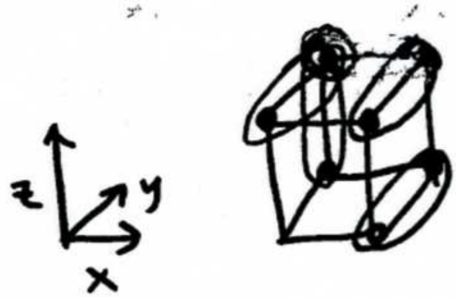
CSEE 6861
Prof. Nowick

PRIME GENERATION PROBLEM EXAMPLE: Generate all Primes

Hyperculee for a function f:



← Initial cover F (covers entire ON-set and DC-set)

(Note that some of the implicants in F are not even prime!)

$$F = x'z + x'y + x'yz + xz + xz'$$

$\boxed{\text{Step 1}}$ Termination rules B1-B4 + U1 do not apply. Go to step 2. (See PLA below.)

$\boxed{\text{Step 2.}}$ Pick splitting variable. A good heuristic is described in Handout #21 (TAUTOLOGY). Here, variable x is binate (appears as both x + x'). It also contributes to the most implicants. Therefore, pick x. (z could have been picked, but x is more "balanced," see Handout #21)

Recursively compute:

(i) $A1 = x \cdot \text{Primes}(F_x)$

(ii) $A0 = x' \cdot \text{Primes}(F_{x'})$

$\boxed{\text{First, compute } F_x + F_{x'}}$ (cofactors):

PLA: $F =$

	x	y	z	f
1	0	-	1	1
2	0	1	-	1
3	0	1	1	1
4	1	-	1	1
5	1	-	0	1

⇒ Compute F_x: ⇒ $\boxed{F_x =}$

	y	z	f
4	-	1	1
5	-	0	1

(cont.)

Step 2. (cont.)

Compute $F_{x'}$:

PLA:

$$F = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} x & y & z & f \\ \hline 0 & - & 1 & 1 \\ 0 & 1 & - & 1 \\ 0 & 1 & 1 & 1 \\ 1 & - & 1 & 1 \\ 1 & - & 0 & 1 \end{bmatrix} \Rightarrow \boxed{F_{x'}} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{bmatrix} y & z & f \\ \hline - & 1 & 1 \\ 1 & - & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Next, compute Primes(F_x) and Primes($F_{x'}$)

$\Rightarrow$ Both F_x & $F_{x'}$ are terminal cases.

F_x: $\boxed{\text{Rule B2}}$ - single input dependence.
F_x is a tautology. Return the universal cube:

$$\boxed{\text{Primes}(F_x) = \{1\}}$$

$F_{x'}$: $\boxed{\text{Rule U1}}$ - unate function.
$F_{x'}$ is unate. Do "SCC", then return the result:

$$\boxed{\text{Primes}(F_{x'}) = \text{SCC}(F_{x'}) = \{y, z\}}$$

where $\text{SCC}\{yz, y, z\} = \{y, z\}$

(cont.)

PRIME GENERATION PROBLEM EXAMPLE (cont.)

Step 2. (cont.)

Finally, compute A1 & A0:

where $\boxed{A1 = x \cdot Primes(F_x)}$

$$= x\{1\} = \boxed{\{x\}}$$

and $\boxed{A0 = x' \cdot Primes(F_{x'})}$

$$= x'\{y, z\} = \boxed{\{x'y, x'z\}}$$

Compute consensus (A1, A0): take pairwise consensus, of each cube in A1 with each cube in A0:

$$consensus(x, x'y) = y$$

$$consensus(x, x'z) = z$$

$$\Rightarrow \boxed{consensus(A1, A0) = \{y, z\}}$$

Take union of the 3 sets:

$$A1 \cup A0 \cup consensus(A1, A0) = \{x, x'y, x'z, y, z\}$$

Apply SCC operator:

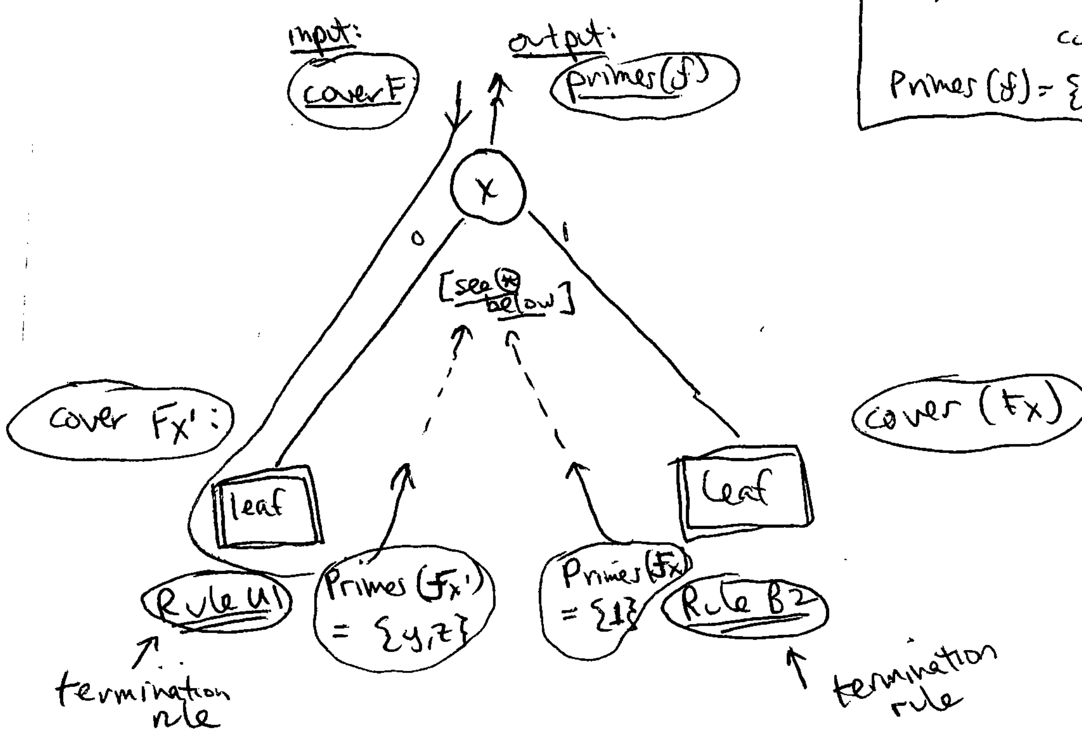
$$SCC(\{x, x'y, x'z, y, z\}) = \{x, y, z\}$$

$\Rightarrow$ this is final result!

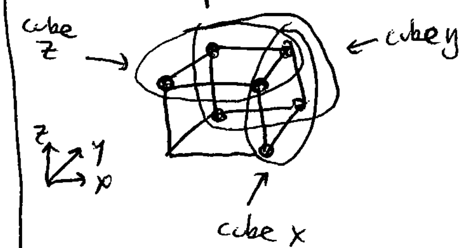
$\Rightarrow$ $\boxed{\text{Return Result: } \{x, y, z\}}$ = set of all primes of f.

Prime Generation Problem EXAMPLE (cont.)

Final Recursive Flow:



final cover of primes:



Primes (f) = {x, y, z}

(*) <u>At root node</u>: combining recursive results from 2 child nodes:

Recursive Prime Generation Theorem:

$$\boxed{Primes(f) = SCC(A1 \cup A0 \cup consensus(A1, A0))},$$

<u>where</u>:

$A1 = x \cdot Primes(F_x) = x \cdot \{1\} = \{x\}$, (1 = universal cube; x = cube "x")

$A0 = \bar{x} \cdot Primes(F_{x'}) = \bar{x} \cdot \{y, z\} = \{\bar{x}y, \bar{x}z\}$,

$consensus(A1, A0) = \{y, z\}$ (see page 3),

$A1 \cup A0 \cup consensus(A1, A0) = \{x, x'y, x'z, y, z\}$

and $SCC(A1 \cup A0 \cup consensus(A1, A0)) = \{x, y, z\}$
(where cubes x'y & x'z have been deleted)