Logic Optimization

Heuristic Based

Virendra Singh

Professor

Department of Electrical Engineering &

Dept. of Computer Science & Engineering

Indian Institute of Technology Bombay

http://www.ee.iitb.ac.in/~viren/

E-mail: viren@{ee, cse}.iitb.ac.in

EE-677: Foundations of VLSI CAD

Lecture 29 on 18 Oct 2021

CADSL

Heuristic logic minimization

- Provide irredundant covers with "reasonably small" sizes

minimal

- Fast and applicable to many functions
  - Much faster than exact minimization → QM +
- Avoid bottlenecks of exact minimization
  - Prime generation and storage
  - Covering
- Motivation
  - Use as internal engine within multi-level synthesis tools

Heuristic minimization -- principles

- Start from initial cover
  - Provided by designer or extracted from hardware language model
- Modify cover under consideration
  - Make it prime and irredundant
  - Perturb cover and re-iterate until a small irredundant cover is obtained
- Typically the size of the cover decreases
  - Operations on limited-size covers are fast

Heuristic minimization - operators

- Expand
  - Make implicants prime
  - Removed covered implicants
- Reduce
  - Reduce size of each implicant while preserving cover
- Reshape
  - Modify implicant pairs: enlarge one and reduce the other
- Irredundant
  - Make cover irredundant

Rough comparison of minimizers

- MINI
  - Iterate EXPAND, REDUCE, RESHAPE
- Espresso
  - Iterate EXPAND, IRREDUNDANT, REDUCE
- Espresso guarantees an irredundant cover
  - Because of the irredundant operator
- MINI may return irredundant covers, but can guarantee only minimality w.r.t. single implicant containment

$\rightarrow$ $ax\bar{b} \rightarrow abc$

$abc \rightarrow ax\bar{b}$

$ab'c$

ON, DC

Expand: Naïve implementation

- For each implicant
  - For each care literal
    - Raise it to don't care if possible
  - Remove all implicants covered by expanded implicant
- Issues
  - Validity check of expansion
  - Order of expansion

f

abc* → a**·*

abc*

ON-set

DC-set

OFF-set X

Tautology

$\overline{f}$

Containment

Complementation & intersections

abc → ab* → a** → ***

a*c → **c

Validity check

- Espresso, MINI
  - Check intersection of expanded implicant with OFF-set
  - Requires complementation
- Presto
  - Check inclusion of expanded implicant in the union of the ON-set and DC-set
  - Reducible to recursive tautology check

Ordering heuristics

- Expand the cubes that are unlikely to be covered by other cubes
- Selection:
  - Compute vector of column sums
  - *Weight*: inner product of cube and vector
  - Sort implicants in ascending order of weight
- Rationale:
  - Low weight correlates to having few 1s in densely populated columns

Example

- f = a'b'c' + ab'c' + a'bc' + a'b'c

  DC-set = abc'

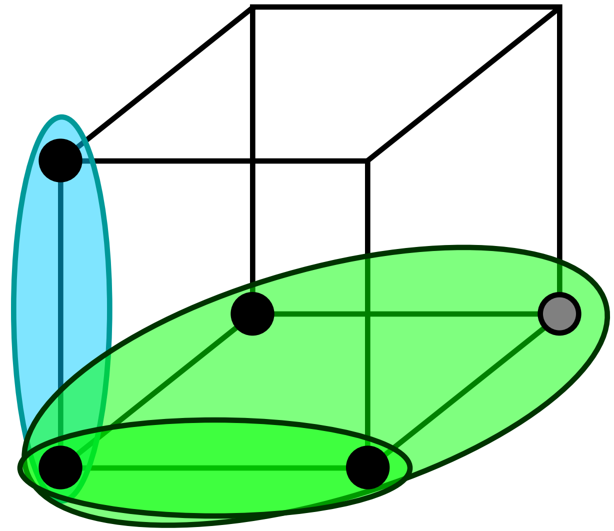
$$
\begin{array}{c|ccc}
 & a & b & c \\
\hline
\bar{a}\bar{b}\bar{c} & 10 & 10 & 10 \\
a\bar{b}\bar{c} & \boxed{01} & \boxed{10} & \boxed{10} \\
\bar{a}b\bar{c} & 10 & 01 & 10 \\
\bar{a}\bar{b}c & 10 & 10 & 01
\end{array}
\quad \text{ON set}
$$

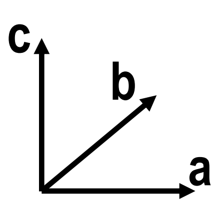
$3\times1 + 1\times0 + 3\times1 + 1\times0 + 3\times1 + 1\times0$

- Ordering:
  - Vector: $[3\ 1\ 3\ 1\ 3\ 1]^T$
  - Weights: (9, 7, 7, 7)
- Select second implicant.

Example (2)

α	10	10	10
β	01	10	10
γ	10	01	10
δ	10	10	01





Example (3)

$\bar{f}$

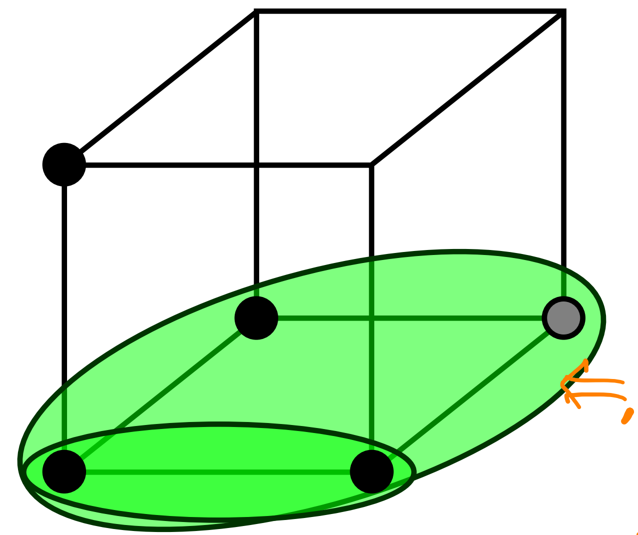
$\Rightarrow ac + bc$

- OFF-set: $\bar{f}$

01	11	01
11	01	01

$a\bar{b}\bar{c} \Rightarrow * \bar{b}\bar{c}$

- Expand 01 10 10:
  - 11 10 10 valid.
  - 11 11 10 valid.
  - 11 11 11 invalid.



- Update cover to:

11	11	10
10	10	01

01 11 01 ←

01 11 00 X | 01 10 00 X

11 01 00 X | 11 00 00 X

null

$\bar{a}\bar{b}c$

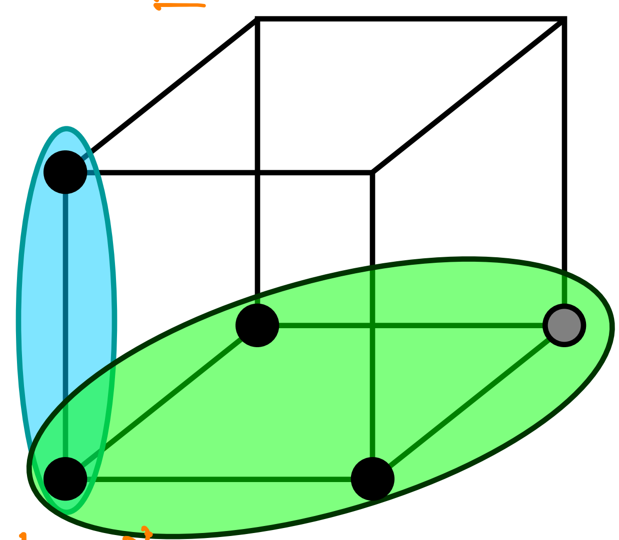
Example (4)

11	11	10
10	10	01

01 11 01
11 01 01 off set

$a\bar{b}c$

- Expand 10 10 01:
  - 11 10 01 invalid.
  - 10 11 01 invalid.
  - 10 10 11 valid.



00 10 01 valid
10 00 01 valid

01 10 01
00 11 01 x
10 01 01

- Expand cover:

11	11	10
10	10	11

$\bar{c}$

$\bar{a}\bar{b}$

$\bar{c} + \bar{a}\bar{b}$

Example (5)

Irredundant

α' 11 11 10
β' 10 10 11

β' is contained in f'

$f = \{\alpha', \beta'\}$

$f' = \{\cancel{\beta'}\}$

IRREDUNDANT

$f' =$ 11 11 10 ✓

to 11 11 10
10 10 11 ✓

10 10 10
01 01 00

$\Rightarrow$ 11 11 | 10

not tautology.

Expand → IRREDUNDAND

Postunlo

Expand heuristics in ESPRESSO

- Special heuristic to choose the order of literals
- Rationale:
  - Raise literals to that expanded implicant
    - Covers a maximal set of cubes
    - Overlaps with a maximal set of cubes
    - The implicant is as large as possible
- Intuitive argument
  - Pair implicant to be expanded with other implicants, to check the fruitful directions for expansion

Expand in Espresso

- Compare implicant with OFF-set.
  - Determine possible and impossible directions of expansion
- Detection of feasibly covered implicants
  - If there is an implicant β whose supercube with α is feasible, expand α to that supercube and remove β
- Raise those literals of α to overlap a maximum number of implicants
  - It is likely that the uncovered part of those implicant is covered by some other expanded cube
- Find the largest prime implicant
  - Formulate a covering problem and solve it heuristically

Reduce

$\bar{a}\bar{b}\cancel{*} \rightarrow \bar{a}\bar{b}\bar{c}$

reduction

- Sort implicants
  - Heuristics: sort by descending weight
  - Opposite to the heuristic sorting for expand
- Maximal reduction can be determine exactly
- Theorem:
  - Let α be in F and Q = F U D – { α }
    Then, the maximally reduced cube is:
    $\acute{\alpha} = \alpha \cap \text{supercube}(Q'_{\alpha})$

Example

Expansions

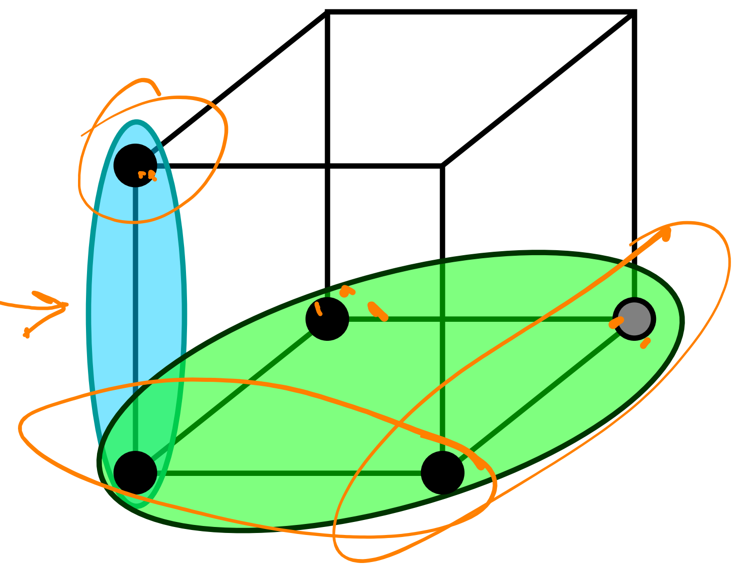
$b\bar{c}$

11 01 10

- Expand cover: $\bar{c}$

  11 11 10
  10 10 11

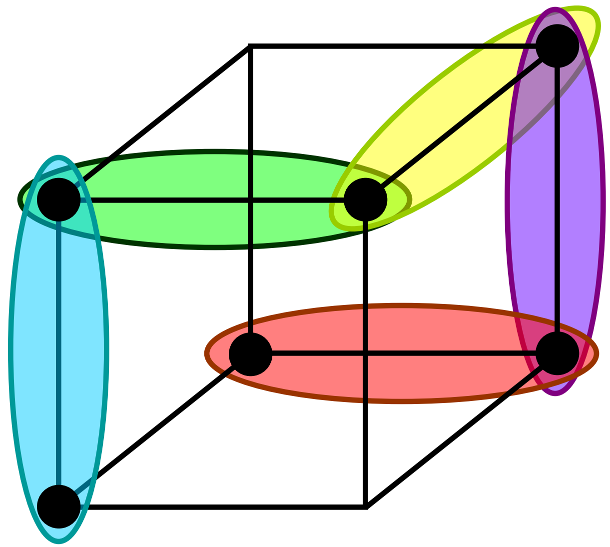
- Select first implicant:
  – Cannot be reduced.
- Select second implicant:
  – Reduced to 10 10 01
- Reduced cover:

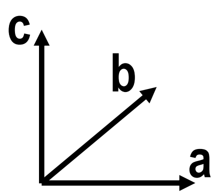


11 11 10
10 10 01

Irredundant cover

α	10	10	11
β	11	10	01
γ	01	11	01
δ	01	01	11
ε	11	01	10





Irredundant cover

- Relatively essential set E^r
  - Implicants covering some minterms of the function not covered by other implicants
  - Important remark: we do not know all the primes!
- Totally redundant set R^t
  - Implicants covered by the relatively essentials
- Partially redundant set R^p
  - Remaining implicants

Irredundant cover

- Find a subset of R^p that, together with E^r covers the function
- Modification of the tautology algorithm
  - Each cube in R^p is covered by other cubes
  - Find mutual covering relations
- Reduces to a covering problem
  - Apply a heuristic algorithm.
  - Note that even by applying an exact algorithm, a minimum solution may not be found, because we do not have all primes.

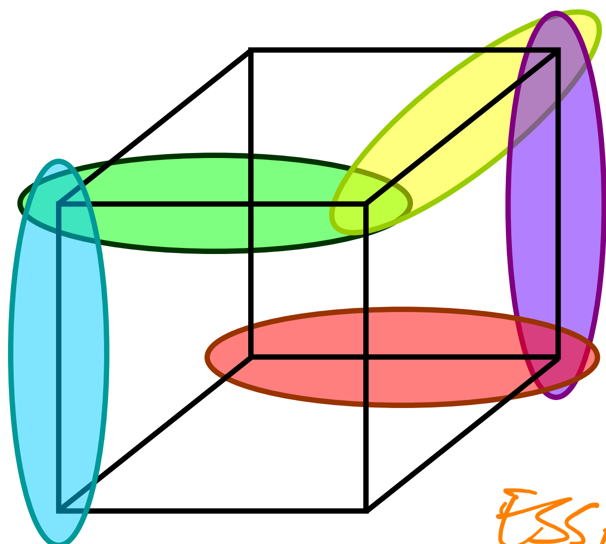
Example

α	10	10	11
β	11	10	01
γ	01	11	01
δ	01	01	11
ε	11	01	10

2 level Minimization

Development of 2-level optimizer using $\boxed{\text{Expand}}$ → Irredundant → Reduce.

- $E^r = \{\alpha, \varepsilon\}$
- $R^t = \varnothing$
- $R^p = \{\beta, \gamma, \delta\}$

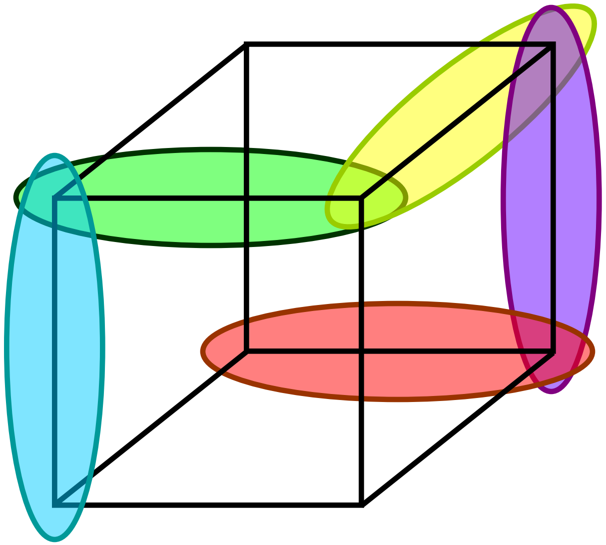


ESSPRESSO ✓

Group of 3.

Example (2)

- Covering relations:
  - β is covered by $\{\alpha, \gamma\}$.
  - γ is covered by $\{\beta, \delta\}$.
  - δ is covered by $\{\gamma, \varepsilon\}$.
- Minimum cover: $\gamma \cup E^r$



ESPRESSO algorithm in short

- Compute the complement
- Extract essentials
- Iterate
  - Expand, irredundant and reduce
- Cost functions:
  - Cover cardinality φ_1
  - Weighted sum of cube and literal count φ_2

ESPRESSO algorithm in detail

```
espresso(F,D) {
    R = complement(F ∪ D);
    F = expand(F,R);
    F = irredundant(F,D);
    E = essentials(F,D);
    F = F – E;  D = D ∪ E;
    repeat {
        φ2 = cost(F);
        repeat {
            φ1 = |F |;
            F = reduce(F,D);
            F = expand(F,R);
            F = irredundant(F,D);
        } until (|F | ≥ φ1);
        F = last_gasp(F,D,R);
    } until ( | F | ≥ φ1);
    F = F ∪ E;  D = D – E;
    F = make_sparse(F,D,R);}
```

Heuristic two-level minimization Summary

- Heuristic minimization is iterative
- Few operators are applied to covers
- Underlying mechanism
  - Cube operation
  - Unate recursive mechanism
- Efficient algorithms

Thank You