HEURISTIC TWO-LEVEL LOGIC OPTIMIZATION

©*Giovanni De Micheli*

Stanford University

Outline

- Heuristic logic minimization.
- Principles.
- Operators on logic covers.
- Espresso.

Heuristic minimization

- Provide irredundant covers
  with 'reasonably small' cardinality.
- Fast and applicable to many functions.
- Avoid bottlenecks of exact minimization:
  - Prime generation and storage.
  - Covering.

Heuristic minimization
Principles

- Local minimum cover:
  - Given initial cover.
  - Make it prime.
  - Make it irredundant.
- Iterative improvement:
  - Improve on cardinality by 'modifying'
    the implicants.

Heuristic minimization Operators

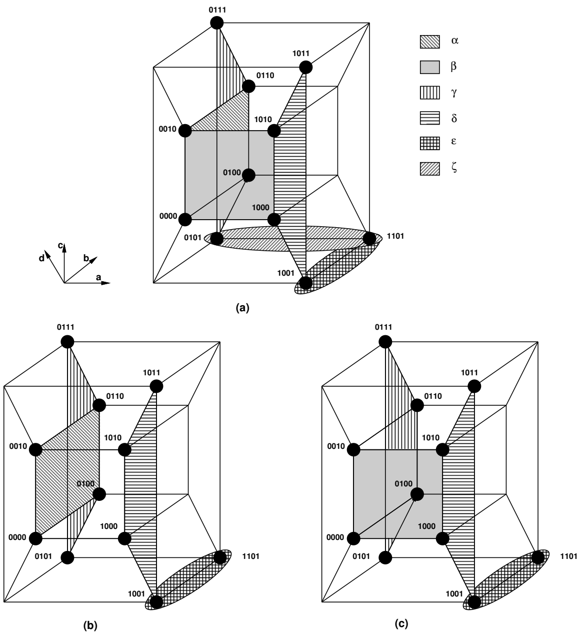
- **Expand**:
  - Make implicants prime.
  - Remove covered implicants.
- **Reduce**:
  - Reduce size of each implicant while preserving cover.
- **Reshape**:
  - Modify implicant pairs: enlarge one and reduce the other.
- **Irredundant**:
  - Make cover irredundant.

Example

0000	1
0010	1
0100	1
0110	1
1000	1
1010	1
0101	1
0111	1
1001	1
1011	1
1101	1

α	0**0	1
β	*0*0	1
γ	01**	1
δ	10**	1
ϵ	1*01	1
ζ	*101	1

Example

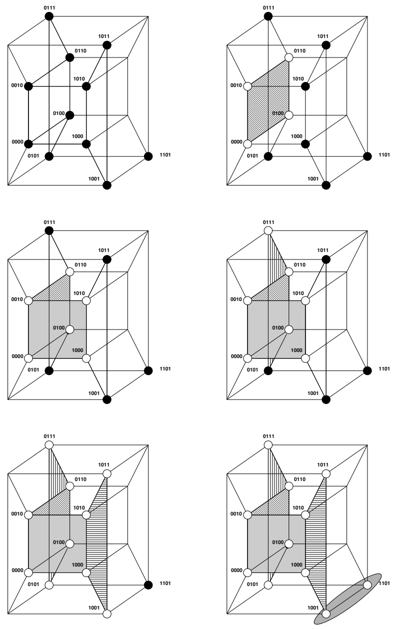


Example Expansion

- Expand 0000 to $\alpha = 0{*}{*}0$.
  - Drop 0100, 0010, 0110 from the cover.
- Expand 1000 to $\beta = {*}0{*}0$.
  - Drop 1010 from the cover.
- Expand 0101 to $\gamma = 01{*}{*}$.
  - Drop 0111 from the cover.
- Expand 1001 to $\delta = 10{*}{*}$.
  - Drop 1011 from the cover.
- Expand 1101 to $\epsilon = 1{*}01$.
- Cover is: $\{\alpha, \beta, \gamma, \delta, \epsilon\}$.

Example

© GDM



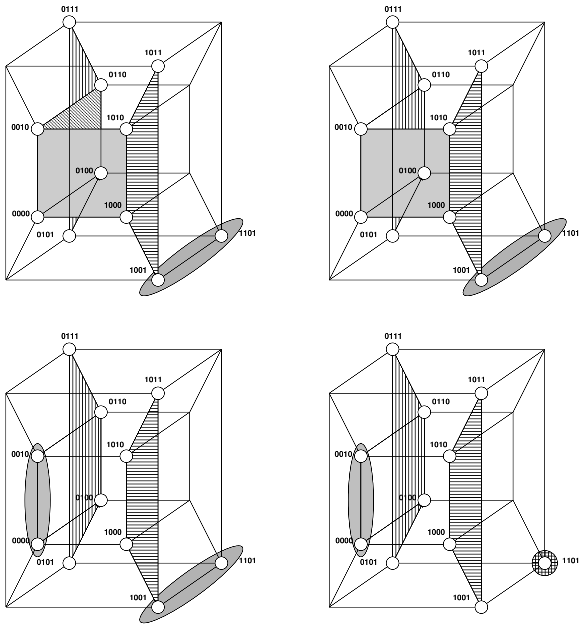
Example
Reduction

© GDM

- Reduce 0**0 to nothing.
- Reduce $\beta = *0*0$ to $\widetilde{\beta} = 00*0$
- Reduce $\epsilon = 1*01$ to $\widetilde{\epsilon} = 1101$
- Cover is: $\{\widetilde{\beta}, \gamma, \delta, \widetilde{\epsilon}\}$.

Example

© GDM

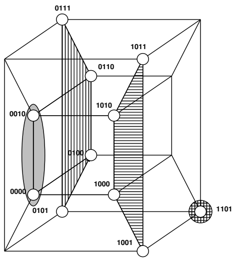


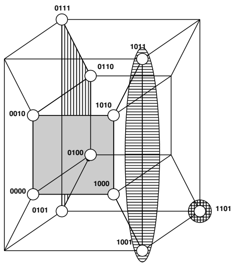
Example
Reshape

© GDM

- Reshape $\{\widetilde{\beta}, \delta\}$ to: $\{\beta, \widetilde{\delta}\}$
  - where $\widetilde{\delta} = 10*1$.
- Cover is: $\{\beta, \gamma, \widetilde{\delta}, \widetilde{\epsilon}\}$.

Example

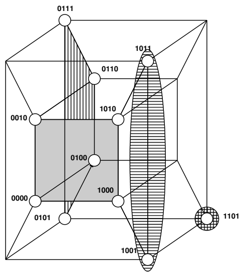


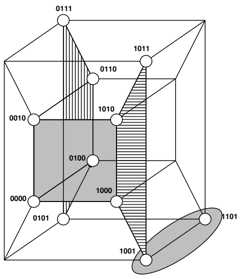


Example
Second expansion

- Expand $\widetilde{\delta} = 10*1$ to $\delta = 10**$.
- Expand $\widetilde{\epsilon} = 1101$ to $\epsilon = 1*01$.

Example





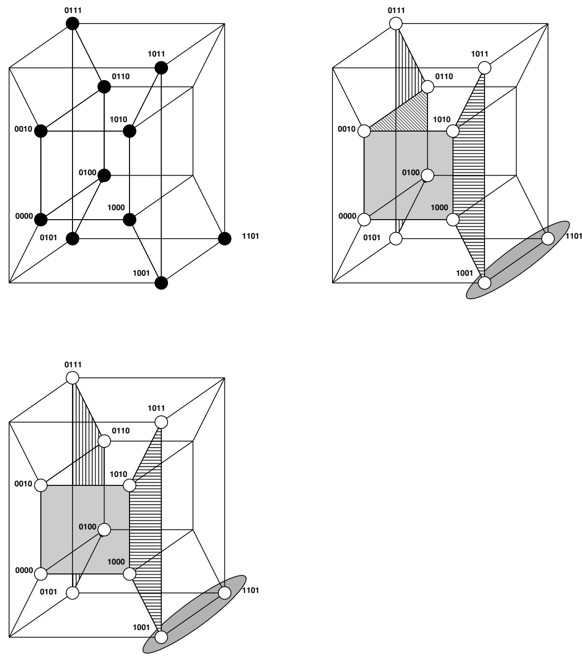
Example
(MINI summary)

- Expansion:
  - Cover: $\{\alpha, \beta, \gamma, \delta, \epsilon\}$.
  - Prime, redundant, minimal w.r. to scc.
- Reduction:
  - α eliminated.
  - $\beta = *0*0$ reduced to $\widetilde{\beta} = 00*0$.
  - $\epsilon = 1*01$ reduced to: $\widetilde{\epsilon} = 1101$.
  - Cover: $\{\widetilde{\beta}, \gamma, \delta, \widetilde{\epsilon}\}$.
- Reshape:
  - $\{\widetilde{\beta}, \delta\}$ reshaped to: $\{\beta, \widetilde{\delta}\}$ where $\widetilde{\delta} = 10*1$.
- Second expansion:
  - Cover: $\{\beta, \gamma, \delta, \epsilon\}$.
  - Prime, irredundant.

Alternative example (ESPRESSO)

- Expansion:
  - Cover: $\{\alpha, \beta, \gamma, \delta, \epsilon\}$.
  - Prime, redundant, minimal w.r. to scc.
- Irredundant:
  - Cover: $\{\beta, \gamma, \delta, \epsilon\}$.
  - Prime, irredundant.

Example



Expand naive implementation

- For each implicant
  - For each *care* literal
    - ∗ Raise it to *don't care* if possible.
  - Remove all covered implicants.
- Problems:
  - Validity check.
  - Order of expansions.

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.
  - Can be reduced to recursive tautology check.

Heuristics

© GDM

- Expand first 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

© GDM

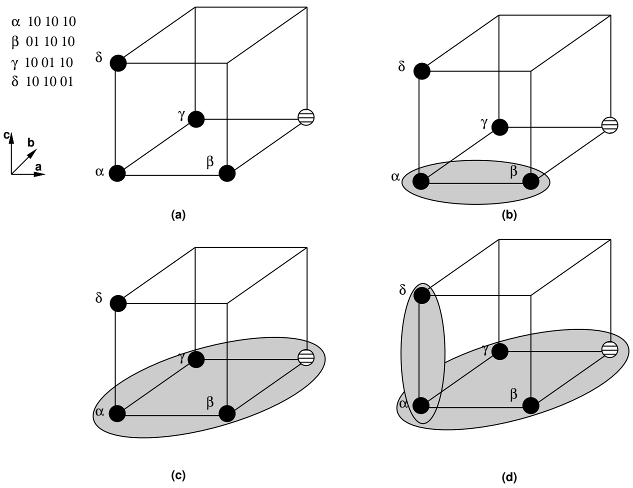
- $f = a'b'c' + ab'c' + a'bc' + a'b'c$
  DC-set $= abc'$

10	10	10
01	10	10
10	01	10
10	10	01

- Ordering:
  - Vector: $[313131]^T$
  - Weights: $(9, 7, 7, 7)$.
- Select second implicant.

Example (2)

© GDM



Example (3)

© GDM

- OFF-set:

01	11	01
11	01	01

- 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

Example (4)

11 11 10
10 10 01

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

11 11 10
10 10 11

Expand

- Smarter heuristics for choosing literals to be expanded.
- Four step procedure in Espresso.
- Rationale:
  - Raise literals so that expanded implicant:
    * Covers a maximal set of cubes.
    * Making it as large as possible.

Definitions

- $free$:
  - Set of entries that can be raised to 1.
- *Overexpanded cube*
  - Cube whose entries in $free$ are raised.
- *Feasible cover*
  - Expand a cube to cover another one while keeping it as an implicant of the function.

Expand in ESPRESSO

- *Determine the essential parts.*
  - Determine which entries can never be raised, and remove them from $free$.
  - Determine which parts can always be raised, raise them, and remove them from *free*.
- *Detection of feasibly covered cubes.*
  - If there is an implicant β whose supercube with α is feasible, repeat the following steps.
    * Raise the appropriate entry of α and remove it from *free*.
    * Remove from $free$ entries that can never be raised or that can always be raised and update α.
- *Expansion guided by the overexpanded cube.*
  - While the overexpanded cube of α covers some other cubes of F, repeat the following steps.
    * Raise a single entry of α as to overlap a maximum number of those cubes.
    * Remove from $free$ entries that can never be raised or that can always be raised and update α.
- *Find the largest prime implicant.*
  - Formulate a covering problem and solve it by a heuristic method.

Reduce

© GDM

- Sort implicants:
  - Heuristic: sort by descending weight.
- For each implicant:
- Lower as many * as possible to 1 or 0.
- *Theorem*:
  - Let $\alpha \in F$ and $Q = F \cup D - \{\alpha\}$.
    Then, the maximally reduced cube is:
    $\tilde{\alpha} = \alpha \cap supercube(Q'_\alpha)$.

Example

© GDM

- Expanded cover:

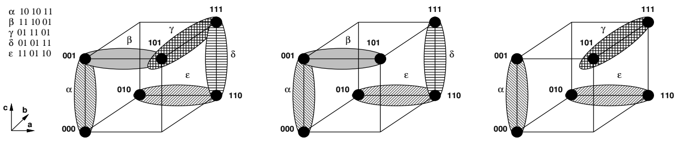
$$\begin{array}{ccc} 11 & 11 & 10 \\ 10 & 10 & 11 \end{array}$$

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

$$\begin{array}{ccc} 11 & 11 & 10 \\ 10 & 10 & 01 \end{array}$$

Irredundant cover

© GDM



Irredundant cover

© GDM

- *Relatively essential* set E^r
  - Implicants covering some minterms of the function not covered by other implicants.
- *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:
  - Heuristic algorithm.

Example

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

- $E^r = \{\alpha, \epsilon\}$
- $R^t = \emptyset$
- $R^p = \{\beta, \gamma, \delta\}$.

Example (2)

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

Espresso algorithm

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

Espresso algorithm

$espresso(F, D)\{$
  $R = complement(F \cup D);$
  $F = expand(F, R);$
  $F = irredundant(F, D);$
  $E = essentials(F, D);$
  $F = F - E;$
  $D = D \cup E;$
  repeat $\{$
    $\phi_2 = cost(F);$
    repeat $\{$
      $\phi_1 = |F|;$
      $F = reduce(F, D);$
      $F = expand(F, R);$
      $F = irredundant(F, D);$
    $\}$ **until** ($|F| \geq \phi_1$);
    $F = last\_gasp(F, D, R);$
  $\}$ **until** ($cost(F) \geq \phi_2$);
  $F = F \cup E;$
  $D = D - E;$
  $F = make\_sparse(F, D, R);$
$\}$

Summary heuristic minimization

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