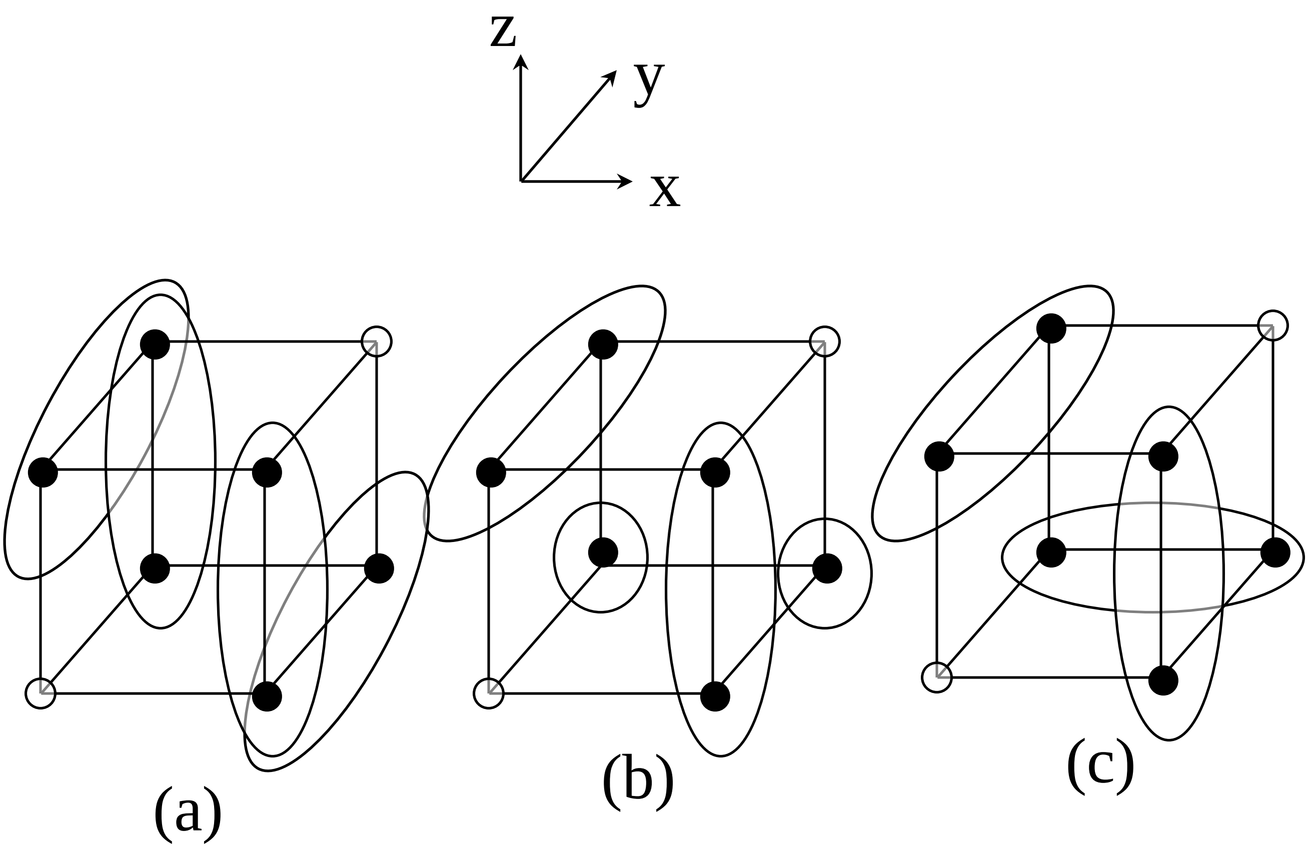
Simple Minimization Loop

```
F = EXPAND(F,D);
F = IRREDUNDANT(F,D);
do {
        cost = |F|;
        F = REDUCE(F,D);
        F = EXPAND(F,D);
        F = IRREDUNDANT(F,D);
} while (|F| < cost );
F = MAKE_SPARSE(F,D);
```

Heuristic Minimization of Two-level Logic



(a) Initial cover
(b) After Reduction
(c) After Expansion in the right direction
and Irredundant cover

Expand

- Expand
  - Carry out one cube at a time.
  - Expand cubes to prime and delete those cubes of F contained in the prime.
  - Order dependent
  - Goal:
    1. Cover as many cubes as possible
    2. Expand as large as possible

Expand

- The Blocking Matrix B(R,C)
  - block the expansion of a cube C not to intersect off-set
  - the cube to be expanded
  - the cover of off-set

  - $B_{ij} =$ 1 if ((C_j = 1) and $M(R)_{ij}$ = 0) or
    ((C_j = 0) and $M(R)_{ij}$ = 1)

- Ex:

C	0 0 2
offset M(R)	2 1 2
	2 1 1
B(R,C)	0 1 0
	0 1 0

Blocking Matrix

C	0 0 2
offset M(R)	2 1 2
	2 1 1
B(R,C)	0 1 0
	0 1 0

- What does a “1” mean in the blocking matrix?
  - The goal of expansion is to raise all the variables of a cube to 2. But expansion of the cube can’t cover the vertex in off-set So, some variable can’t be raised. A “1” in (i,j) means if variable j is not raised (lowing) , the expanded cube will not intersect the cube i of off-set.

Maximum Expansion

- L is a column covering of B
  <=> Every row of B contains a 1 in some
  column which appears in L.
- $C^{+}(L,C)_j = C_j \quad , j \subseteq \quad L$

  $2 \quad$, otherwise

- Proposition : If L is a minimum column covering of B, then $C^{+}(L,C)$ is a largest implicant of the function F.

Expand

- Covering matrix : determined by the cube and the given cover

$$C_{ij} = 1 \text{ if } ((C_j = 1) \text{and } M(F)_{ij} \neq 1) \text{ or } ((C_j = 0) \text{and } M(F)_{ij} \neq 0)$$

$\quad 0$ otherwise

Ex:

C	0 0 2
M(F)	1 0 1
	0 2 0
	1 0 0
C(F,C)	1 0 0
	0 1 0
	1 0 0

Covering Matrix

$$
\begin{array}{ll}
c & \underline{0\ 0\ 2} \\
 & \\
M(F) & 1\ 0\ 1 \\
 & 0\ 2\ 0 \\
 & \underline{1\ 0\ 0} \\
 & \\
C(F,c) & 1\ 0\ 0 \\
 & 0\ 1\ 0 \\
 & 1\ 0\ 0
\end{array}
$$

- What does a “1” mean in the covering matrix?
  => A “1” in (i,j) means if column j is in the minimal column covering, the cube i is not covered by the expanded cube.
- Select a minimal column covering so that many rows do not have a 1 in the column cover.

How to Find a Column Covering ?

Define two sets :

The lowing set : the variables remain the same.

The raising set : the variables raised to 2 (expanded)

Step1: Essential column(B) $\rightarrow$ lowing set

Ex:

$$
\begin{array}{rl}
 & 1\ 2\ 3\ 4 \\
\hline
C = & 0\ 1\ 0\ 1 \\
\hline
B(R,C) = & 1\ 0\ 1\ 0 \\
 & 1\ 0\ 0\ 1 \\
 & 0\ 1\ 0\ 1 \\
 & 0\ 1\ 0\ 0 \\
\hline
\end{array}
$$

lowing set = {2}

$$
\begin{array}{rll}
C(F,C) = & 1\ 1\ 1\ 1 & C_1 \\
 & 1\ 0\ 0\ 1 & C_2 \\
 & 0\ 0\ 1\ 0 & C_3 \\
 & 0\ 1\ 0\ 1 & C_4 \\
 & 0\ 1\ 1\ 1 & C_5
\end{array}
$$

How to Find a Column Covering ?

$$
\begin{array}{cc}
 & 1\ 3\ 4 \\
B(R,C) = & 1\ 1\ 0 \\
 & 1\ 0\ 1
\end{array}
$$

$$
\begin{array}{ccc}
C(F,C) = & 1\ 0\ 1 & C_2 \\
 & 0\ 1\ 0 & C_3
\end{array}
$$

Step 2 : Maximal feasible covering column set :

- Raising columns to cover as many cubes as possible
- To cover $\{C_3\}$, $\{3\}$ must be raised. If after$\{3\}$ being raised, the resultant matrix still has one 1's in each row, $\{3\}$ is a **feasible covering** column set.
- Maximal feasible covering column set is a feasible covering column set which covers a maximal number of cubes
- Select the MFC which commits the least number of columns

Ex : raising set = $\{3\} \cup \{4\}$

lowing set = $\{2\} \cup \{1\}$

$\Longrightarrow$ $C^+ = \{\ 0\ 1\ 2\ 2\}$

How to Find a Column Covering?

Step 3 : If no feasible covering column set (no cube can be covered), choose the column which has the most number of "1"s in C to raise.(Try to partially cover as many cubes as possible)