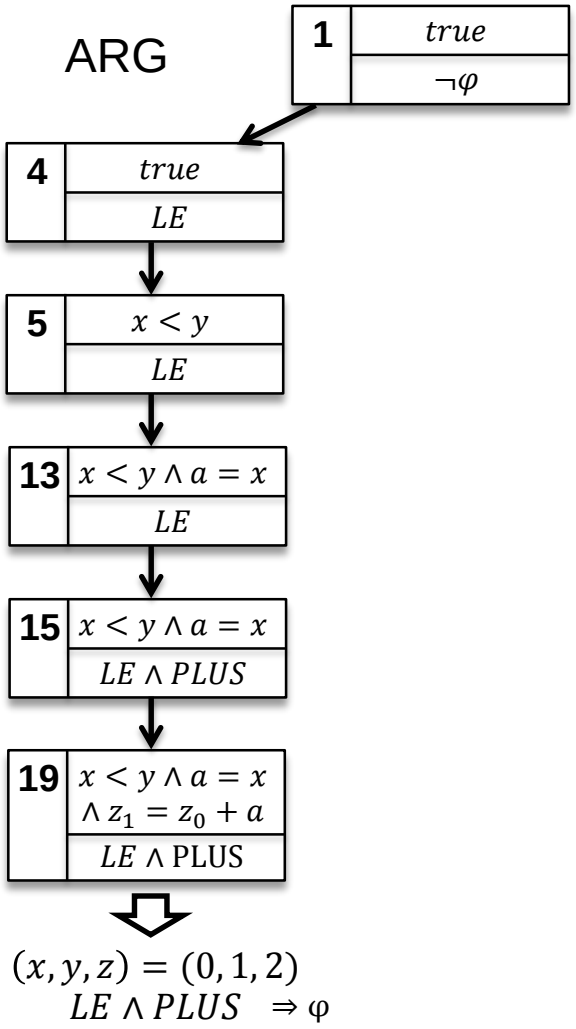
SPL Test-Case Generation with Reuse of Reachability Informations among Products



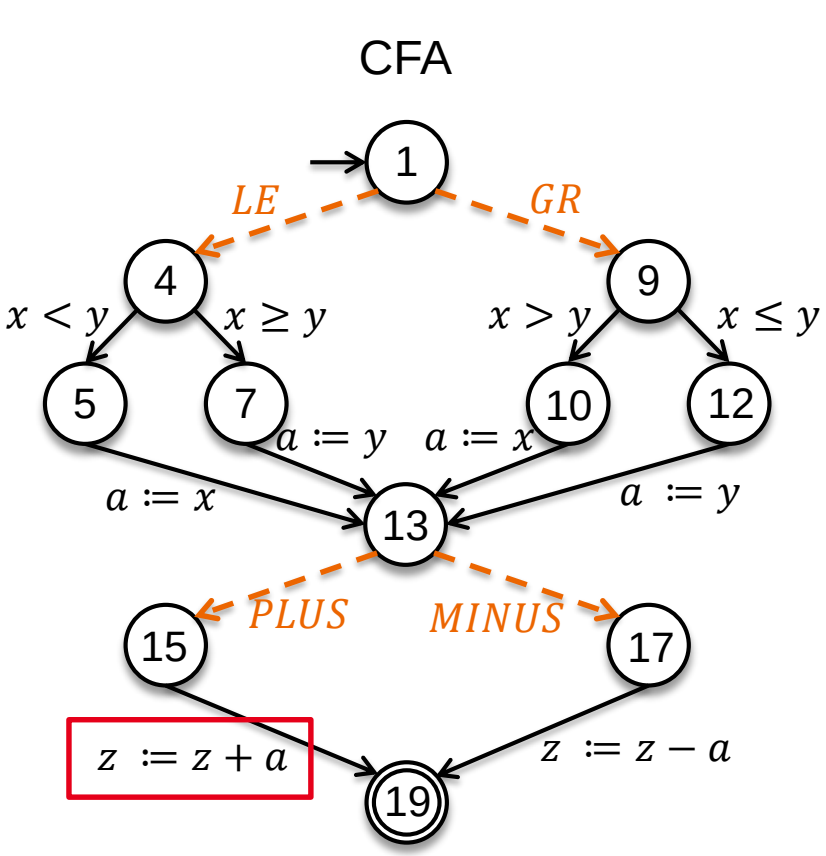


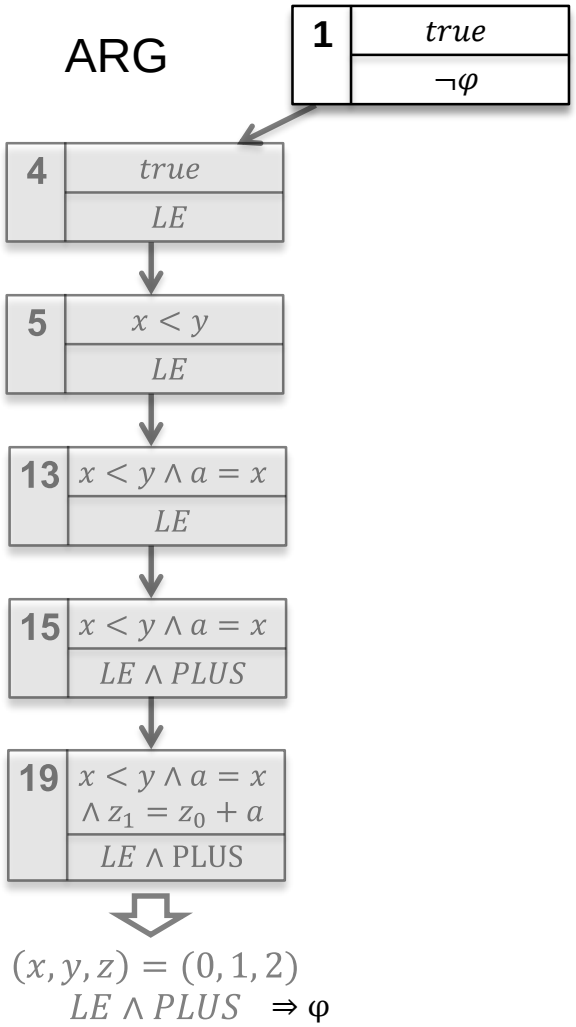
SPL Test-Case Generation with Reuse of Reachability Informations among Products



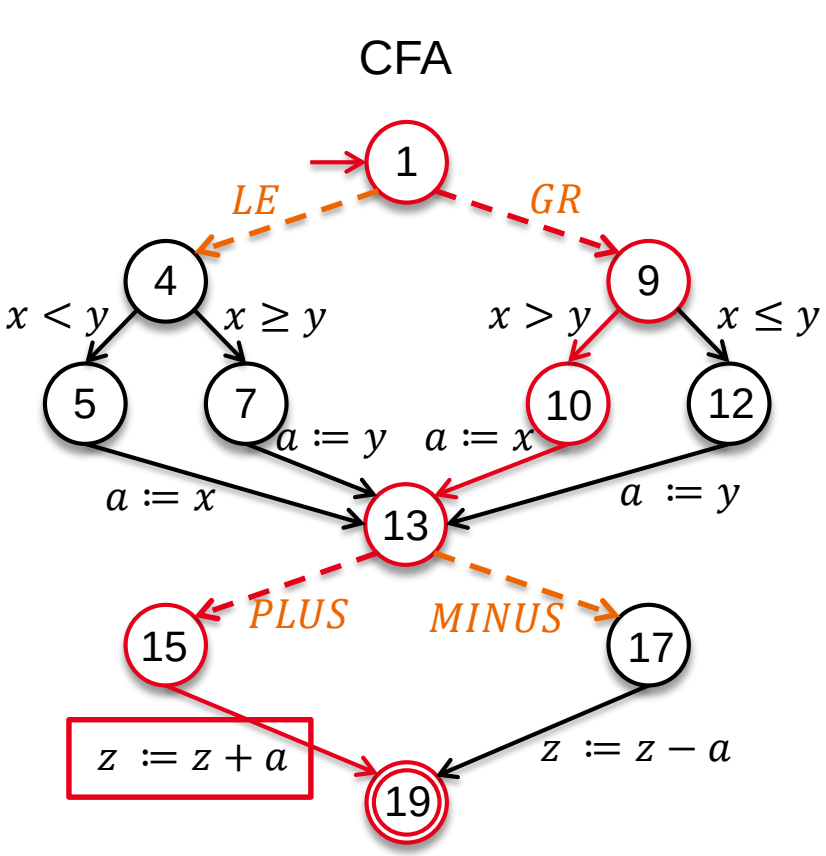


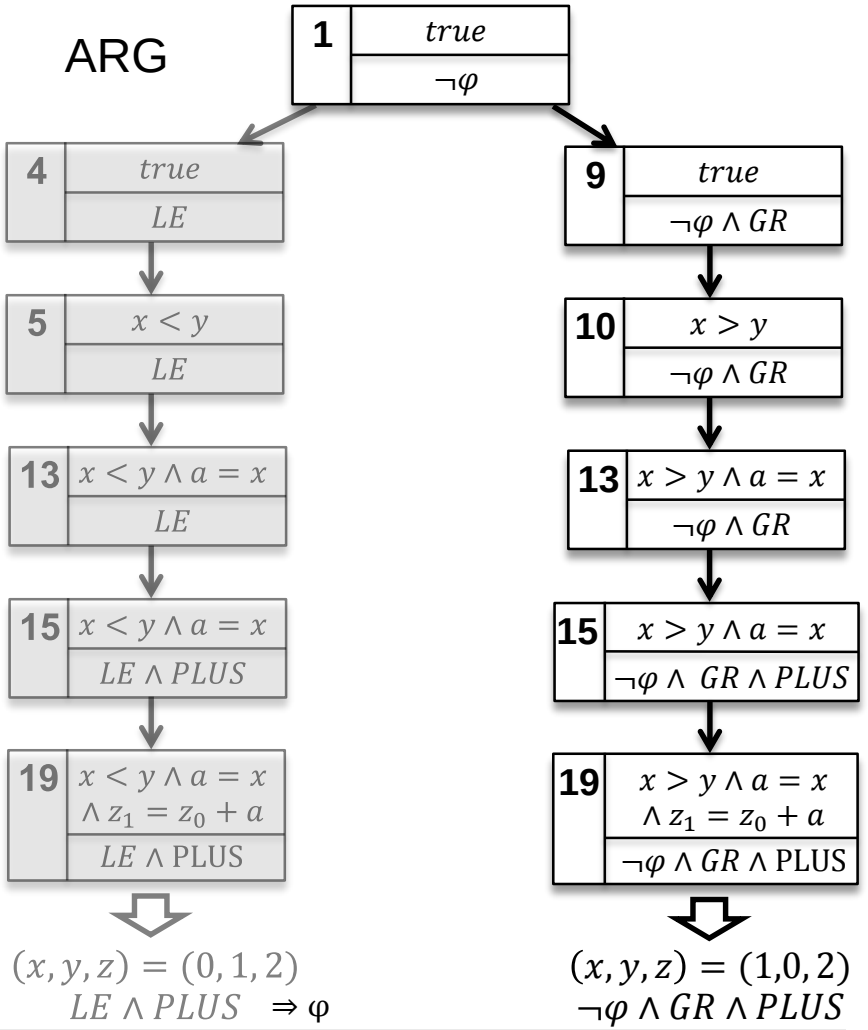
SPL Test-Case Generation with Reuse of Reachability Informations among Products



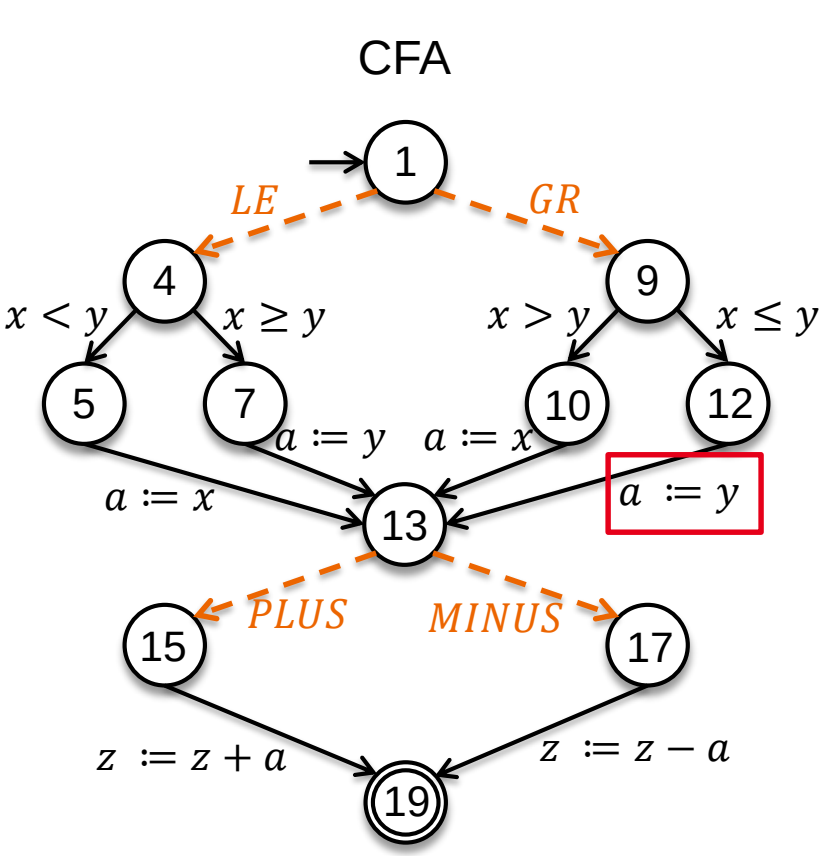


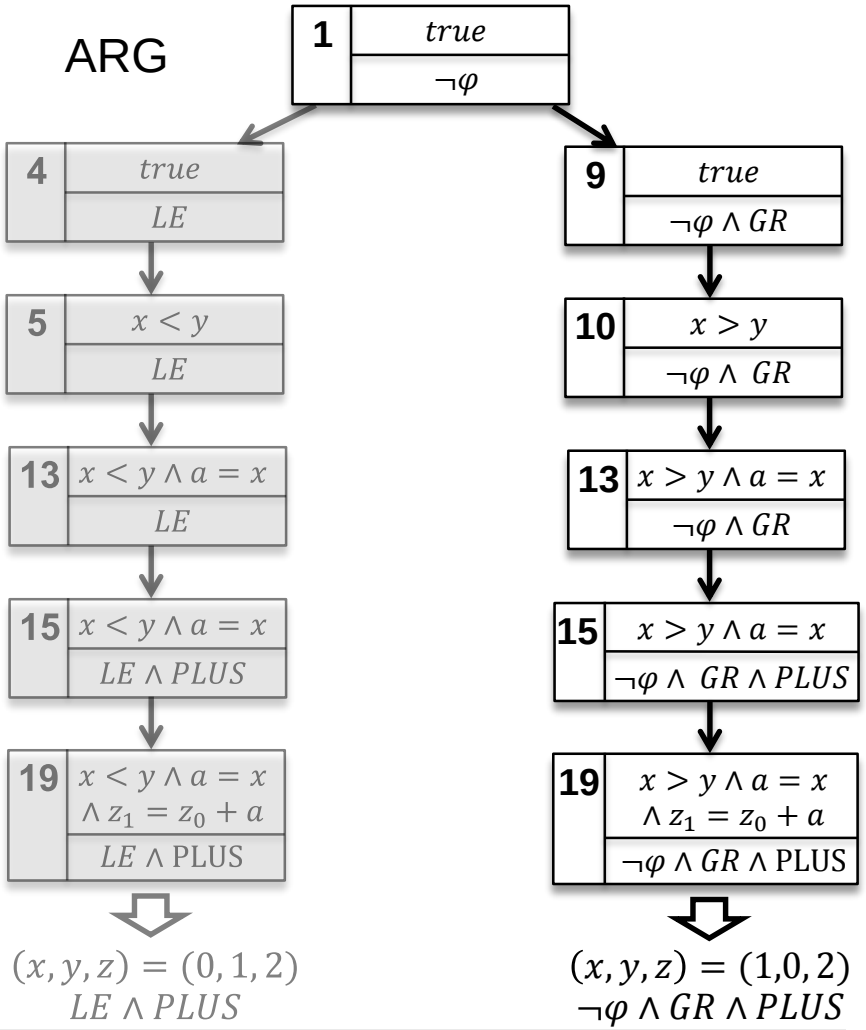
SPL Test-Case Generation with Reuse of Reachability Informations among Products



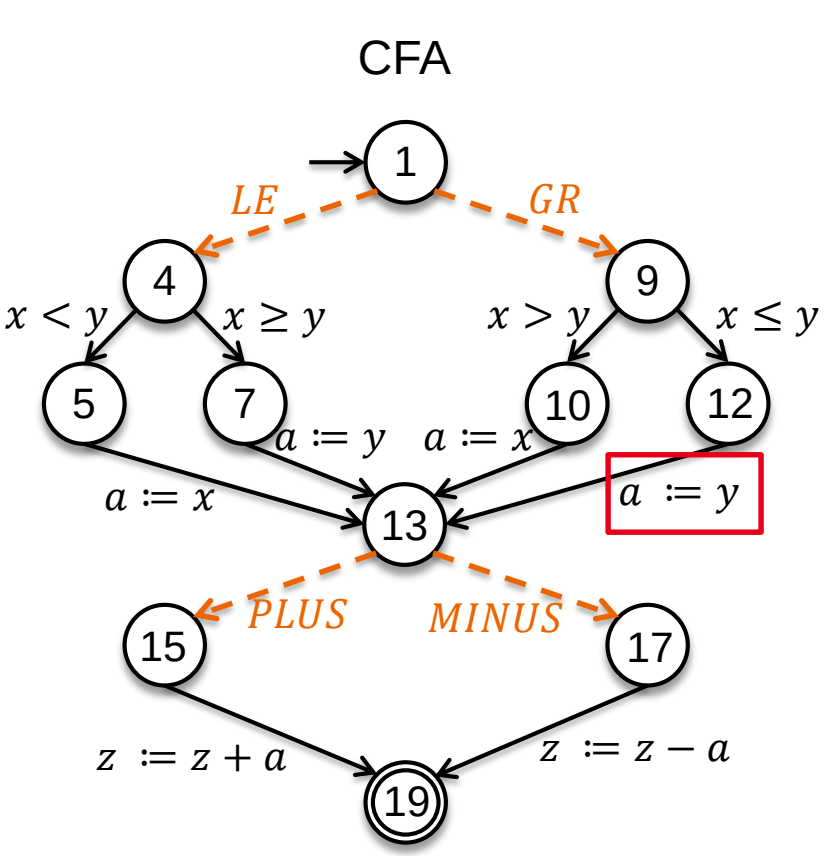


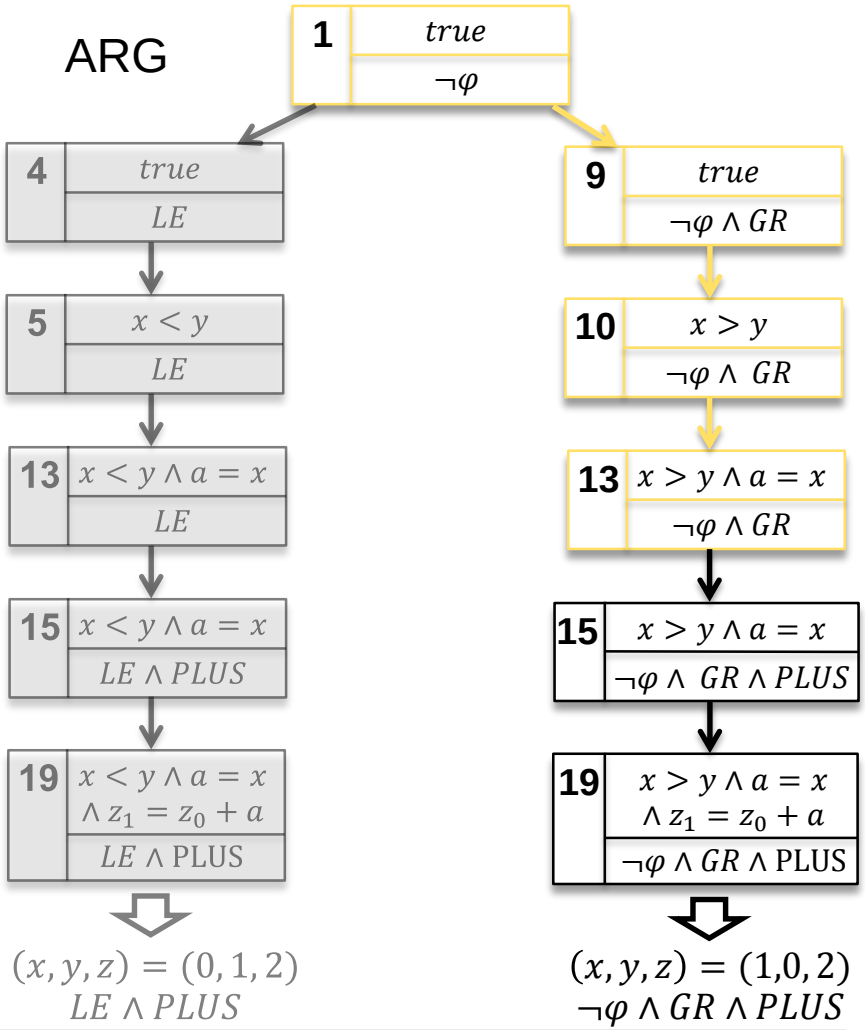
Reuse of Reachability Informations among Test Goals



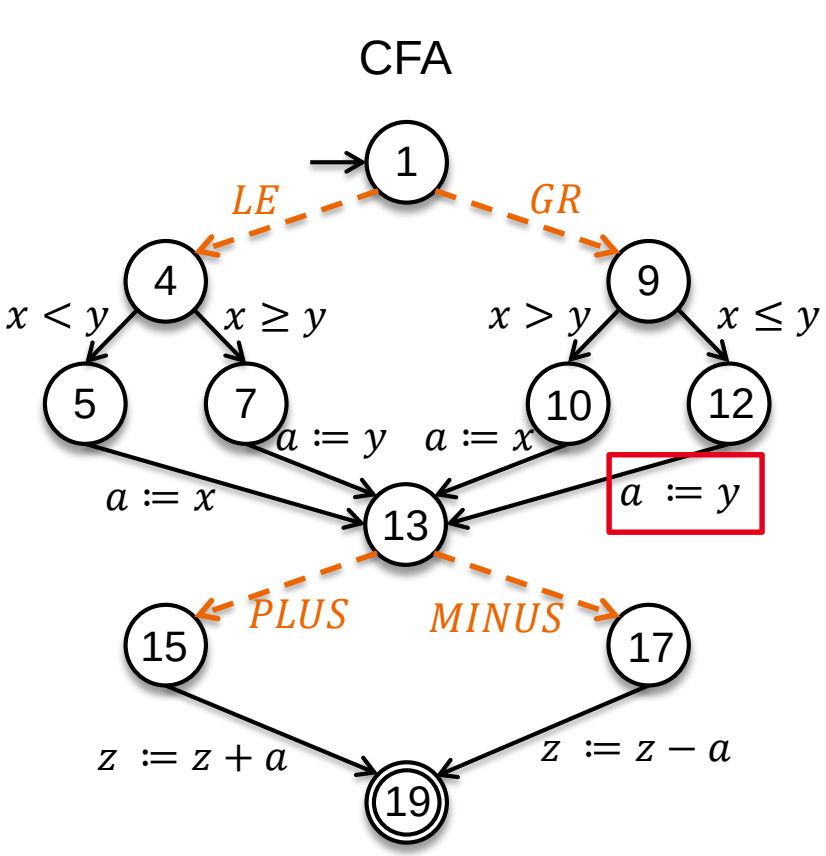


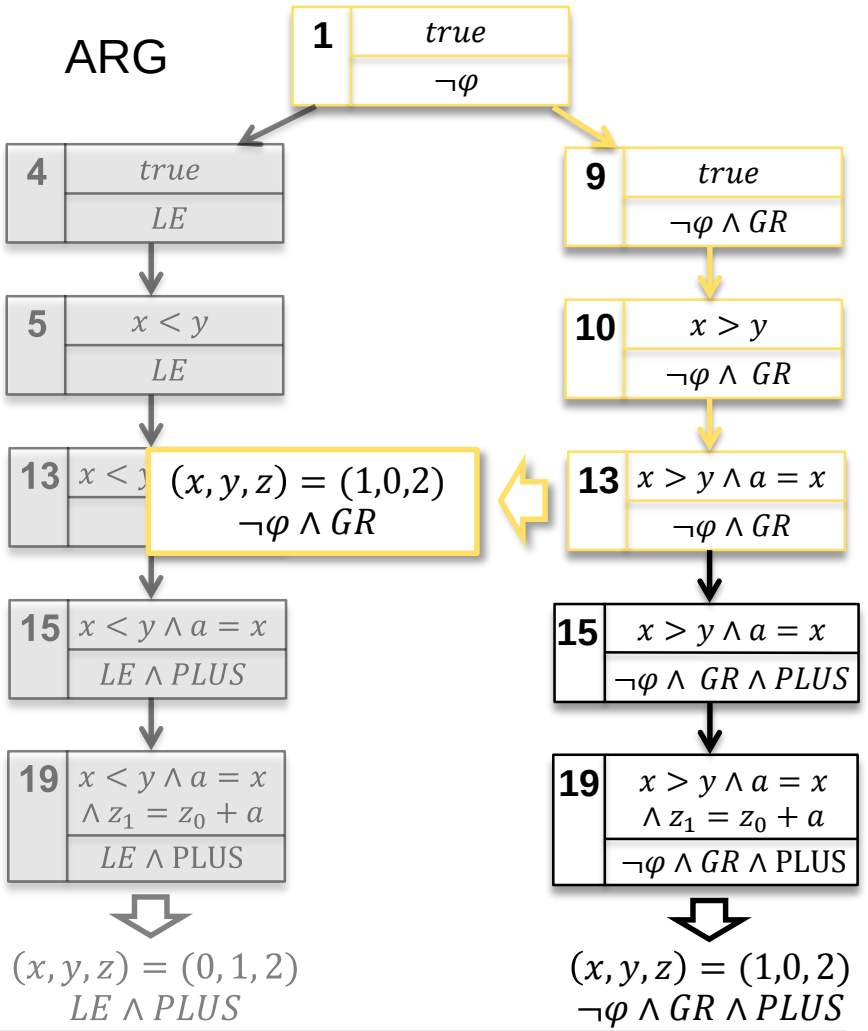
Reuse of Reachability Informations among Test Goals



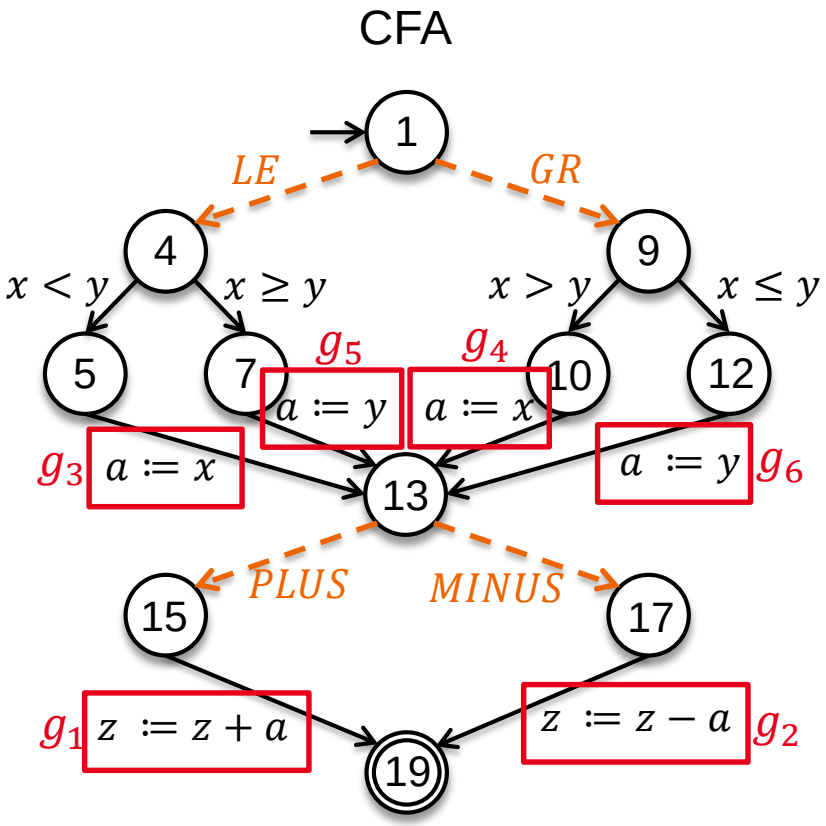


Reuse of Reachability Informations among Test Goals





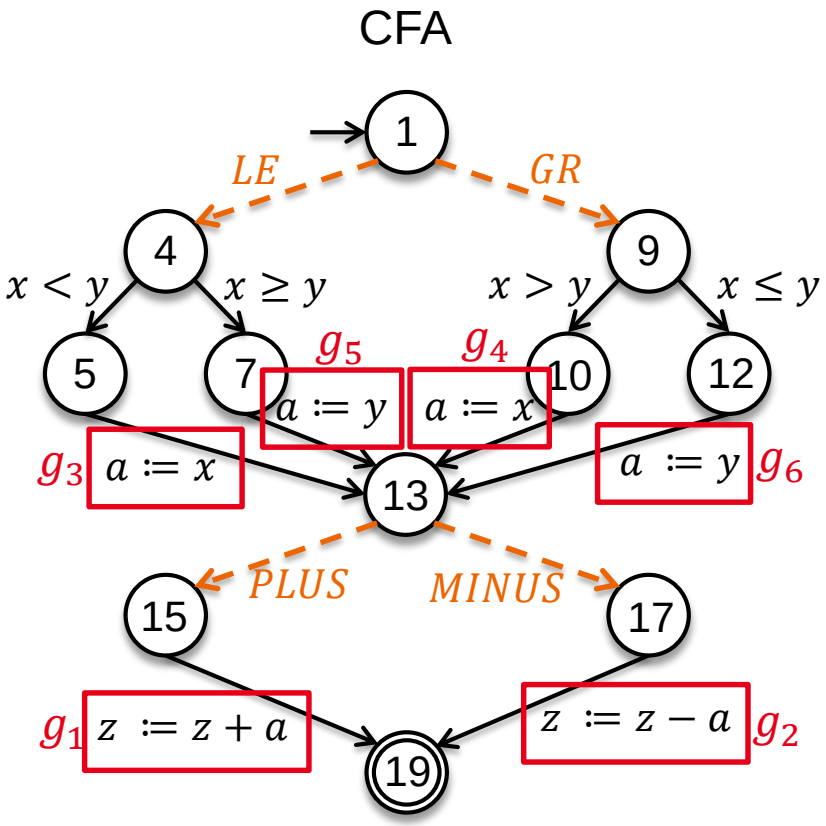
Complete SPL Test Suite



	g_1	g_2	g_3	g_4	g_5	g_6
P_1	tc_1				tc_3'	tc_1'
P_2	tc_2		tc_2'	tc_4'		
P_3		tc_3			tc_3'	tc_1'
P_4		tc_4	tc_2'	tc_4'		

$P_1: PLUS \wedge LE$
$P_2: PLUS \wedge GR$
$P_3: MINUS \wedge LE$
$P_4: MINUS \wedge GR$

Complete SPL Test Suite

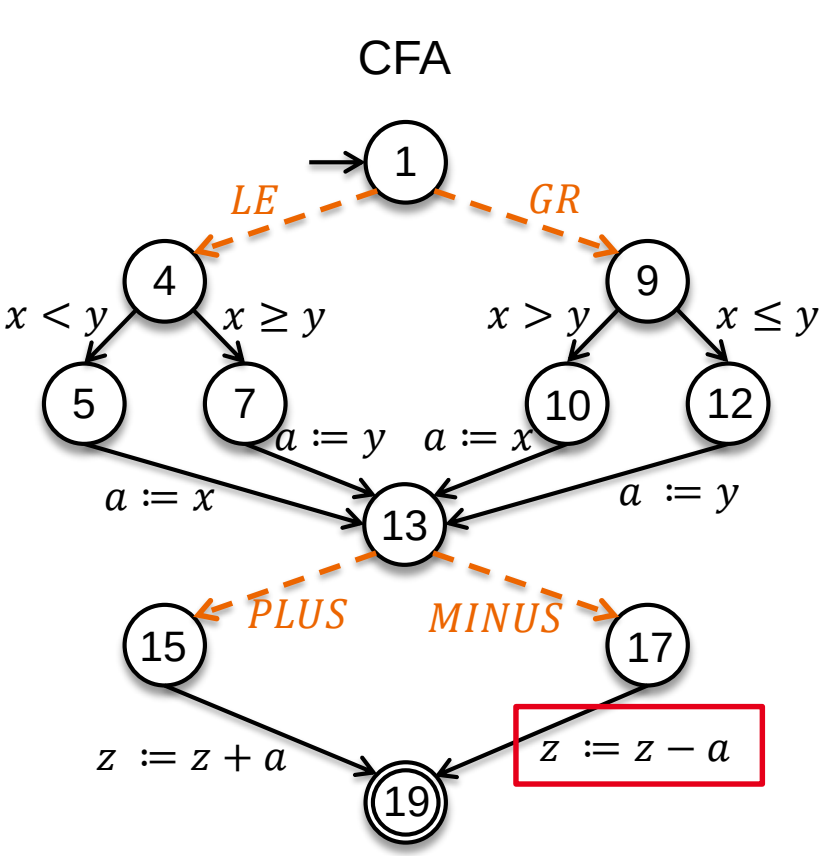


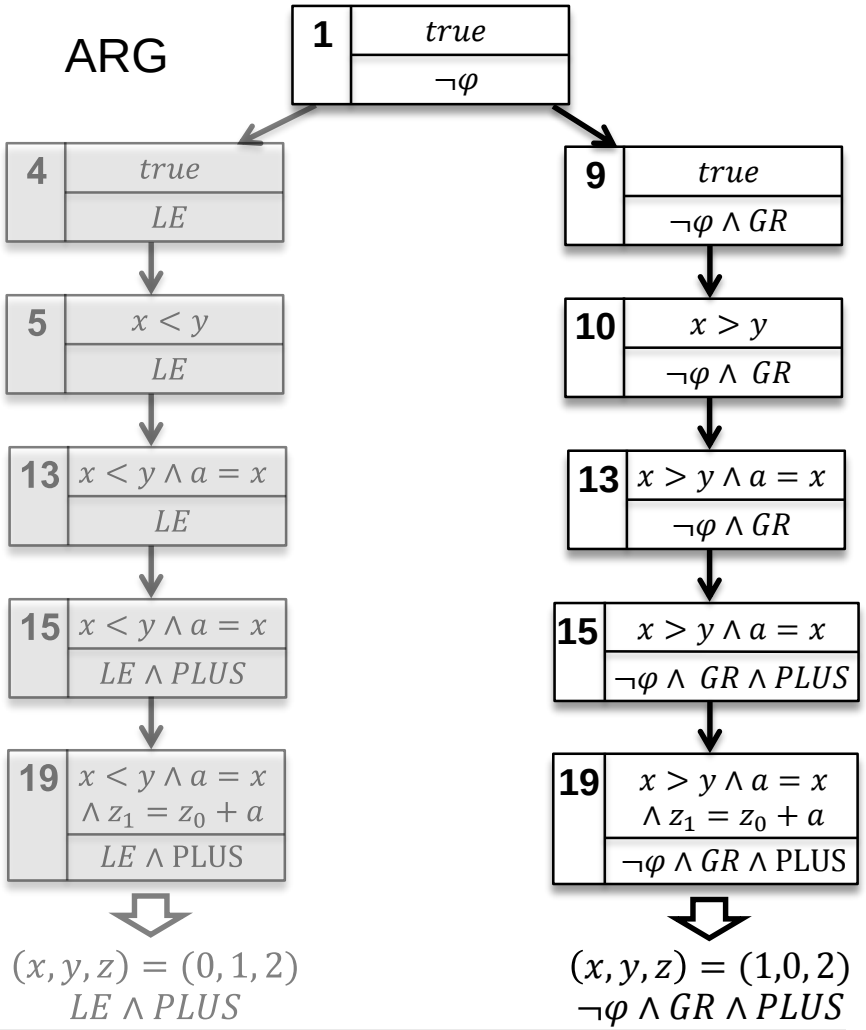
	g_1	g_2	g_3	g_4	g_5	g_6
P_1	tc_1				tc_3'	tc_1'
P_2	tc_2		tc_2'	tc_4'		
P_3		tc_3			tc_3'	tc_1'
P_4		tc_4	tc_2'	tc_4'		

$P_1: PLUS \wedge LE$
$P_2: PLUS \wedge GR$
$P_3: MINUS \wedge LE$
$P_4: MINUS \wedge GR$

TC	Presence Condition
tc_1	$GR \wedge PLUS$
tc_1'	GR

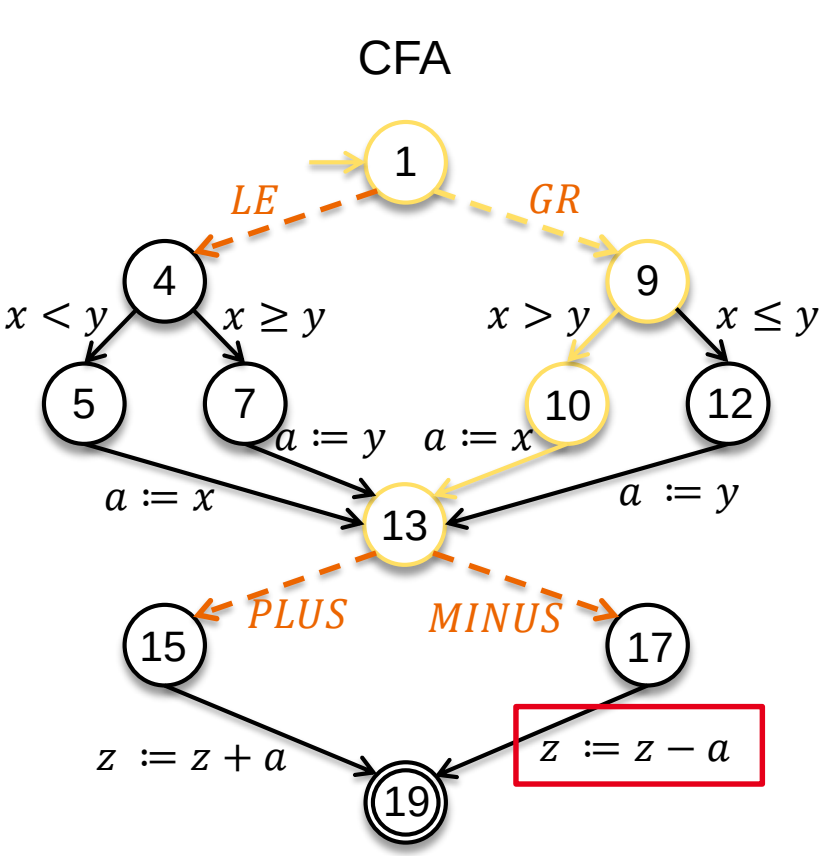
Ongoing Work: ARG Reuse

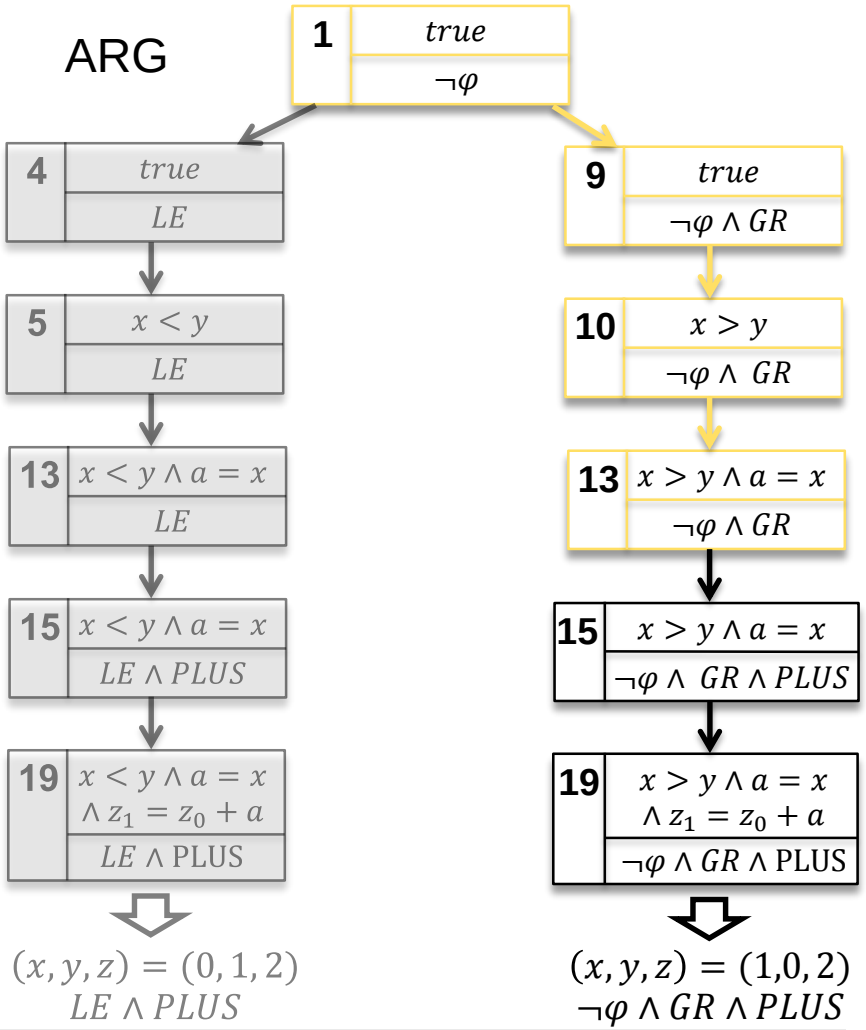




[*Information Reuse for Multi-goal Reachability Analyses.* Dirk Beyer, Andreas Holzer, Michael Tautschnig, and Helmut Veith. In Proc. ESOP, Springer, 2013]

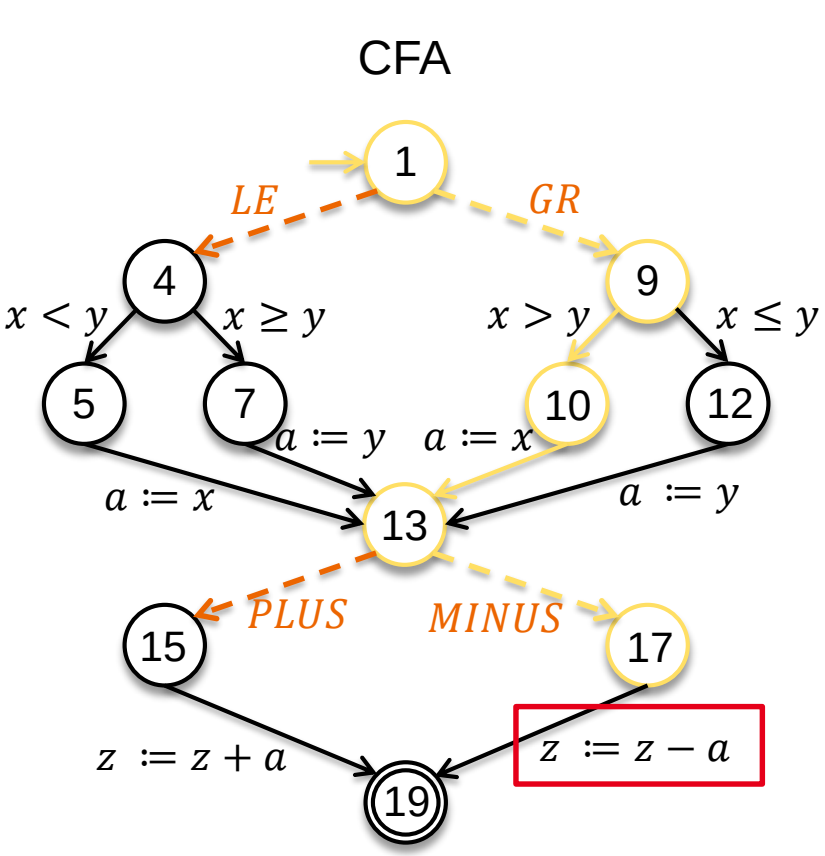
Ongoing Work: ARG Reuse

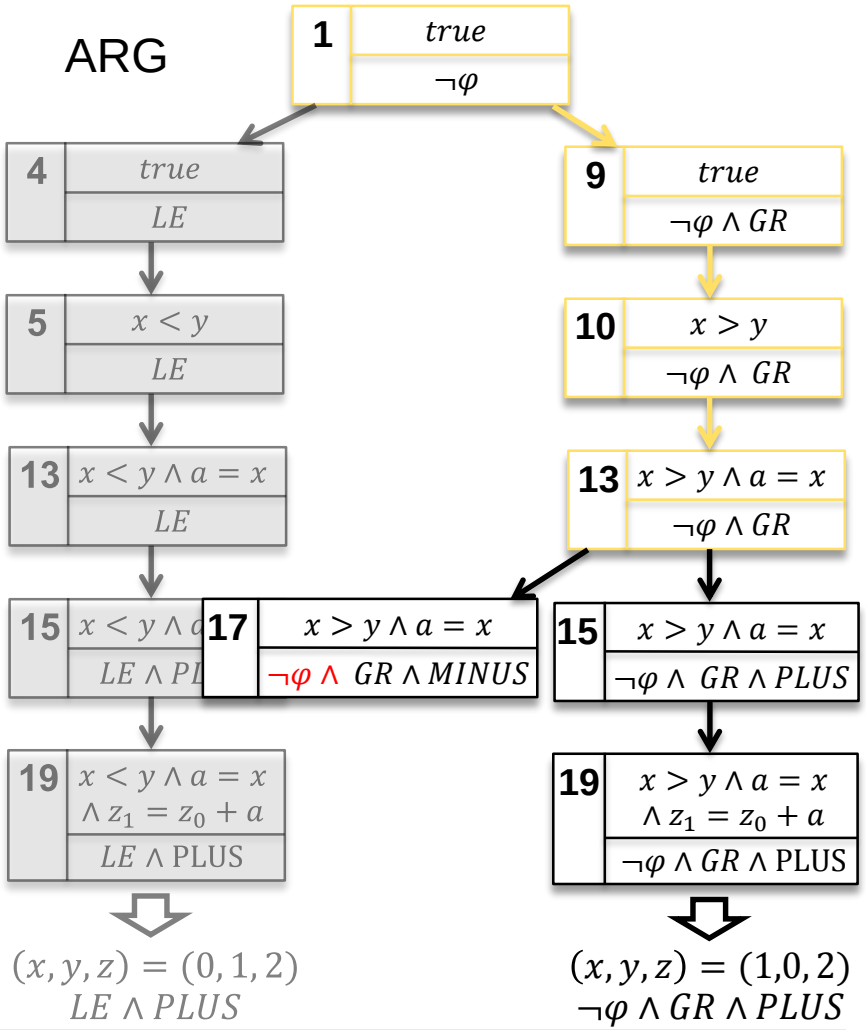




[*Information Reuse for Multi-goal Reachability Analyses.* Dirk Beyer, Andreas Holzer, Michael Tautschnig, and Helmut Veith. In Proc. ESOP, Springer, 2013]

Ongoing Work: ARG Reuse





[*Information Reuse for Multi-goal Reachability Analyses.* Dirk Beyer, Andreas Holzer, Michael Tautschnig, and Helmut Veith. In Proc. ESOP, Springer, 2013]