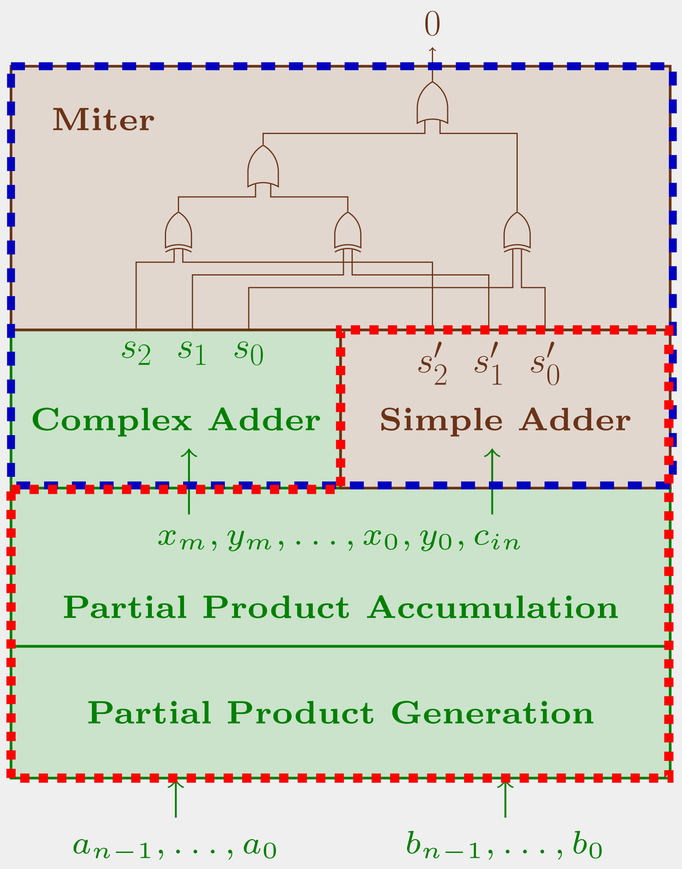
From DRUP to PAC and Back

Daniela Kaufmann, Armin Biere, Manuel Kauers

JOHANNES KEPLER UNIVERSITY LINZ

Certifying Arithmetic Circuits



SAT

Delete Reverse Unit Propagation

```
p cnf 3 5          -2 0            1  1 -2 -3 0 0
1 -2 -3  0       d  3 0            2  1  2  0 0
1  2     0       d  1 -2 -3 0      3 -1 -2  0 0
-1 -2    0       d -1 -2 0         4 -1  2  0 0
-1  2    0          0              5  3  0  0
       3  0                       6 -2  0 3 1 5 0
                                  7  0  4 2 6 0
```

Computer Algebra

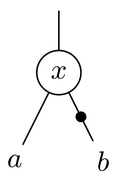
Practical Algebraic Calculus

```
<input>            <proof>
  -c*b*a+c*b;        * :         b*a,      c, c*b*a;
  b*a;               + : -c*b*a+c*b, c*b*a,    c*b;
```

Derive one single proof certificate

From DRUP to PAC...

1. Polynomial encoding



	Propostional Formula	Polynomial Relation
	$(x \leftrightarrow a \wedge \overline{b}) = \top$	$-x + a(1-b) = 0$
CNF	$(x \vee \overline{a} \vee b) = \top$	$(1-x)a(1-b) = 0$
	$(\overline{x} \vee a) = \top$	$x(1-a) = 0$
	$(\overline{x} \vee \overline{b}) = \top$	$xb = 0$

2. PAC encoding of resolution steps

```
<input>
  1 x*y;
  2 y*z-y-z+1;
  3 -fz+1-z;
```

```
<proof>
  4  *  3,  y-1, -fz*y+fz-y*z+y+z-1;
  5  +  2,    4, -fz*y+fz;
  6  *  1,   fz, fz*x*y;
  7  *  5,    x, -fz*x*y+fz*x;
  8  +  6,    7, fz*x;
  9  *  3,    x, -fz*x-x*z+x;
 10  +  8,    9, -x*z+x;
```

3. Combine proofs

`<input1> + <input2>`

`<proof1> + <proof2>`

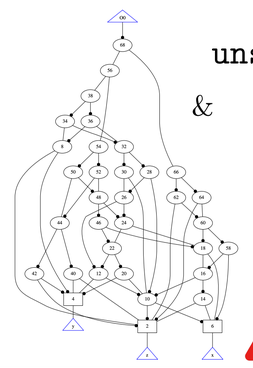
...and Back

1. SMT encoding

```
(set-logic QF_BV)
(declare-fun x () (_ BitVec 1))
(declare-fun y () (_ BitVec 1))
(declare-fun z () (_ BitVec 1))
(assert
(let (($v0 (bvadd (bvand #b011 ((_ sign_extend 2) x))
                  (bvand #b111 ((_ sign_extend 2) z)))))
(let (($w0 (bvadd (bvand #b010 ((_ sign_extend 2) y))
                  (bvand #b101 ((_ sign_extend 2) x)))))
(let (($p0 (bvadd (bvand #b010 ((_ sign_extend 2) y))
                  (bvand #b111 ((_ sign_extend 2) z)))))
(let (($e0 (= (bvadd $v0 $w0)  $p0)))

(not $e0))))))
(check-sat)
```

2. SMT solver



unsat

&

3. AIG to CNF

```
p cnf 34 78
-4  1   0
-4 -2   0
 4 -1 2 0
-5 -1   0
-5 -3   0
 5  1 3 0
```

4. Combine CNFs

Merging two unsatisfiable CNFs by combining target clauses.

5. SAT solver

```
d 27 26 25 0
d 19 18 17 0
d 28 -27 -19 0
-9 0
d 7 -3 -1 0
d -8 -7 0
0
```

Experimental Results

architecture	n	Separate Proofs: DRUP gen	DRUP check	DRUP size	PAC gen	PAC check	PAC size	total	Combined Proof: PAC gen	PAC check	PAC total	PAC size	DRUP aig	DRUP smt	DRUP cnf	DRUP check	DRUP total	DRUP size
btor	16	-	-	-	0	0	5 181	0	-	-	-	-	0	3	136	177	316	11 079 431
sp-ar-cl	16	0	0	1 299	0	0	7 962	0	2	2	3	185 588	0	7	300	264	570	19 317 884
sp-dt-lf	16	0	0	1 167	0	0	7 787	0	1	1	2	136 349	0	6	279	277	562	18 153 668
bp-ct-bk	16	0	0	1 029	0	0	7 205	0	1	1	2	128 720	0	7	TO	-	-	-
bp-wt-cl	16	0	0	2 902	0	0	7 946	0	30	11	41	614 742	0	7	TO	-	-	-
btor	32	-	-	-	0	0	21 629	0	-	-	-	-	0	32	2 887	TO	-	-
sp-ar-cl	32	0	0	14 927	0	1	33 834	1	133	31	164	1 597 897	0	56	TO	-	-	-
sp-dt-lf	32	0	0	3 138	0	1	33 451	1	2	3	5	321 720	0	52	TO	-	-	-
bp-ct-bk	32	0	0	2 276	0	1	27 312	1	1	2	3	217 128	0	49	TO	-	-	-
bp-wt-cl	32	1	1	46 502	0	1	30 561	2	3 133	242	3 375	5 536 176	0	55	TO	-	-	-

time in sec
TO = 3600 sec

Design, Automation and Test in Europe Conference (DATE) 2020