Projet de programmation en C
A. ED-DBALI

Objectif

Il s'agit de simplifier une expression booléenne donnée sous forme normale disjonctive canonique (abréviation $\mathcal{FNDC}$), avec la méthode de Quine-McCluskey (vue en M3I3 et entièrement reprise ici).

Description de la méthode

Expression booléenne sous $\mathcal{FNDC}$

Soit $\mathcal{V}$ un ensemble fini ou dénombrable de variables (dit ensemble des variables booléennes). On note les constantes booléennes `vrai` et `faux` par 1 et 0. Une expression booléenne est soit une variable, soit la constante 0 ou 1, soit la négation d'une autre expression, soit la conjonction ou la disjonction de deux expressions. Soient e_1 et e_2 deux expressions booléennes. Leur conjonction, dite produit booléen, est notée $e_1 \bullet e_2$ (ou plus simplement $e_1\, e_2$). Leur disjonction, dite somme booléenne, est notée $e_1 + e_2$. La négation d'une expression booléenne e est notée $\overline{e}$ pour négation booléenne.

L'opération somme booléenne est moins prioritaire que le produit. L'expression $x\ y + \overline{x}\ z$ s'interprète en $(x\ y) + (\overline{x}\ z)$.

On appelle *expression atomique* des variables $x_1, ..., x_n$ de $\mathcal{V}$, le produit $y_1\, y_2 \ldots y_n$ où pour chaque $i \in \{1, ..., n\}$, $y_i = x_i$ ou $y_i = \overline{x_i}$ (les y_i s'appellent des littéraux). Une expression sous $\mathcal{FNDC}$ est une somme d'expressions atomiques. Les expressions x, $\overline{x}\ y\ z + x\ \overline{y}\ \overline{z}$ sont des expressions sous $\mathcal{FNDC}$. Par contre les expressions $x + \overline{x}\ y$ et $(x + y)(\overline{x} + y)$ ne le sont pas.

Une expression booléenne faisant intervenir n variables est associée à une fonction booléenne de $\{0,1\}^n$ vers $\{0,1\}$. Ces fonctions sont en général une représentation mathématique de circuits électroniques à n entrées et une sortie (chaque connecteur logique ($+$, $\bullet$ ou $^{-}$) correspond à un composant électronique). Une expression sous $\mathcal{FNDC}$ est obtenue immédiatement à la lecture de la table de vérité de la fonction qu'elle représente. Une fonction peut se représenter par plusieurs expressions, mais celle utilisant le moins de connecteurs fournira le meilleur circuit (en taille, efficacité et coût). La simplification d'une expression consiste donc à trouver une expression équivalente ayant le minimum de connecteurs logiques. Il existe 2 méthodes "mécaniques" permettant de simplifier une expression sous $\mathcal{FNDC}$: la méthode de Karnaugh et celle de Quine-McCluskey.

La méthode de Karnaugh est graphique et donc visuelle. Elle ne marche raisonnablement que sur des expressions jusqu'à 4 variables. Au-delà, la méthode de Quine-McCluskey s'impose.

Simplification d'une expression : méthode de Quine-McCluskey

La méthode de Quine-McCluskey (1950) se décompose en deux étapes. La première détermine les produits qui peuvent intervenir dans l'expression simplifiée, La seconde ceux qui seront retenus dans l'expression finale.

Prenons un exemple :

Il s'agit de trouver une expression simplifiée de l'expression :

$$x\ y\ z + x\ \overline{y}\ z + \overline{x}\ y\ z + \overline{x}\ \overline{y}\ z + \overline{x}\ \overline{y}\ \overline{z}$$

• **Première étape** : On associe, dans une première phase, des chaînes de bits aux produits de cette expression (`0` pour un littéral négatif $\overline{x_i}$ et `1` pour un littéral positif x_i) :
$x\ y\ z \rightarrow$ `111`, $x\ \overline{y}\ z \rightarrow$ `101`, $\overline{x}\ y\ z \rightarrow$ `011`, $\overline{x}\ \overline{y}\ z \rightarrow$ `001` et $\overline{x}\ \overline{y}\ \overline{z} \rightarrow$ `000`.

Ensuite, on regroupe ces produits en fonction du nombre de 1 figurant dans la chaîne :

N^o du produit	Produit	Chaîne	Nombre de 1
1	$x\ y\ z$	`111`	3
2	$x\ \overline{y}\ z$	`101`	2
3	$\overline{x}\ y\ z$	`011`	2
4	$\overline{x}\ \overline{y}\ z$	`001`	1
5	$\overline{x}\ \overline{y}\ \overline{z}$	`000`	0

Table 1: Construction des chaînes

La seconde phase de cette étape consiste à combiner les produits qui ne diffèrent que d'un bit exactement. On obtient d'autres produits, plus simples puisqu'ils ont une variable de moins que les précédents.

Dans l'exemple, le produit 1 se combine avec le produit 2 pour donner le produit $x\ z$. La chaîne qui correspond à ce nouveau produit est notée `1-1` : le y a été remplacé par un tiret "`-`". La justification de cette transformation est que l'expression $x\,y\,z + x\,\overline{y}\,z$ est équivalente à $x\,z\,(y+\overline{y})$, elle-même équivalente à $x\,z\,1$ qui n'est autre que l'expression $x\,z$. Les autres combinaisons sont données dans le tableau suivant :

Combinaison	Produit résultat	Chaîne
(1, 2)	$x\ z$	`1-1`
(1, 3)	$y\ z$	`-11`
(2, 4)	$\overline{y}\ z$	`-01`
(3, 4)	$\overline{x}\ z$	`0-1`
(4, 5)	$\overline{x}\ \overline{y}$	`00-`

Table 2: Premières combinaisons

Dans ce nouveau tableau, on cherchera à combiner les produits à 2 variables pour les simplifier en un terme à une variable. Deux termes peuvent se combiner si leur chaîne ont un tiret au même endroit et si le reste ne diffère que d'un bit. Dans l'exemple, les termes (1, 2) et (3, 4) se combinent pour donner le terme z (la chaîne correspondante est `--1`). Une autre combinaison possible s'obtient avec les termes (1, 3) et (2, 4) pour donner également le terme z.

Ces 2 combinaisons qui aboutissent au même terme se résument par :

Combinaison	Produit résultat	Chaîne
(1, 2, 3, 4)	z	`--1`

Table 3: Deuxièmes combinaisons

car les combinaisons ((1, 2), (3, 4)) et ((1, 3), (2, 4)) se réecrivent simplement (1, 2, 3, 4).

La première étape s'arrête à ce stade puisqu'il n'y a plus moyen de combiner des termes.
Pour passer du tableau 1 au tableau 2, nous avons utilisé tous les produits. Donc aucun des produits d'origine ne peut figurer dans l'expression finale. Pour passer du tableau 2 au tableau 3, nous avons utilisé, pour aboutir au terme z, tous les produits sauf $\overline{x}\ \overline{y}$. Les termes candidats à figurer dans l'expression finale sont donc z et $\overline{x}\ \overline{y}$.

• **Deuxième étape** : Il s'agit ici de sélectionner parmi les termes trouvés à la première étape, ceux qui seront retenus dans l'expression finale. Pour cela, on forme un tableau qui a une ligne

par terme candidat et une colonne par terme d'origine. On met une croix à l'intersection d'une ligne et d'une colonne si le terme d'origine a été utilisé pour former le produit correpondant au terme de la ligne considérée. Naturellement, chaque produit d'origine doit être **couvert** par au moins un terme candidat.
L'exemple donne :

	1 $x\,y\,z$	2 $x\,\overline{y}\,z$	3 $\overline{x}\,y\,z$	4 $\overline{x}\,\overline{y}\,z$	5 $\overline{x}\,\overline{y}\,\overline{z}$
z	×	×	×	×	
$\overline{x}\,\overline{y}$				×	×

Table 4: Choix des termes de la première étape

car z est une combinaison des produits 1, 2, 3 et 4 et $\overline{x}\,\overline{y}$ est une combinaison des produits 4 et 5.

Le choix des termes définitifs se fait comme suit :

On cherche l'ensemble minimal (en cardinalité) des termes candidats qui couvriront tous les produits d'origine. Pour cela, s'il n'y a qu'une croix dans une colonne, le terme correspondant doit être retenu. Si ce terme couvre d'autres produits d'origine, on réduit le tableau en retirant le terme et les produits qu'il couvre. On réitère cette opération jusqu'à ce que tous les produits d'origine soient retirés du tableau.
Dans le cas où il n'y a plus de colonne à une seule croix et si plusieurs termes couvrent le même produit, on peut choisir n'importe lequel de ces termes. Nous sommes en présence d'un non déterminisme (c'est-à-dire que l'ensemble minimal n'est pas unique).

Cette étape est la moins évidente à programmer.

Dans l'exemple, la colonne 1 a une seule croix. Elle correspond au terme z. Donc ce terme figurera dans l'expression finale. Ce même terme couvre également les produits 2, 3 et 4. Le tableau auquel on retire le terme z et les produits qu'il couvre est :

	5 $\overline{x}\,\overline{y}\,\overline{z}$
$\overline{x}\,\overline{y}$	×

Ce nouveau tableau nous indique que seul le terme $\overline{x}\,\overline{y}$ couvre le produit 5. Donc le terme $\overline{x}\,\overline{y}$ est retenu. Dans cet exemple, tous les termes issus de la première étape ont été conservés. L'expression simplifiée équivalente à l'expression sous $\mathcal{FNDC}$ d'origine est donc : $z+\overline{x}\,\overline{y}$.

Autre exemple

Soit e l'expression sous $\mathcal{FNDC}$ suivante :

$$e = w\,x\,y\,\overline{z} + w\,\overline{x}\,y\,z + w\,\overline{x}\,y\,\overline{z} + \overline{w}\,x\,y\,z + \overline{w}\,x\,\overline{y}\,z + \overline{w}\,\overline{x}\,y\,z + \overline{w}\,\overline{x}\,\overline{y}\,z$$

• **Première étape** :
La correspondance *produits → chaînes de bits* donne le tableau suivant :

N^o du produit	Produit	Chaîne de bits	Nbr de 1
1	$w\,x\,y\,\overline{z}$	1110	3
2	$w\,\overline{x}\,y\,z$	1011	3
3	$\overline{w}\,x\,y\,z$	0111	3
4	$w\,\overline{x}\,y\,\overline{z}$	1010	2
5	$\overline{w}\,x\,\overline{y}\,z$	0101	2
6	$\overline{w}\,\overline{x}\,y\,z$	0011	2
7	$\overline{w}\,\overline{x}\,\overline{y}\,z$	0001	1

$\Downarrow$ (1^{eres} combinaisons)

Combinaison	Terme résultat	Chaîne
(1, 4)	$w\ y\ \overline{z}$	`1-10`
(2, 4)	$w\ \overline{x}\ y$	`101-`
(2, 6)	$\overline{x}\ y\ z$	`-011`
(3, 5)	$\overline{w}\ x\ z$	`01-1`
(3, 6)	$\overline{w}\ y\ z$	`0-11`
(5, 7)	$\overline{w}\ \overline{y}\ z$	`0-01`
(6, 7)	$\overline{w}\ \overline{x}\ z$	`00-1`

$\Downarrow$ (2^{emes} combinaisons)

Combinaison	Terme résultat	Chaîne
(3, 5, 6, 7)	$\overline{w}\ z$	`0--1`

- **Deuxième étape** :

Correspondances (termes de la première étape) $\rightarrow$ (produits d'origine) :

	1 $w\ x\ y\ \overline{z}$	2 $w\ \overline{x}\ y\ z$	3 $w\ \overline{x}\ y\ \overline{z}$	4 $\overline{w}\ x\ y\ z$	5 $\overline{w}\ x\ \overline{y}\ z$	6 $\overline{w}\ \overline{x}\ y\ z$	7 $\overline{w}\ \overline{x}\ \overline{y}\ z$
$\overline{w}\ z$			$\times$		$\times$	$\times$	$\times$
$w\ y\ \overline{z}$	$\times$			$\times$			
$w\ \overline{x}\ y$		$\times$		$\times$			
$\overline{x}\ y\ z$		$\times$				$\times$	

La colonne 3 n'a qu'une croix. Donc le terme $\overline{w}\ z$ doit être retenu. Ce terme couvre également les produits 5, 6 et 7.
La colonne 1 n'a qu'une croix. Donc le terme $w\ y\ \overline{z}$ doit être retenu. Ce terme couvre également le produit 4.
Après simplification du tableau (retrait des 2 termes sélectionnés avec les produits qu'ils couvrent) on obtient :

	2 $w\ \overline{x}\ y\ z$
$w\ \overline{x}\ y$	$\times$
$\overline{x}\ y\ z$	$\times$

Au vu de ce tableau, on a le choix entre les 2 termes $w\ \overline{x}\ y$ et $\overline{x}\ y\ z$ pour couvrir le produit 2.
L'expression finale s'écrit donc : $\overline{w}\ z + w\ y\ \overline{z} + w\ \overline{x}\ y$ ou bien $\overline{w}\ z + w\ y\ \overline{z} + \overline{x}\ y\ z$

Bon travail !