



**HAL**  
open science

# Implantation automatisée des circuits intégrés sur réseaux prédifusés CMOS

Mohamed Abdou Janati Idrissi

► **To cite this version:**

Mohamed Abdou Janati Idrissi. Implantation automatisée des circuits intégrés sur réseaux prédifusés CMOS. Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Grenoble - INPG, 1985. Français. NNT: . tel-00315968

**HAL Id: tel-00315968**

**<https://theses.hal.science/tel-00315968>**

Submitted on 2 Sep 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# THESE

*présentée à*

**l'Institut National Polytechnique de Grenoble**

*pour obtenir le grade de*  
**DOCTEUR DE 3<sup>ème</sup> CYCLE**  
**«Informatique»**

*par*

**Mohamed Abdou JANATI IDRISSI**

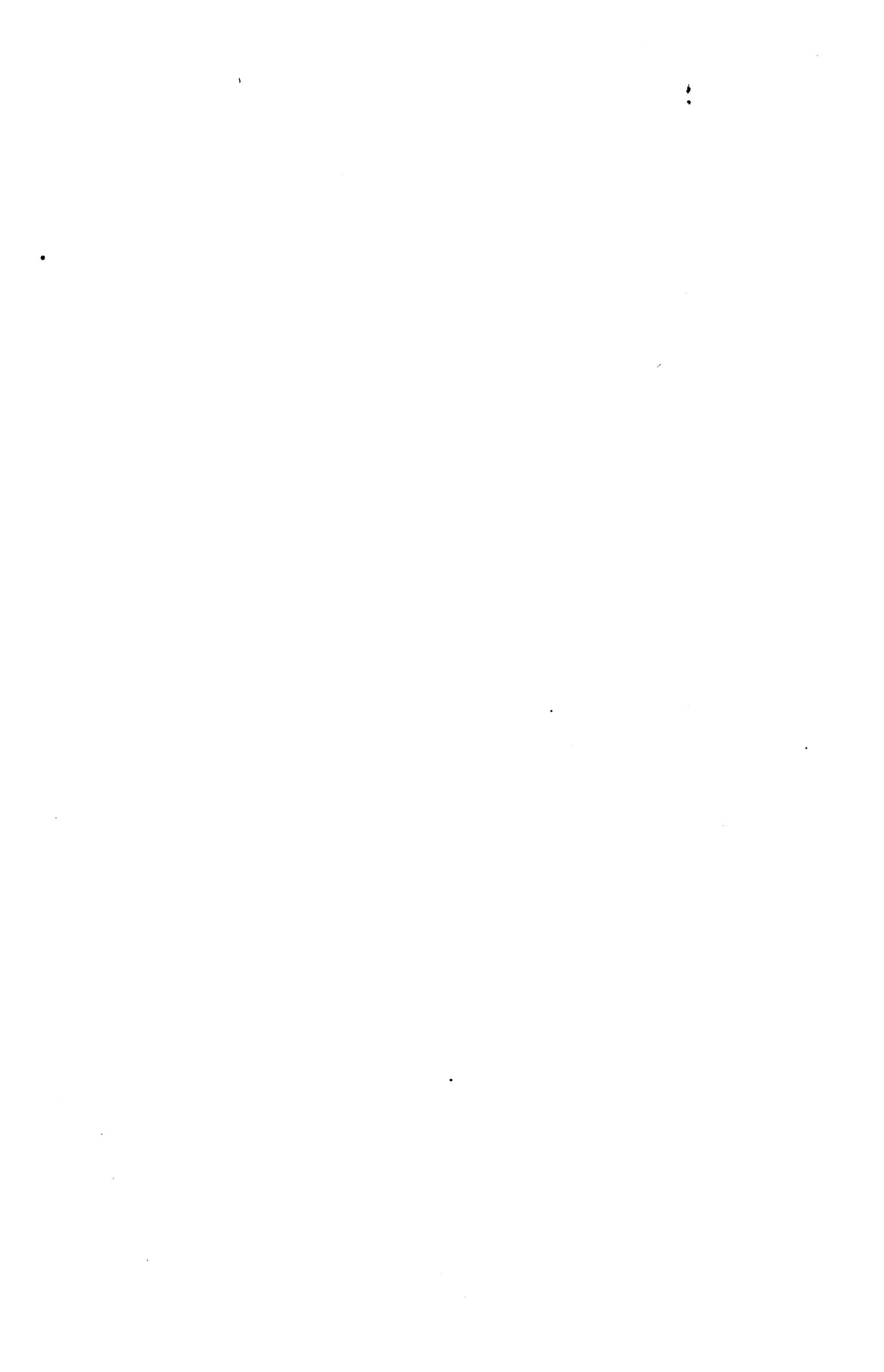


**IMPLANTATION AUTOMATISEE  
DES CIRCUITS INTEGRES SUR  
RESEAUX PREDIFFUSES CMOS.**



**Thèse soutenue le 1er juillet 1985 devant la commission d'examen.**

<b>J.R. BARRA</b>	<b>Président</b>
<b>M. BECKER</b>	
<b>M. FONTET</b>	<b>Examineurs</b>
<b>M. LAURENT</b>	
<b>G. SAUCIER</b>	



**INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE**

**Année universitaire 1982-1983**

**Président de l'Université : D. BLOCH**

**Vice-Président : René CARRE  
Hervé CHERADAME  
Marcel IVANES**

**PROFESSEURS DES UNIVERSITES :**

<b>ANCEAU François</b>	<b>E.N.S.I.M.A.G.</b>
<b>BARRAUD Alain</b>	<b>E.N.S.I.E.G.</b>
<b>BAUDELET Bernard</b>	<b>E.N.S.I.E.G.</b>
<b>BESSON Jean</b>	<b>E.N.S.E.E.G.</b>
<b>BLIMAN Samuel</b>	<b>E.N.S.E.R.G.</b>
<b>BLOCH Daniel</b>	<b>E.N.S.I.E.G.</b>
<b>BOIS Philippe</b>	<b>E.N.S.H.G.</b>
<b>BONNETAIN Lucien</b>	<b>E.N.S.E.E.G.</b>
<b>BONNIER Etienne</b>	<b>E.N.S.E.E.G.</b>
<b>BOUVARD Maurice</b>	<b>E.N.S.H.G.</b>
<b>BRISSONNEAU Pierre</b>	<b>E.N.S.I.E.G.</b>
<b>BUYLE BODIN Maurice</b>	<b>E.N.S.E.R.G.</b>
<b>CAVAIGNAC Jean-François</b>	<b>E.N.S.I.E.G.</b>
<b>CHARTIER Germain</b>	<b>E.N.S.I.E.G.</b>
<b>CHENEVIER Pierre</b>	<b>E.N.S.E.R.G.</b>
<b>CHERADAME Hervé</b>	<b>U.E.R.M.C.P.P.</b>
<b>CHERUY Arlette</b>	<b>E.N.S.I.E.G.</b>
<b>CHIAVERINA Jean</b>	<b>U.E.R.M.C.P.P.</b>
<b>COHEN Joseph</b>	<b>E.N.S.E.R.G.</b>
<b>COUMES André</b>	<b>E.N.S.E.R.G.</b>
<b>DURAND Francis</b>	<b>E.N.S.E.E.G.</b>
<b>DURAND Jean-Louis</b>	<b>E.N.S.I.E.G.</b>
<b>FELICI Noël</b>	<b>E.N.S.I.E.G.</b>
<b>FOULARD Claude</b>	<b>E.N.S.I.E.G.</b>
<b>GENTIL Pierre</b>	<b>E.N.S.E.R.G.</b>
<b>GUERIN Bernard</b>	<b>E.N.S.E.R.G.</b>
<b>GUYOT Pierre</b>	<b>E.N.S.E.E.G.</b>
<b>IVANES Marcel</b>	<b>E.N.S.I.E.G.</b>
<b>JAUSSAUD Pierre</b>	<b>E.N.S.I.E.G.</b>
<b>JOUBERT Jean-Claude</b>	<b>E.N.S.I.E.G.</b>
<b>JOURDAIN Geneviève</b>	<b>E.N.S.I.E.G.</b>
<b>LACOUME Jean-Louis</b>	<b>E.N.S.I.E.G.</b>
<b>LATOMBE Jean-Claude</b>	<b>E.N.S.I.M.A.G.</b>

.../...

LESSIEUR Marcel	E.N.S.H.G.
LESPINARD Georges	E.N.S.H.G.
LONGUEUE Jean-Pierre	E.N.S.I.E.G.
MAZARE Guy	E.N.S.I.M.A.G.
MOREAU René	E.N.S.H.G.
MORET Roger	E.N.S.I.E.G.
MOSSIERE Jacques	E.N.S.I.M.A.G.
PARIAUD Jean-Charles	E.N.S.E.E.G.
PAUTHENET René	E.N.S.I.E.G.
PERRET René	E.N.S.I.E.G.
PERRET Robert	E.N.S.I.E.G.
PIAU Jean-Michel	E.N.S.H.G.
POLOUJADOFF Michel	E.N.S.I.E.G.
POUPOT Christian	E.N.S.E.R.G.
RAMEAU Jean-Jacques	E.N.S.E.E.G.
RENAUD Maurice	U.E.R.M.C.P.P.
ROBERT André	U.E.R.M.C.P.P.
ROBERT François	E.N.S.I.M.A.G.
SABONNADIÈRE Jean-Claude	E.N.S.I.E.G.
SAUCIER Gabrielle	E.N.S.I.M.A.G.
SCHLENKER Claire	E.N.S.I.E.G.
SCHLENKER Michel	E.N.S.I.E.G.
SERMET Pierre	E.N.S.E.R.G.
SILVY Jacques	U.E.R.M.C.P.P.
SOHM Jean-Claude	E.N.S.E.E.G.
SOUQUET Jean-Louis	E.N.S.E.E.G.
VEILLON Gérard	E.N.S.I.M.A.G.
ZADWORNY François	E.N.S.E.R.G.

**PROFESSEURS ASSOCIES**

BASTIN Georges	E.N.S.H.G.
BERRIL John	E.N.S.H.G.
CARREAU Pierre	E.N.S.H.G.
GANDINI Alessandro	U.E.R.M.C.P.P.
HAYASHI Hirashi	E.N.S.I.E.G.

**PROFESSEURS UNIVERSITE DES SCIENCES SOCIALES (Grenoble II)**

BOLLIET Louis  
Chatelin Françoise

**PROFESSEURS E.N.S. Mines de Saint-Etienne**

RIEU Jean  
SOUSTELLE Michel

**CHERCHEURS DU C.N.R.S.**

FRUCHART Robert  
VACHAUD Georges

Directeur de Recherche  
Directeur de Recherche

.../...

ALLIBERT Michel	Maître de Recherche
ANSARA Ibrahim	Maître de Recherche
ARMAND Michel	Maître de Recherche
BINDER Gilbert	
CARRE René	Maître de Recherche
DAVID René	Maître de Recherche
DEPORTES Jacques	
DRIOLE Jean	Maître de Recherche
GIGNOUX Damien	
GIVORD Dominique	
GUELIN Pierre	
HOPFINGER Emil	Maître de Recherche
JOUD Jean-Charles	Maître de Recherche
KAMARINOS Georges	Maître de Recherche
KLEITZ Michel	Maître de Recherche
LANDAU Ioan-Dore	Maître de Recherche
LASJAUNIAS J.C.	
MERMET Jean	Maître de Recherche
MUNIER Jacques	Maître de Recherche
PIAU Monique	
PORTESEIL Jean-Louis	
THOLENCE Jean-Louis	
VERDILLON André	

**CHERCHEURS du MINISTERE de la RECHERCHE et de la TECHNOLOGIE (Directeurs et Maîtres de Recherches, ENS Mines de St. Etienne)**

LESBATS Pierre	Directeur de Recherche
BISCONDI Michel	Maître de Recherche
KOBYLANSKI André	Maître de Recherche
LE COZE Jean	Maître de Recherche
LALAUZE René	Maître de Recherche
LANCELOT Francis	Maître de Recherche
THEVENOT François	Maître de Recherche
TRAN MINH Canh	Maître de Recherche

**PERSONNALITES HABILITEES à DIRIGER des TRAVAUX de RECHERCHE (Décision du Conseil Scientifique)**

ALLIBERT Colette	E.N.S.E.E.G.
BERNARD Claude	E.N.S.E.E.G.
BONNET Rolland	E.N.S.E.E.G.
CAILLET Marcel	E.N.S.E.E.G.
CHATILLON Catherine	E.N.S.E.E.G.
CHATILLON Christian	E.N.S.E.E.G.
COULON Michel	E.N.S.E.E.G.
DIARD Jean-Paul	E.N.S.E.E.G.
EUSTAPOPOULOS Nicolas	E.N.S.E.E.G.
FOSTER Panayotis	E.N.S.E.E.G.

.../...

GALERIE Alain	E.N.S.E.E.G.
HAMMOU Abdelkader	E.N.S.E.E.G.
MALMEJAC Yves	E.N.S.E.E.G. (CENG)
MARTIN GARIN Régina	E.N.S.E.E.G.
NGUYEN TRUONG Bernadette	E.N.S.E.E.G.
RAVAINE Denis	E.N.S.E.E.G.
SAINFORT	E.N.S.E.E.G. (CENG)
SARRAZIN Pierre	E.N.S.E.E.G.
SIMON Jean-Paul	E.N.S.E.E.G.
TOUZAIN Philippe	E.N.S.E.E.G.
URBAIN Georges	E.N.S.E.E.G. (Laboratoire des ultra-réfractaires ODEILLON)
GUILHOT Bernard	E.N.S. Mines Saint Etienne
THOMAS Gérard	E.N.S. Mines Saint Etienne
DRIVER Julien	E.N.S. Mines Saint Etienne
BARIBAUD Michel	E.N.S.E.R.G.
BOREL Joseph	E.N.S.E.R.G.
CHOVET Alain	E.N.S.E.R.G.
CHEHIKIAN Alain	E.N.S.E.R.G.
DOLMAZON Jean-Marc	E.N.S.E.R.G.
HERAULT Jeanny	E.N.S.E.R.G.
MONLLOR Christian	E.N.S.E.R.G.
BORNARD Guy	E.N.S.I.E.G.
DESCHIZEAU Pierre	E.N.S.I.E.G.
GLANGEAUD François	E.N.S.I.E.G.
KOFMAN Walter	E.N.S.I.E.G.
LEJEUNE Gérard	E.N.S.I.E.G.
MAZUER Jean	E.N.S.I.E.G.
PERARD Jacques	E.N.S.I.E.G.
REINISCH Raymond	E.N.S.I.E.G.
ALEMANY Antoine	E.N.S.H.G.
BOIS Daniel	E.N.S.H.G.
DARVE Félix	E.N.S.H.G.
MICHEL Jean-Marie	E.N.S.H.G.
OBLÉD Charles	E.N.S.H.G.
ROWE Alain	E.N.S.H.G.
VAUCLIN Michel	E.N.S.H.G.
WACK Bernard	E.N.S.H.G.
BERT Didier	E.N.S.I.M.A.G.
CALMET Jacques	E.N.S.I.M.A.G.
COURTIN Jacques	E.N.S.I.M.A.G.
COURTOIS Bernard	E.N.S.I.M.A.G.
DELLA DORA Jean	E.N.S.I.M.A.G.
FONLUPT Jean	E.N.S.I.M.A.G.
SIFAKIS Joseph	E.N.S.I.M.A.G.
CHARUEL Robert	U.E.R.M.C.P.P.
CADET Jean	C.E.N.G.
COEURE Philippe	C.E.N.G. (LETI)

.../...

<b>DELHAYE Jean-Marc</b>	<b>C.E.N.G. (STT)</b>
<b>DUPUY Michel</b>	<b>C.E.N.G. (LETI)</b>
<b>JOUBE Hubert</b>	<b>C.E.N.G. (LETI)</b>
<b>NICOLAU Yvan</b>	<b>C.E.N.G. (LETI)</b>
<b>NIFENECKER Hervé</b>	<b>C.E.N.G.</b>
<b>PERROUD Paul</b>	<b>C.E.N.G.</b>
<b>PEUZIN Jean-Claude</b>	<b>C.E.N.G. (LETI)</b>
<b>TAIEB Maurice</b>	<b>C.E.N.G.</b>
<b>VINCENDON Marc</b>	<b>C.E.N.G.</b>

**LABORATOIRES EXTERIEURS**

<b>DEMOULIN Eric</b>	<b>C.N.E.T.</b>
<b>DEVINE</b>	<b>C.N.E.T. (R.A.B.)</b>
<b>GERBER Roland</b>	<b>C.N.E.T.</b>
<b>MERCKEL Gérard</b>	<b>C.N.E.T.</b>
<b>PAULEAU Yves</b>	<b>C.N.E.T.</b>
<b>GAUBERT C.</b>	<b>I.N.S.A. Lyon</b>



باسم الله الرحمن الرحيم

A mes parents,  
A mon frère Ali,  
A toute ma famille.



Je tiens à exprimer toute ma reconnaissance à Madame Gabriële SAUCIER, Professeur à l'ENSIMAG, d'avoir bien voulu m'accueillir dans son équipe de recherche, d'avoir su encadrer ce travail et de m'avoir fait l'honneur d'accepter d'être rapporteur de cette thèse.

Je tiens à remercier :

Monsieur Jean René BARRA, Professeur à l'USMG, de m'avoir fait l'honneur d'accepter de présider le jury de cette thèse et d'avoir bien voulu me faire profiter de ses suggestions constructives notamment dans le développement du chapitre III de ce mémoire,

Madame Monique BECKER, Chargé de recherches au CNRS, de s'être intéressée à ce travail et d'avoir accepté de faire partie du jury de cette thèse,

Monsieur Max FONTET, Professeur à l'Université de Paris VI, et Monsieur Marc LAURENT de Matra Harris, d'avoir accepté de faire partie du jury de cette thèse.

Je tiens également à remercier :

Monsieur Eric ROUSSAT de Cit-Alcatel, mes collègues Edmond KOUKA et Michel TRICOT pour leur collaboration,

Tous mes collègues du Laboratoire "Circuits et Systèmes" avec qui il me fût très agréable de travailler. Qu'il me soit permis de citer tout particulièrement Madame Catherine BELLON pour sa disponibilité, ses conseils et ses encouragements dans l'élaboration de ce mémoire, Monsieur Bernard MARTINET dont j'ai pu apprécier toutes les qualités au cours de notre collaboration.

Enfin, je tiens à adresser mes plus vifs remerciements à toutes les personnes qui ont assuré la réalisation technique de cet ouvrage :  
Madame Solange ROCHE pour la frappe ainsi que Monsieur Daniel IGLESIAS et l'équipe de reprographie de l'IMAG pour le tirage.

## TABLE DES MATIERES

### INTRODUCTION

### PARTIE I : METHODES ET ALGORITHMES POUR LE PLACEMENT AUTOMATIQUE DES CIRCUITS INTEGRES

#### I - IMPLANTATION DE CIRCUITS INTEGRES

##### I - 1. Conception d'un circuit intégré

##### I - 2. Implantation d'un circuit intégré

###### 1.2.1. Définitions

###### 1.2.2. Nécessité d'une automatisation de l'implantation

#### II - PLACEMENT

##### II - 1. Classification du problème de placement

##### II - 2. Coût d'un placement

###### 2.2.1. Définition du coût d'un placement

##### II - 3. Méthodes de placement

###### 2.3.1. Méthodes exactes

###### 2.3.2. Méthodes approchées

#### III - METHODES DE PLACEMENT INITIAL

##### III - 1. Méthodes stochastiques

##### III - 2. Méthodes constructives

###### 3.2.1. Méthodes ascendantes

###### 3.2.1.1. Généralités

###### 3.2.1.2. Exemples de méthodes ascendantes

###### 3.2.1.3. Inconvénients des méthodes présentées

###### 3.2.1.4. Propositions d'amélioration des méthodes constructives ascendantes

### 3.2.2. Méthodes descendantes

#### 3.2.2.1. Heuristique de "Kernighan et Linn"

#### 3.2.2.2. Deux catégories de méthodes descendantes

## IV - METHODES ITERATIVES D'AMELIORATION

### IV - 1. Méthodes stochastiques

#### 4.1.1. Amélioration du placement par la méthode des recuits successifs (Simulated annealing)

### IV - 2. Méthodes dirigées

#### 4.2.1. Echange de n éléments à la fois

#### 4.2.2. Echange par paires

## CONCLUSION

## PARTIE II : L'IMPLANTATION AUTOMATISEE DE CIRCUITS SUR RESEAUX PREDIFFUSES CMOS

### INTRODUCTION

#### I - PRESENTATION DU RESEAU CIT-ALCATEL

##### I - 1. Description de la structure prédiffusée

##### I - 2. Description de la bibliothèque de cellules

##### I - 3. Problèmes spécifiques

##### I - 4. Données du problème

#### II - PRESENTATION GENERALE DU SYSTEME IACMOS

##### II - 1. Description du système

##### II - 2. Domaine d'application du système IACMOS, description d'une structure proposée pour 2 couches en AL

### III - PLACEMENT

III - 1. Description générale de la méthode utilisée

III - 2. Placement initial constructif

3.2.1. Préstructuration logique

3.2.1.1. Discussion

3.2.2. Placement sur une structure hôte virtuelle

3.2.2.1. Procédure placer-élément-logique

3.2.2.2. Procédure placer-élément-périphérique

3.2.3. Placement final sur la structure hôte réelle

III - 3. Amélioration itérative

### CONCLUSION

### IV - LE TRACE DE CONNEXIONS

IV - 1. Introduction

IV - 2. Divers types de stratégies

IV - 3. Description de la méthode utilisée

## PARTIE III : VERS L'UTILISATION DES METHODES DE CLASSIFICATION

### I - LE PROBLEME A RESOUDRE

### II - RAPPEL SUR LES METHODES DE CLASSIFICATION AUTOMATIQUE

### III - STRATEGIE DE RESOLUTION

III - 1. Une méthode de classification automatique

III - 2. Qualité du résultat obtenu et proposition d'une amélioration  
complémentaire

### IV - EXPERIMENTATION

### V - CONCLUSION



## INTRODUCTION

L'accroissement bien connu de complexité des circuits intégrés rend de plus en plus cruciale la nécessité de définir une méthodologie de conception dont une étape est l'implantation.

Cette méthodologie doit être sûre, une des réponses à cette préoccupation est de traiter automatiquement le dessin du circuit. Cette automatisation du tracé des circuits permet de plus de dégager en partie les concepteurs d'une tâche pénible et source d'erreur.

Les progrès dans les techniques d'implantation automatique se font autour de deux axes :

- \* recherche de structures plus faciles à implanter, par exemple mise au point de nouveaux "styles" de dessin (PLA, MOS multi-drain, réseaux prédiffusés, précaractérisés, etc....)

- \* amélioration de méthodes d'implantation pour une structure donnée.

C'est dans cette deuxième tendance que se situe notre travail.

Dans la première partie nous étudierons différentes méthodes de placement en particulier les méthodes ascendantes. Nous montrerons leurs faiblesses et proposerons quelques améliorations originales.

Ces méthodes sont illustrées sur un exemple pratique dans la deuxième partie, elles ont donné lieu à un outil industriel de haut niveau.

La partie III concerne une partie originale de cette thèse à savoir  
l'utilisation des méthodes de classification dans le domaine de l'implantation  
automatisée.

Cette approche semble la plus porteuse d'avenir.

**PARTIE I**

**METHODES ET ALGORITHMES POUR LE PLACEMENT AUTOMATIQUE  
DES CIRCUITS INTEGRES**



## I - IMPLANTATION DE CIRCUITS INTEGRES

I - 1. Conception d'un circuit intégré

La complexité de la conception d'un circuit VLSI (Very Large Scale Integration) nécessite une démarche descendante. D'une manière très schématique, le processus de conception peut être présenté comme suit :

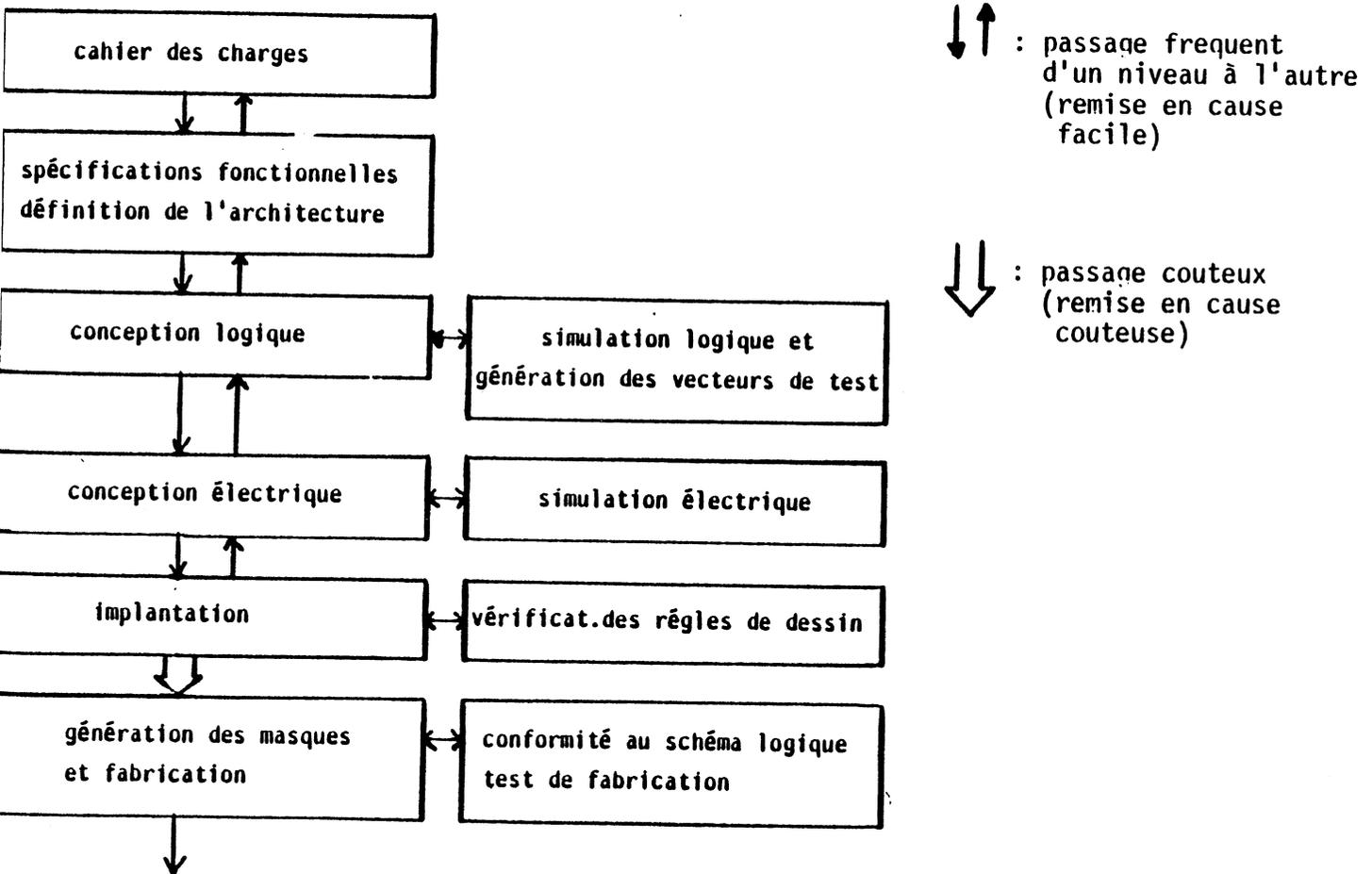


figure 1 : Etapes de la conception d'un circuit intégré

L'implantation qui est à la charnière entre la conception et la fabrication apparaît donc comme une des tâches les plus lourdes de la conception.

## I - 2. Implantation d'un circuit intégré

### 1.2.1. *Définitions*

L'implantation consiste à immerger un réseau logique sur une structure hôte : "La puce", en optimisant de nombreux critères et en respectant de nombreuses contraintes. Ces contraintes et ces critères dépendent de la technologie utilisée, et de la méthodologie de conception.

Le réseau logique est défini par un réseau R(E,C) [KUN 72]

E : ensemble des modules

C : ensemble des connexions reliant ces modules.

Lors de la phase d'implantation, les modules sont définis par leur forme extérieure ; il s'agit en général de rectangles sur les périmètres desquels sont situés les points de contact à connecter.

Les connexions représentent les équipotentiels électriques. Elles relient par leurs points de contact d'une part ces modules entre eux, d'autre part certains modules à des points particuliers : point d'entrée-sortie situés à la périphérie du circuit, point d'alimentation, point de masse.

La structure hôte est définie par un ensemble de places (chaque place pouvant recevoir un module). Les places peuvent être :

- non définies à priori et situées sur des parties quelconques de la puce (circuits à la demande),
- prédéfinies sur la puce (réseaux prédiffusés).

La réalisation des connexions dans le deuxième cas se fait essentiellement dans les zones réservées uniquement à cet effet : canaux d'interconnexion.

Ces canaux peuvent être :

- (i) non fixés à priori
- (ii) fixés dans deux directions
- (iii) fixés dans une seule direction

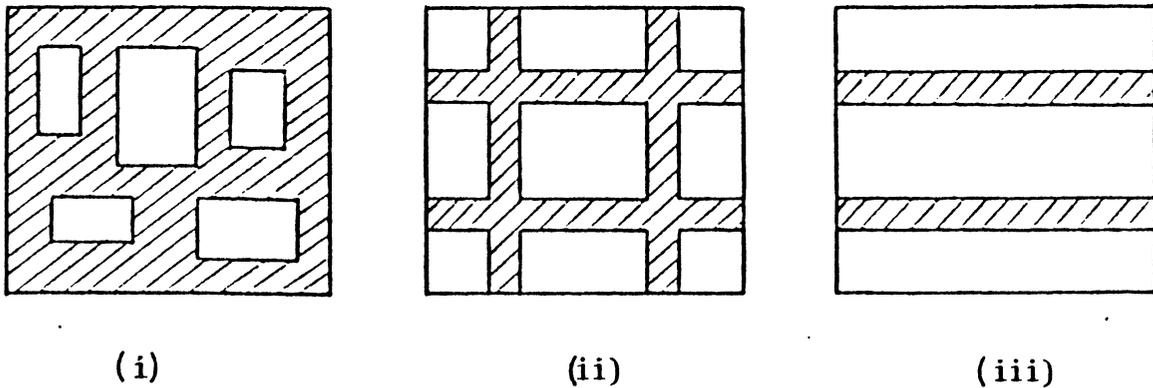


Figure 2 : Canaux d'interconnexion ( zones hachurées )

L'implantation consiste donc en une immersion du réseau initial dans la structure topologique de la puce. Elle est généralement décomposée en deux étapes : le placement des modules puis le tracé des connexions entre les modules ou routages. Bien que très dépendantes, ces deux étapes de l'implantation, à cause de leur complexité ne peuvent être abordées simultanément, elles sont donc résolues séquentiellement : placement, puis tracé sans pour autant interdire une remise en cause du placement durant l'étape du tracé.

### 1.2.2. Nécessité d'une automatisation de l'implantation

Nous allons exposer comparativement trois façons de procéder :

- placement et routage automatique
- placement manuel + automatique, routage automatique
- placement manuel complet, routage automatique.

Pour cela nous les comparerons sur une étude de cas consistant en un circuit dont les caractéristiques sont :

- 534 modules
- 515 équipotentielles
- 1776 accès de cellules à connecter.

Des programmes du commerce ont été utilisés pour les trois démarches.

a) Placement et routage automatique

Le placement a nécessité quatre à cinq heures CPU. Le routage a pris quatre heures et demi de temps CPU, et réalisé 78 % des interconnexions à connecter.

b) Placement automatique + manuel - Routage automatique

Un placement manuel des structures répétitives, des composants les plus importants, et des entrées-sorties, soit environ 66 à 70 % des éléments a été réalisé. Les 30 % restants ont été placés de façon aléatoire, puis réorganisés par amélioration itérative sans modifier les éléments implantés manuellement. L'amélioration a nécessité deux heures et demi de temps CPU, la préparation du placement une journée et l'édition du placement une demi journée. Le routage automatique a nécessité quatre heures et demi de temps CPU et a réalisé 93,5 % des interconnexions.

c) Placement manuel complet - Routage automatique

La préparation d'un tel placement a nécessité un à deux jours d'ingénieurs spécialistes en réseau prédiffusé, et une demi à une journée d'édition sur terminal graphique. Le routage a réalisé 99,5 % des connexions et a dépensé huit heures de temps CPU.

Nous constatons sur l'exemple de ce petit circuit :

- une différence de temps du traitement considérable entre les deux approches : manuelle et automatique
- l'approche utilisant un placement manuel rend aussi impossible une solution finale. Seulement 95 % des connexions sont traitées, soit 10 % de plus que dans l'approche automatique.

Ces deux derniers points, en plus de l'avènement des circuits à très haute intégration ont rendu obligatoires dans cette dernière décennie la prise en compte des préoccupations suivantes :

- l'automatisation de l'implantation,
- l'amélioration des algorithmes déjà utilisés,
- la recherche de solutions plus "structurées".

Un éventail considérable de publications concernant ces problèmes apparaît ainsi chaque année dans les actes des différentes conférences internationales (DAC, ICCD, ICCAD, VLSI, etc...) et traduit la collaboration sur ce domaine d'un bon nombre de chercheurs d'horizons multiples : informaticiens, électroniciens, mathématiciens et physiciens.

Nous allons illustrer dans la première partie de cette thèse les tendances les plus significatives dans le domaine du placement. Nous parlerons du routage au cours de la deuxième partie.

## II - PLACEMENT

Etant donnés  $n$  éléments avec leur graphe de connectivité, il s'agit de trouver un emplacement optimisé pour ces éléments sur une structure hôte définie soit par  $m$  positions ( $m \geq n$ ), soit par une surface  $S$  ( $S \geq S_1 \cup S_2 \cup \dots \cup S_n$  où  $S_1, S_2, \dots, S_n$  sont des surfaces individuelles de chaque élément).

Cette optimisation concerne le tracé des connexions ultérieur ; il s'agit de trouver un placement tel que ce tracé soit possible et soit de bonne qualité.

### II - 1. Classification du problème de placement

Certaines difficultés pour résoudre ce problème de placement sont dues aux raisons suivantes :

a) le nombre de placements possibles dépend du nombre  $n$  de modules à placer et des contraintes de la structure hôte. Il est dans la plupart des cas fort important ( $\approx n!$ ).

b) La taille des circuits actuels est considérable ; la difficulté de faire face aux contraintes topologiques et technologiques rendent difficile et coûteuse la recherche d'une solution réalisable.

c) La diversité des critères qui permettent de caractériser et d'évaluer un placement (longueur des connexions, minimisation et uniformisation de la densité, minimisation de la surface ...) est importante.

d) Le tracé n'étant pas connu durant l'étape du placement, l'évaluation des critères cités au point (c) ne peut être que prédictive. De plus, la taille importante des circuits ne permet qu'une évaluation simple à mettre en oeuvre et peu coûteuse en temps de calcul. Cependant une estimation plus fine du critère de faisabilité du tracé par exemple est nécessaire pour une structure à ressources d'interconnexion limitée.

e) L'existence d'antagonismes entre critères. La figure suivante est un exemple d'illustration appuyant cette affirmation.

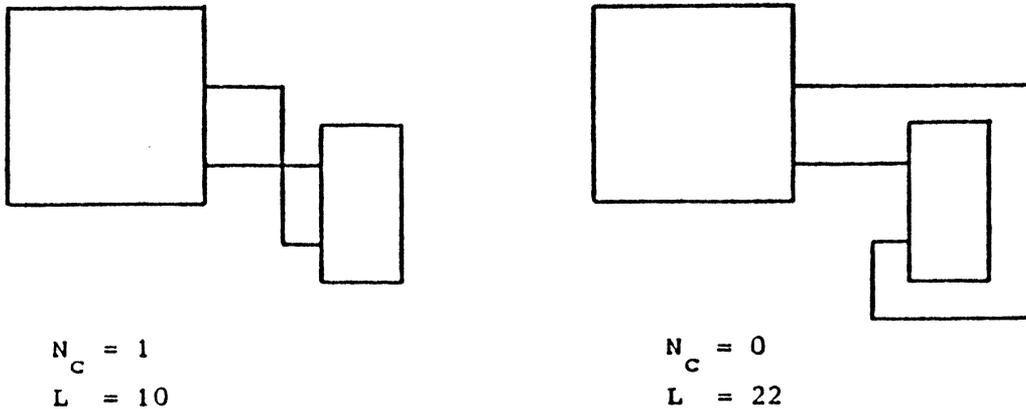


Figure 3 : nombre de croisements contre longueur de connexions

$N_c$  : Nombre de croisements des connexions

L : Longueur des connexions

f) La présence de contraintes sévères contribue fortement à dégrader la qualité des solutions comme l'illustrent les 2 figures suivantes [MAS 85]

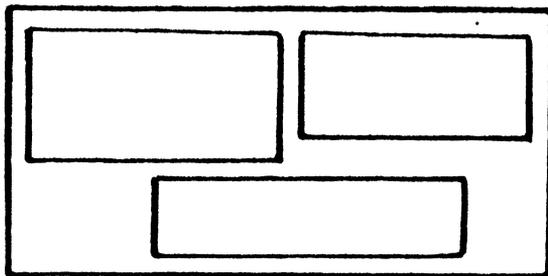


Figure 4 : forme géométrique rigide → mauvaise occupation de l'espace

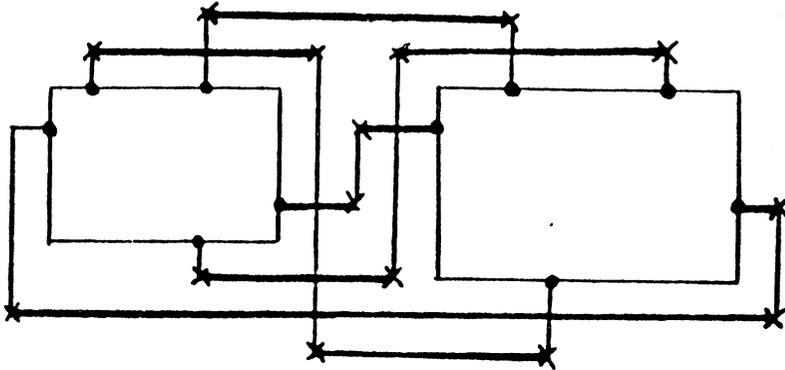


figure 5 : position fixée des E/S → accroissement surface de cablage

g) la difficulté de définir une fonction objective, permettant d'affirmer qu'un "placement est meilleur" qu'un autre, tout en tenant compte des exigences précédentes.

En fait, le problème du placement, même dans sa formulation la plus simplifiée demeure très complexe. En effet, la recherche d'un placement d'un ensemble de modules dont la longueur totale des connexions (distance Euclidienne) est inférieure à une valeur fixée  $L$  est un problème NP complet [DO 80].

## II - 2. Coût d'un placement

Parmi les critères usuellement retenus pour apprécier la qualité d'un placement on peut citer :

- la longueur des connexions,
- surface et forme du circuit,
- nombre de croisements de connexions,
- la densité des connexions.

A chaque placement  $P$ , on associe une valeur  $C_i(P)$  de  $\mathbf{R}$  calculée de manière prédictive (le tracé des connexions n'étant pas connu durant l'étape de placement). Cette valeur  $C_i(P)$  appelée coût élémentaire permet de mesurer la grandeur que le critère  $i$  optimise. Une évaluation prédictive des coûts associés à ces différents critères est donnée en [SER 82, BEL 84, BRE 77,...].

### 2.2.1. *Définition du coût d'un placement*

La qualité d'un placement  $P$  est évaluée par son coût  $C(P)$ . Le coût est défini à partir :

- des  $n$  coûts élémentaires  $\{C_1(P), C_2(P), \dots, C_n(P)\}$  associés à  $n$  critères jugés importants par l'implanteur. Cette importance est guidée par : la nature du circuit à implanter (dense ou peu dense ...), les contraintes technologiques et topologiques de la structure hôte, la méthode du placement utilisée.

- de la relation qui lie ces  $n$  coûts élémentaires et qui doit refléter le compromis ou l'ordre de priorité entre les  $n$  critères associés. Cette relation prend dans la pratique une des deux formes suivantes :

a) une forme vectorielle

$$C(P) = \begin{pmatrix} C_1(P) \\ \vdots \\ C_n(P) \end{pmatrix} \quad \text{défini sur } \mathbb{R}^n \text{ tel que } C_i \text{ caractérise le critère } i \text{ et}$$

$(\forall (i,j) \in [1,n]^2, i < j)$  alors (le critère  $i$  est plus important que le critère

b) une forme polynomiale

$$C(P) = \sum_{i=1}^n \lambda_i (C_i(P))^{p_i}$$

où le couple  $(\lambda_i, p_i)$  traduit l'importance du critère  $i$ .

### Définitions

Soient  $P$  et  $P'$  deux placements pour lesquels les coûts sont respectivement  $C(P)$  et  $C(P')$ .

$P$  est meilleur que  $P'$   $\iff C(P) < C(P')$  (cas de minimisation)

$C(P) < C(P')$  se traduit par :

$$\text{cas b) : } \sum_{i=1}^n \lambda_i (C_i(P))^{p_i} < \sum_{i=1}^n \lambda_i (C_i(P'))^{p_i}$$

cas a) : par la relation d'ordre partielle suivante :

$$C(P) \neq C(P')$$

et

$$\exists j \in [1,n] \text{ tq } C_i(P) = C_i(P') \text{ pour tout } i < j$$

$$C_j <_{\mathbb{R}} C'_j$$

Le résultat d'une méthode de placement dépend impérativement du choix du coût d'un placement. Il est souvent difficile d'évaluer ce coût : difficulté pour classer les critères importants par ordre de priorité décroissante pour une forme vectorielle, difficulté pour trouver le couple de paramètres  $(\lambda_i, p_i)$  associé à chaque critère  $i$  pour une forme polynomiale. Ces difficultés apparaissent encore plus quand le nombre de critères augmente. De plus, la valeur de certains critères dépend spécifiquement du circuit à implanter. Elle peut aussi dépendre de la phase du placement dans laquelle on se trouve. Le concepteur peut alors dans ce cas-là modifier automatiquement les objectifs du placement, après le traitement d'un certain pourcentage des éléments du circuit :

- soit en modifiant l'ordre de priorité des critères (dans la forme vectorielle)
- soit en modifiant les pondérations  $(\lambda_i, p_i)$  des critères (dans la forme polynomiale).

Nous pensons donc qu'une définition intelligente et réfléchie du coût d'un placement est une condition impérative d'un placement satisfaisant d'un circuit.

### Coût partiel

On appellera coût partiel par rapport à un sous-réseau  $R'$  de  $R(E,C)$  ( $R(E,C)$  étant le réseau à implanter), l'évaluation du coût sur  $R'$ .  $R'$  est défini par un sous-ensemble de  $E$  et les connexions les reliant.

### Coût réduit

On appellera coût réduit  $C_r(P)$  d'un placement  $P$ , une évaluation du coût de ce placement à partir d'un sous-ensemble de l'ensemble des  $n$  critères définissant  $C(P)$ .

## II - 3. Méthodes de placement

La quasi totalité des circuits à implanter sont de taille suffisamment grande pour qu'il ne soit pas permis d'appliquer des méthodes exactes de résolution. Ces méthodes conduisent à une solution optimale au prix d'un temps de calcul prohibitif. Parmi ces méthodes on peut citer :

### *2.3.1. Méthodes exactes*

Des méthodes énumératives qui cherchent à énumérer les  $n!$  permutations des  $n$  modules.

Des méthodes semi-énumératives qui explorent partiellement l'arbre des solutions possibles. Ces méthodes sont basées sur la méthode "branch and bound" définie par Little [LI 63]. Divers exemples de ces méthodes ainsi que leurs applications sont clairement exposés dans [HAN 72] et [REI 77]. Ces méthodes sont rarement employées et restent inapplicables pour un nombre élevé de modules ( $n \geq 15$ ), mais pourraient être amenées à se développer dans le cadre de traitement par des processeurs spécialisés parallèles.

Des méthodes approchées dites des heuristiques seront appliquées dans le but de donner la meilleure solution possible dans un temps de calcul relativement court. L'évolution des ordinateurs permettra l'application d'heuristiques de plus en plus performantes. Dans le paragraphe qui suit nous allons parler d'une catégorie d'heuristiques très utilisées pour la résolution du problème du placement automatique.

### 2.3.2. Méthodes approchées

Le principe de toutes les méthodes de type est le même :

- obtenir dans un premier temps, un placement initial en affectant à chaque module du réseau logique une place dans la structure hôte :

méthode de placement initial,

- ensuite on tente d'améliorer ce placement initial en changeant la position d'un nombre réduit de modules.

Nous allons nous intéresser à ces deux points dans le cadre de deux chapitres différents, en raison du large éventail d'études qu'ils ont engendré dans la littérature technique.

### III - METHODES DE PLACEMENT INITIAL

#### III - 1. Méthode stochastique [HAN 72]

Plusieurs placements, dont on évalue les coûts, sont générés suivant la méthode de Monte-Carlo. la meilleure solution est retenue lorsque le calcul s'arrête :

- soit après un temps fixé à l'avance, c'est-à-dire après un nombre fixé de solutions générées,
  - soit quand la meilleure solution est satisfaisante, c'est-à-dire que son coût est inférieur ou égal à une valeur fixée à l'avance.
- De telles approches ne sont pas réalistes car le concepteur dispose souvent durant cette étape du placement initial de bonnes informations à propos des éléments adjacents convenables.

#### III - 2. Méthodes constructives

##### 3.2.1. *Méthodes ascendantes*

##### 3.2.1.1. Généralités

Pour ce type de méthodes, la construction du placement initial se fait par pas, à partir d'éléments initiaux placés manuellement ou par programme.

A chaque itération le programme effectue les trois actions suivantes :

1. choix du module à placer
2. trouver une place pour ce module
3. placer ce module à cette place.

Quand un module a été affecté, sa place est fixée et ne peut plus être modifiée. Ces méthodes sont celles décrites par Breuer dans [HAN 72].

Les règles de sélection et de positionnement permettent de définir différentes méthodes.

(i) Règles de sélection des modules

La sélection se base essentiellement sur des critères de connectivité entre l'élément considéré à un certaine étape et les modules placés. Des mesures de connectivité entre modules distincts, des fonctions de sélection basées sur ces mesures ont été définies et discutées dans [HAN 72]. D'autres critères peuvent être pris en compte lors de la définition d'une règle de sélection : la taille du module, le type du module, la forme du module. En général, une règle de sélection en fournit pas une seule solution. Dans ce cas d'égalité de la fonction de sélection sur plusieurs éléments, une règle de discrimination est utilisée pour affiner cette sélection. On peut :

- soit choisir arbitrairement,
- soit optimiser une fonction de sélection secondaire,
- soit poursuivre la comparaison par rapport ux éléments non placés connectés à chacun de ces éléments candidats.

On peut également définir des classes d'éléments candidats : une classe d'éléments représente l'ensemble des modules pour lesquels la valeur de la fonction de sélection est la même. A chaque étape l'ensemble des éléments candidats est l'ensemble des éléments non placés de la classe en cours de traitement. Lorsque tous les éléments d'une classe sont placés, on passe à la classe suivante.

(ii) Règles de positionnement

A chaque étape on définit l'ensemble des places auxquelles peut-être affecté l'élément considéré à cette étape. En général c'est l'ensemble des places contiguës aux places du circuit occupées par les modules déjà placés (minimisation de la longueur). Ensuite on discrimine cet ensemble selon des

critères tenant compte de la forme du circuit et de la saturation des ressources d'interconnexion. Parmi les places restantes après discrimination on garde celle qui minimise la longueur. Cette approche est utilisée dans la deuxième partie.

### 3.2.1.2. Exemples de méthodes ascendantes

Les méthodes suivantes sont détaillées et analysées par Hanan et Kurtzberg dans [HAN 72].

#### a) Construction par paires (pair-linking method)

A chaque étape la méthode sélectionne l'élément le plus connecté à l'un des éléments déjà positionnés, puis le place le plus près possible de cet élément de manière à minimiser la longueur des connexions. Dans le cas où plusieurs places libres sont équidistantes de ce module déjà placé. On calcule les coordonnées  $(\bar{u}, \bar{v})$  du barycentre des coordonnées de tous les modules déjà placés et qui doivent être liés au nouvel élément. La place candidate la plus proche de ce barycentre est alors adoptée. Le placement est initialisé en choisissant comme noyau :

- soit l'ensemble des modules à place imposée,
- soit la paire des éléments ayant le plus grand nombre de connexions communes.

Cette méthode est utilisée principalement sur les réseaux structurés en bandes. Sa complexité est de l'ordre de  $n^2$  [KUR 65] où  $n$  est le nombre de modules à placer

b) Méthode par agrégation (cluster-development methode)

A chaque étape du placement on sélectionne l'élément le plus connecté à l'ensemble des éléments déjà positionnés. Chacun des éléments positionnés est affecté d'un poids, fonction du nombre de connexions avec l'élément sélectionné et leur barycentre est calculé. L'élément sélectionné est alors placé à ce barycentre, ou lorsque cette position est impossible, le plus près de ce barycentre.

La complexité de cette méthode est aussi de l'ordre de  $n^2$ .

c) Méthode par relaxation

Pour ces méthodes, une notion de zone de relaxation est introduite pour déterminer l'ensemble des places candidates pour un module lors d'une étape donnée. La zone de relaxation d'un élément candidat représente une région dans laquelle la résultante des forces d'attraction, exercées par les éléments déjà placés sur l'élément candidat, s'annule. Les forces d'attraction sont constituées par les connexions de l'élément candidat déjà présentes sur le réseau par au moins un point de contact placé.

Les règles de calcul de cette zone de relaxation permettent de définir différentes méthodes. Des méthodes de ce type ont été exposées dans [SOU 81, COF 79, HAN 72]. Ces heuristiques sont particulièrement adaptées aux problèmes de placement pour lesquels la position d'une partie ou de la totalité des cellules périphériques est prédéterminée.

### 3.2.1.3. Inconvénients des méthodes présentées

Il est vrai que ces méthodes sont particulièrement rapides et permettent quelquefois d'obtenir des solutions acceptables lorsque l'objectif cherché est la minimisation de la longueur des connexions. Cependant elles demeurent inefficaces en présence de grands circuits ou de circuits denses à cause des limitations suivantes :

#### 1 - Limitations d'ordre logique

Ces méthodes n'appréhendent pas globalement le problème.

- Le choix de l'élément à placer lors d'une étape donnée se fait par rapport au sous-ensemble des éléments déjà placés et non relativement à la totalité des modules du circuit.

- L'affectation de ce module à une place de la structure hôte se fait progressivement en cherchant à optimiser un coût partiel qui est la restriction de la fonction coût au sous-réseau constitué par les éléments déjà placés et de leurs connexions. Elle ne tient donc pas compte des modules des étapes suivantes. L'optimisation de ce coût partiel n'implique pas forcément une optimisation du coût global, elle risque par contre de conduire à une mauvaise solution comme le montre l'exemple suivant tiré de [PAL 84].

Supposons qu'on veuille placer et connecter trois modules A, B, et C sur une rangée. A est connecté une seule fois à B, B et C sont liés par deux connexions. Toutes les solutions à ce problème sont illustrées sur la figure. Le coût d'une solution est défini comme la longueur horizontale des connexions. Le coût de chaque solution est représenté entre parenthèses.

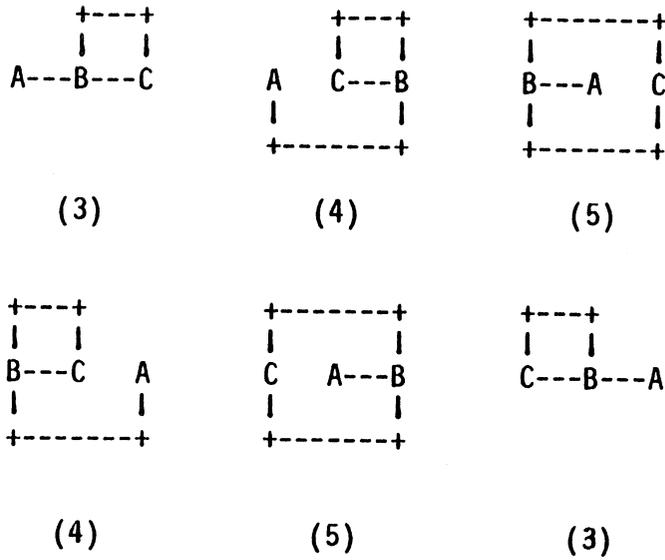


figure 6

On peut illustrer une séquence de résolution de cet exemple suivant la figure représentée ci-dessous :

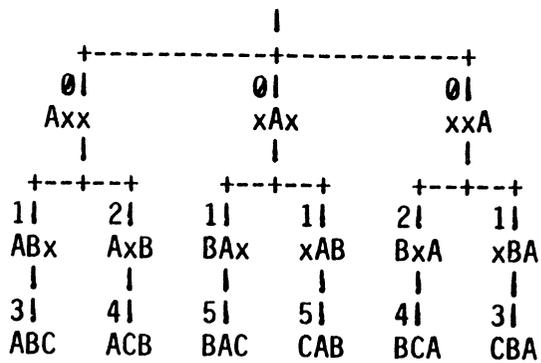


figure 7

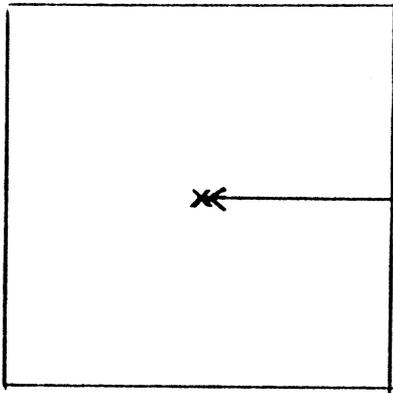
Au premier niveau on choisit A, A est ensuite affecté potentiellement à chacune des trois places de la rangée, le coût de chacune de ces affectations est supposé égal à zéro. A l'étape suivante, l'élément B est choisi car il est plus connecté à A. Les règles de positionnement ainsi que les coûts partiels sont de nouveau pris en considération, quatre noeuds se retrouvent avec le même coût minimum (1), deux de ces noeuds conduisent à la meilleure solution (3), mais les deux autres conduisent à la plus mauvaise (5).

## 2 - Limitation topologique

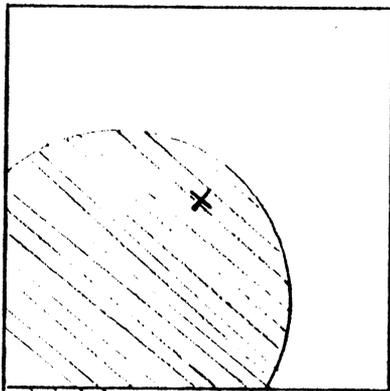
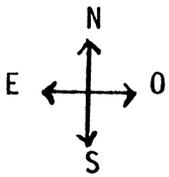
- Le choix de la position initiale est déterminant. Un écart vis à vis de la position optimale de l'élément initial dans la structure hôte est susceptible de dévier le placement du résultat optimal escompté.

- Les contraintes topologiques de la structure hôte peuvent être trop restrictives. En effet, l'évolution du placement vers un résultat optimum peut être entravée par des contraintes topologiques locales qui poussent à rechercher des places admissibles dans le respect de la structure hôte alors qu'une place candidate optimale pourrait se situer en dehors de cette structure imposée.

Nous illustrons ceci par la figure 8. La structure hôte est représentée par un carré.

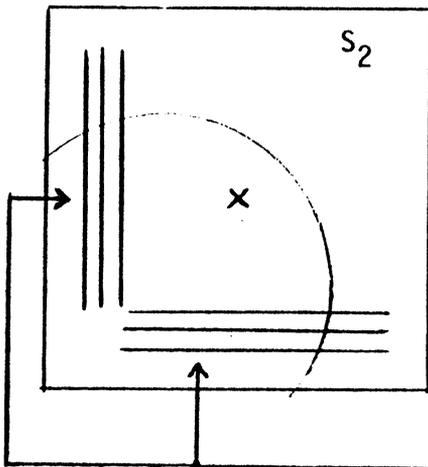


élément initial choisi comme élément central et affecté au centre de la structure hôte



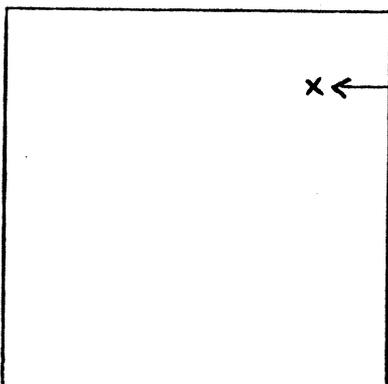
debut du placement...

blocaje aux frontières (E,S) de la structure hôte



on place sur la surface restante libre ( $S_2$ )

connexions (trop longues!)



nouvelle position initiale

En fait, il est possible que l'algorithme, ait abouti à ce resultat; si on avait affecté l'élément initial à cette position.

figure 8 : Limitation tonologique

3 - Un autre inconvénient de ces méthodes réside dans le fait qu'elles ne cherchent à minimiser que la longueur des connexions. Breuer dans [BRE 77] a discuté le besoin d'abandonner cette fonction objective classique basée sur la longueur des connexions et a introduit de nouvelles fonctions objectives basées sur la densité des connexions.

L'augmentation des densités d'intégration des circuits (jusqu'à 80 % de la surface d'un circuit peut-être réservée à l'interconnexion [SOU 81]) rend primordiale la préparation du tracé des connexions lors du placement afin de garantir le tracé de toutes les connexions et de tendre vers l'optimisation des critères propres au tracé (minimisation et uniformisation des densités estimées des connexions dans les canaux pour les structures à canaux d'interconnexion par exemple).

Il s'avère donc nécessaire de rechercher d'autres approches et moyens susceptibles de contourner ces difficultés ; ceci sera abordé au chapitre suivant sous forme de propositions.

#### 3.2.1.4. Propositions d'amélioration des méthodes constructives ascendantes

Celles-ci sont basées sur deux idées principales :

1 - La première s'appuie sur une vision globale du problème. On peut y distinguer deux apports originaux :

- a) les approches structurées tiendront compte de la structure logique dans sa globalité,
- b) le placement sur une structure hôte virtuelle permettra de guider l'optimisation locale vers une optimisation globale.

2 - La deuxième idée concerne la préparation du tracé lors de la phase du placement.

### 3.2.1.4.1. Pour mieux appréhender le problème

#### 3.2.1.4.1.1. Les approches structurées

Nous nous intéressons ici à une approche structurée du problème de placement initial en considérant que la structure du réseau logique à implanter doit guider la stratégie d'implantation. Un ordonnancement total ou partiel des éléments logiques ou une structuration en classes sont recherchés manuellement ou automatiquement avant leur affectation à des places du réseau d'accueil.

##### a) Ordonnancement manuel ou interactif

Le concepteur détermine le prochain élément à placer et le système C.A.O. recherche la meilleure place pour cet élément. Il est clair que s'il vient de placer une porte d'une bascule ; il cherchera à placer la porte "suivante" de la même bascule. L'ordre de ces éléments est une conséquence directe de la structure architecturale du réseau logique et peut-être obtenu à partir d'une description de type CADOC [CRA 85]. Le concepteur ou un système de niveau supérieur ayant désigné une ressource fonctionnelle à placer (registre par exemple), les constituants de cette ressource (bascules puis portes à l'intérieur de cette bascule) sont proposés pour le placement selon une approche hiérarchisée.

Les figures suivantes illustrent l'implantation d'un multiplieur de deux nombres de 4 bits sur un réseau prédiffusé CMOS de la société CIT-ALCATEL. Dans cette approche [CHA 85], on a essayé de conserver au maximum la structure logique du circuit du multiplieur. Pour ceci, le plan de masse a été reproduit le plus fidèlement possible tel qu'il avait été fait pour un dessin au micron.

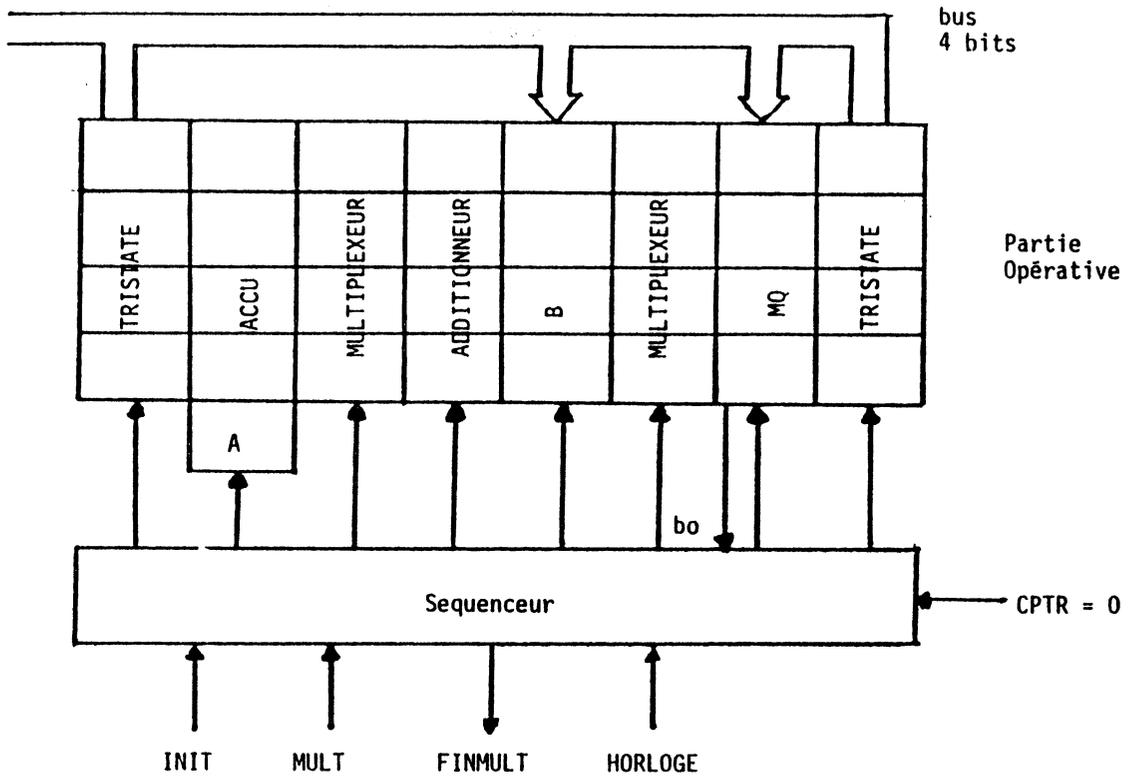
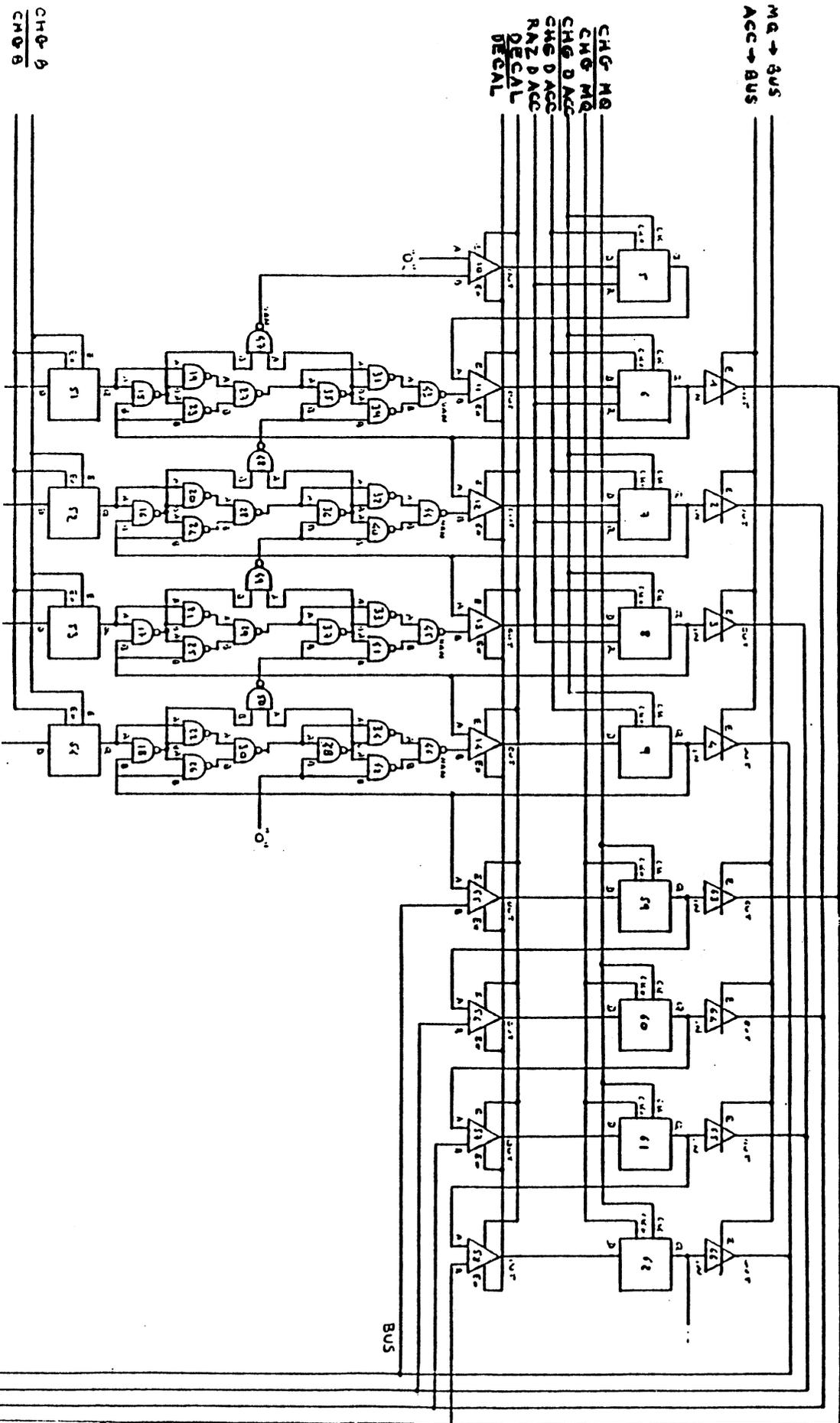


figure 9 : Plan de masse du circuit multiplieur

L'immersion du plan de masse correspondant à la partie opérative à l'intérieur du réseau CMOS a donné les résultats suivants :

Figure 10 : Schéma logique de la partie opérative du multiplieur 4 bits.



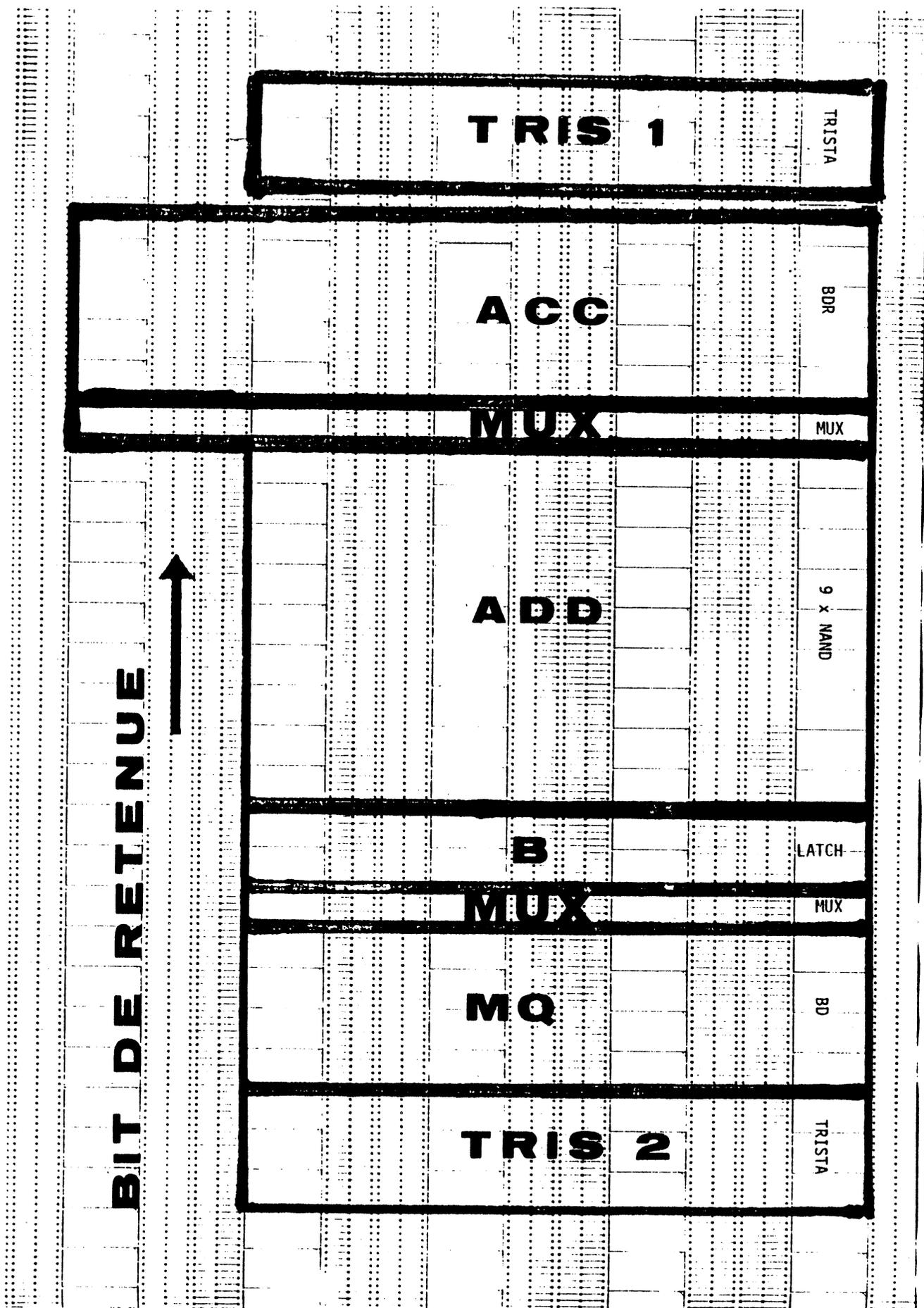


figure 11 : Exemple d'implantation conservant la structure logique du circuit

Une telle méthode de placement des éléments offre plusieurs avantages :

- réalisation rapide (manuellement ou automatiquement moins coûteuse)
- structure fonctionnelle du circuit :
  - . circuit plus facilement vérifiable
  - . circuit plus facilement modifiable.

b) Recherche automatisée d'une structuration du réseau logique

Une structuration automatisée est spécialement recherchée quand les structurations manuelles ou interactives sont inefficaces ou n'ont pas mené à une solution satisfaisante. Nous allons citer des exemples de structuration logique qui ont été utilisées avec succès dans des études pratiques.

1 - Ordonnancement des sommets du réseau logique à partir des points d'entrée-sortie

La répartition des cellules d'entrée/sortie sur la périphérie du circuit permet de définir une préstructuration logique des éléments suivant leur éloignement (profondeur) par rapport à la périphérie. Pour ceci, les points d'entrée-sortie sont mis dans une couche 1. Les (j-1) premières couches étant construites, un élément est classé dans la  $j^{\text{ième}}$  couche s'il est relié à un élément de la  $(j-1)^{\text{ième}}$  couche.

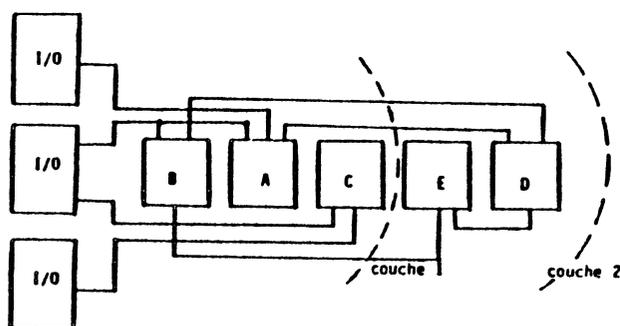
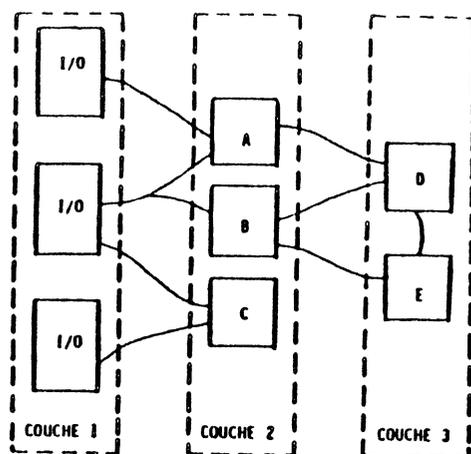


figure 12 : Exemple de construction d'un diagramme en couches

L'indice de profondeur d'un élément est calculé selon la procédure suivante :

- chaque équipotentielle est affecter d'un indice ( $I_{eq}$ ) qui a pour valeur le numéro de la première couche qui contient un élément auquel elle est liée,

- pour chaque élément, l'indice de profondeur est évalué par la somme des indices ( $I_{eq}$ ) des équipotentielles ( $eq$ ) qui lui sont liées, divisée par son nombre total d'équipotentielles.

### Remarques

Le calcul des indices ( $I_{eq}$ ) est effectué en même temps que la construction du diagramme en couches par un algorithme d'expansion dans le graphe du réseau logique.

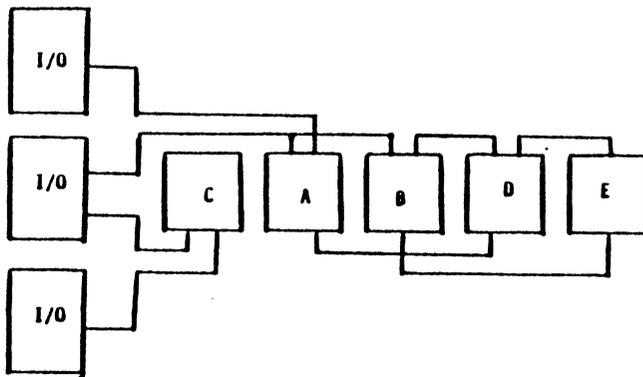
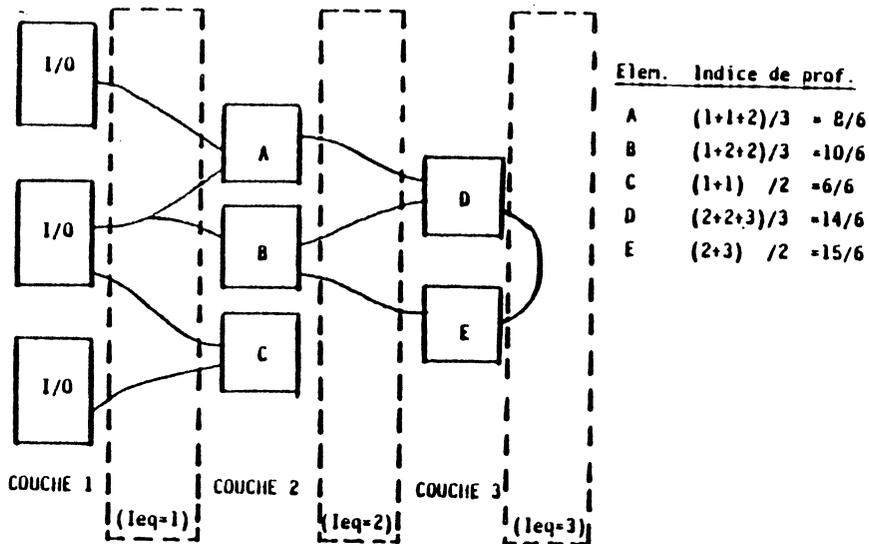


Figure 13 : Construction et interprétation à 1 dimension  
d'un diagramme en couches pondéré

Cette procédure a été appliquée au calcul du noyau central du placement dans le programme PLACA [BEL 84]. Elle a également été utilisée avec succès dans une méthode de placement par développement de noyau sur réseau prédiffusé CMOS (cf. IIème partie), où elle a permis d'affiner l'ordonnancement total de tous les éléments :

Le premier élément choisi étant la ou les cellules dont l'indice de profondeur est maximal, à chaque étape de l'ordonnement, on sélectionne l'élément le plus connecté aux éléments déjà ordonnés et dont l'indice de profondeur est maximum. Des variétés de tels ordonnements sont données dans [BEL 84].

### 2 - Structuration des sommets en classes par bipartitionnements successifs

Une approche par partitionnements successifs [KER 72] est définie comme suit : l'ensemble des éléments est partitionné en deux sous-ensembles en minimisant le nombre de connexions entre ces deux sous-ensembles. Une telle partition est recherchée à nouveau pour chacun des deux sous-ensembles ; ce procédé est renouvelé jusqu'à obtention de classes de dimension souhaitée.

### 3 - Placement initial par l'Analyse de données [BAR 84a]

De nombreux essais de placements ont permis de confirmer l'efficacité et l'apport bénéfique d'une technique de placement originale : le placement initial par l'analyse de données.

A une certaine étape du cycle d'étude d'un problème de placement, on dispose d'un grand tableau rectangulaire résumant les liaisons entre les blocs à placer ou bien les relations entre les blocs et les équipotentielles associées. La formalisation du but à atteindre par une méthode de placement est souvent complexe, imprécise et plus ou moins arbitraire. L'analyse de données propose des techniques permettant, en un temps réduit, de "mettre de l'ordre" dans les données et de suggérer une structure du système en question. Ces méthodes constituent donc une première étape d'analyse permettant d'amorcer de façon intelligente un problème de placement.

On peut ranger les techniques d'analyse de données, en deux grandes familles : les méthodes factorielles et les méthodes de classification automatique.

Les méthodes factorielles se proposent de fournir des représentations synthétiques de vastes ensembles de valeurs numériques. Elles mettent l'accent sur l'aspect positionnement et visualisation des données sur un espace de dimension réduite.

Les techniques de classification automatique, partant des mêmes ensembles de valeurs numériques qui représentent, suivant les cas, une description d'individus ou d'objets à partir d'un certain nombre de variables ou de caractères, sont destinées à produire des regroupements de ces individus ou objets. Le recours aux techniques de classification est sous-tendu par quelques idées générales concernant le champ d'observation. On suppose que certains groupements doivent exister, ou au contraire, on exige que certains regroupements soient effectués. L'intérêt est porté ici sur la mise en évidence des classes d'individus ou des classes de caractères.

Parmi les nombreux avantages offerts par ces techniques nous pouvons citer : la rapidité d'exécution, le faible coût et la facilité d'utilisation des programmes existants. En présence de plus d'une centaine de blocs, une visualisation d'emblée sur le plan est peu intéressante. Nous proposons de segmenter le problème. Les méthodes de classification permettront de regrouper les blocs en un nombre restreint de catégories homogènes, la résolution du problème global est ensuite obtenue par agencement automatique des agrégats. Ces deux techniques sont donc complémentaires.

Les méthodes factorielles ont été appliquées avec succès dans une méthode de placement de blocs rectangulaires [TSI 84]. Les méthodes de classification automatique seront également appliquées dans la troisième partie de cette thèse. De plus amples détails mathématiques relatifs à ces techniques sont exposés dans [BAR 84a].

### 3.2.1.4.2. Placement sur une surface hôte virtuelle

Les limitations d'ordre topologique citées précédemment dans le paragraphe 3.2.1.3. nous ont conduit à immerger la vraie structure hôte dans une structure virtuelle. L'avantage de ce procédé est de "relaxer" le problème du placement vis à vis des contraintes topologiques. Au cours de la construction du placement au sein de la structure virtuelle, l'implanteur respecte toutefois la géométrie de la structure hôte imposée - la relaxation est judicieusement contrôlée de façon à maîtriser des éventuels débordements dans la structure finale obtenue. Une trop forte relaxation pourrait conduire à un placement plus étalé que ne l'exige la vraie structure hôte. Dans ce dernier cas, il se poserait le problème de ramener cette figure à la vraie géométrie imposée. Or, à notre sens, le fait de laisser libre cours à l'évolution du placement pourrait conduire à une ellipsoïde (fig. 14).

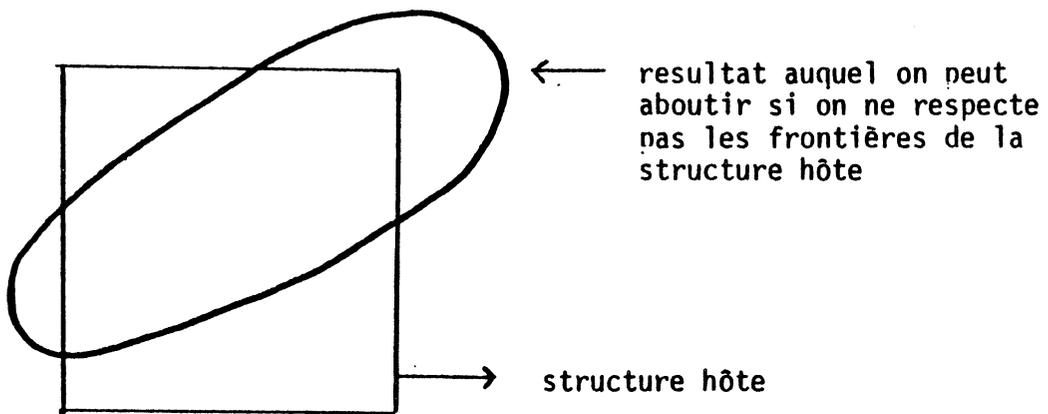


figure 14

La transformation de cette dernière figure en la figure géométrique de la structure hôte initiale fait perdre l'information appréciable acquise par l'optimisation déjà effectuée. De plus, cette transformation n'est pas toujours simple à définir. La démarche adoptée est la suivante :

Etant donnée une structure hôte où il faut placer un circuit, nous déterminons dans un premier temps la meilleure surface englobante (structure virtuelle). Nous construisons à partir du centre de cette structure virtuelle notre placement en respectant la géométrie de la structure hôte réelle.

Exemple 1 : cas où la structure hôte réelle est carrée

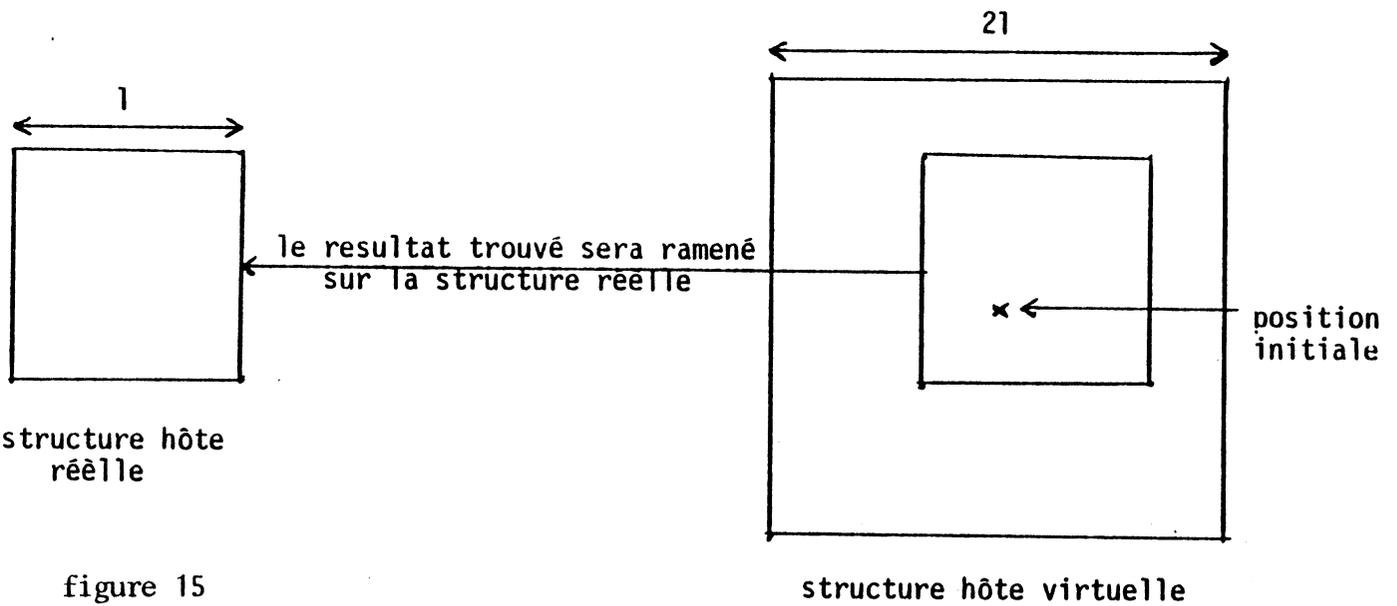


figure 15

Exemple 2 : logique en bande

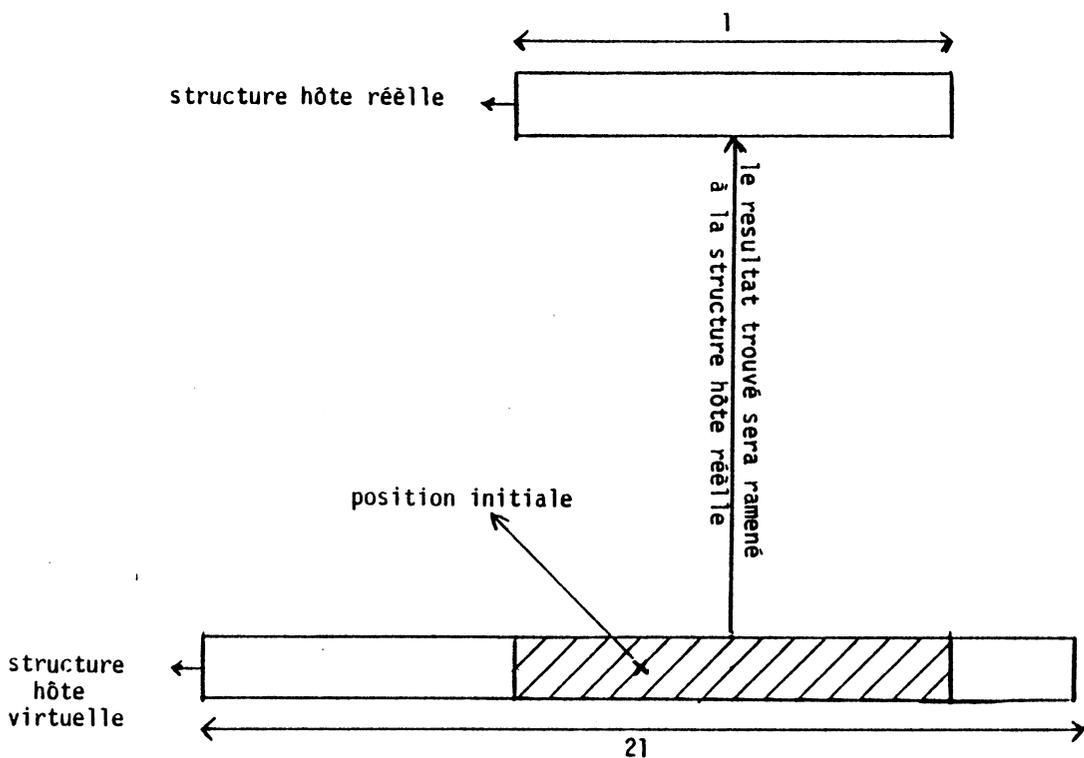


figure 16

Cette approche, très simple mais très efficace, a été appliquée à l'étude présentée dans la deuxième partie de cette thèse.

### 3.2.1.4.3. Préparation du tracé lors de la phase du placement

L'augmentation des densités d'intégration des circuits rend primordiale la préparation du tracé au cours de la phase du placement : certains critères appelés critères de faisabilité devront représenter la première composante du vecteur coût du placement.

$$C(P) = \begin{vmatrix} C_f(P) \\ C_i(P) \end{vmatrix} \quad \text{avec } C_f(P) = \text{coût réduit associé au critère de faisabilité.}$$

$C_f(P)$  doit s'exprimer sous la forme d'une condition. En effet, si la valeur du critère est supérieure à la valeur-seuil prédéterminée, alors l'optimisation de ce critère est prédominante. Par contre si cette valeur est inférieure à la valeur-seuil, l'optimisation du critère peut être abandonnée.

Parmi ces critères de faisabilité qui s'appuient essentiellement sur des mesures de densité, on peut citer :

- une limitation de la densité maximale dans les canaux pour un réseau à canaux d'interconnexion de capacité fixe,
- une limitation de la surface maximale du circuit pour un réseau à canaux de capacité variable (cet objectif est atteint d'abord par minimisation de la densité maximale de connexions dans chaque canal afin de minimiser la largeur nécessaire de ce canal, puis par uniformisation des densités de connexions sur l'ensemble des canaux de façon à atteindre cet objectif sur l'ensemble du réseau hôte).

L'estimation de ces critères est approximative, en effet le tracé n'est pas connu durant l'étape de placement. Cette estimation doit se faire simplement (car utilisée constamment par le programme de placement) et plus ou moins

finement suivant les limitations de ressources de la structure hôte. Dans la deuxième partie de cette thèse, on trouvera un exemple de prédiction fine appliquée avec succès dans le programme IACMOS (implantation automatisée sur réseaux prédiffusés (MOS)).

L'estimation de ces critères repose souvent sur des mesures de densité. Ces mesures se fondent sur la définition de lignes de coupe du réseau. Une ligne de coupe d'un réseau est une ligne de la grille d'implantation, prédéfinie par le concepteur qui traverse tout le réseau perpendiculairement aux canaux ou passages (figure 17).

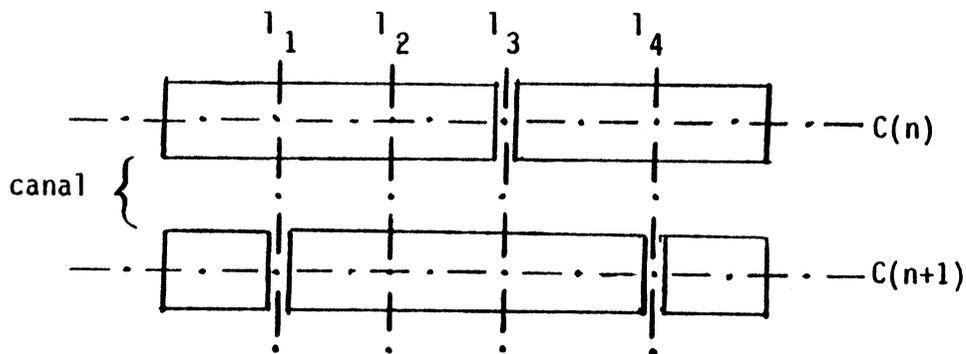


figure 17 : Lignes de coupe d'un réseau en bandes

Le nombre d'équipotentiels qui sont supposées traverser une ligne de coupe au cours de la phase du placement est appelé valeur de densité de cette ligne de coupe. Ces valeurs de densité des lignes de coupe permettront de contrôler les ressources de connectique du réseau hôte. Ainsi, la valeur de densité de la ligne de coupe  $C(n+1)$  permettra de nous informer sur le nombre de passages que nécessite le tracé ultérieur au dessus de la zone active  $(n+1)$ . La valeur maximale de densité des lignes de coupe  $l_1, l_2, l_3, l_4$

dans un canal permettra : d'une part d'évaluer la largeur minimale de ce canal dans les structures à canaux de capacité variable, d'autre part de contrôler un dépassement de capacité dans ce canal pour les structures à canaux de capacité fixe. Des objectifs d'uniformisation et de minimisation des densités peuvent aussi être atteints à partir de ces valeurs de densité. On utilise en général les deux critères suivants :

- minimisation de la valeur maximale de densité

coût élémentaire  $C_1(d) = \max_{c_{1i}} (d_i)$  qui tend à l'uniformisation des

densités sur le réseau

$d_i$  : valeur de densité de la ligne de coupe  $c_{1i}$

- minimisation de la densité totale

coût élémentaire  $C_2(d) = \sum_{c_{1i}} (d_i)$

- Soukup dans [SOU 81] a utilisé

$C_3(d) = \max_{c_{1i}} (d_i) + g \sum_{c_{1i}} (d_i)^2$  avec  $g \geq u$

ce qui lui a permis de tendre simultanément vers l'uniformisation et la minimisation des densités.

Notons enfin que la correspondance entre les critères de densités et leurs évaluations à partir des valeurs de densité des lignes de coupe n'est pas indépendante du nombre de ces lignes de coupe et de leur position sur le réseau hôte. Une étude à ce propos est proposée dans le paragraphe III.7 [BEL 84].

### 3.2.2. Méthodes descendantes

L'objectif essentiel de ces méthodes s'appuie, là encore, sur une vision globale du problème de placement. Elles vont chercher à traiter globalement les éléments du circuit logique en affinant progressivement leurs positions relatives jusqu'à ce que chaque élément soit affecté à une place précise.

Le principe commun de ces méthodes est basé sur des partitionnements successifs du réseau logique en "sous-réseaux" et de l'ensemble des places du circuit en classes appelées macro-places. Chacun de ces partitionnements logiques est basée sur l'heuristique définie par Kernighan et Linn [KER 70]. Le partitionnement topologique correspondant est matérialisé par une ligne de partition.

#### 3.2.2.1. Heuristique de "Kernighan et Linn" [KER 70]

Cette heuristique résoud d'une façon approchée le problème du bipartitionnement optimal de l'ensemble des sommets d'un graphe en deux sous-ensembles.

L'optimisation concerne la minimisation des arrêtes dont les sommets n'appartiennent pas à un même sous-ensemble de la bi-partition. L'heuristique se décompose en deux étapes :

##### a) Partition initiale

L'ensemble des sommets  $X$  du graphe est partitionné en deux sous-ensembles  $X_1$  et  $X_2$ , soit de manière aléatoire, soit de manière constructive. L'existence d'éléments de position fixée dans un problème de placement favorise la construction d'une partition initiale lors du partitionnement de son graphe de connectivité. Il existe de nombreuses heuristiques différentes [KRI 84].

Mais on utilise généralement une des deux approches suivantes :

- remplissage préférentiel d'un côté
  - . tant que  $X_1 \cup X_2 \neq X$  faire
  - . choisir l'élément au coût de sélection maximal, le coût de sélection  $C_s$  est défini par  $C_s = \text{nombre de connexions à } X_1 - \text{nombre de connexions à } X_2$
  - . affecter cet élément à  $X_1$  si ( $C_s$  est positif) et (les conditions de taille sont respectées) sinon l'affecter à  $X_2$ .
- remplissage alterné
  - . tant que  $X_1 \cup X_2 \neq X$  faire
  - . choisir l'ensemble  $X_i$  ( $i=1$  ou  $2$ ) le moins occupé
  - . lui ajouter l'élément qui a le coût de sélection  $C_s$  maximum et dont la taille le permet
$$C_s = \text{connectivité à } X_i - \text{connectivité à } \bar{X}_i \quad \left( C_{X_i}^X \right)$$

#### b) Améliorations itératives

Une amélioration de la partition consiste à définir des sous-ensembles  $A \subset X_1$  et  $B \subset X_2$ , avec  $|A| = |B|$  tels que l'échange entre  $A$  et  $B$  réduise au maximum le nombre d'arrêtes en  $X_1$  et  $X_2$ .

La procédure d'amélioration est reprise avec les nouveaux ensembles  $(X_1 \cup B) - A$  et  $(X_2 \cup A) - B$  et ainsi de suite jusqu'à ce que le résultat soit acceptable. Le critère de l'acceptation est l'arrêt de l'amélioration de la partition ou un nombre d'arrêtes inférieures à un seuil donné.

### 3.2.2.2. Deux catégories de méthodes descendantes

On peut classer les méthodes par bipartitionnements en deux catégories.

#### 3.2.2.2.1. Méthodes à priorité logique

L'algorithme général d'une méthode descendante à priorité logique est le suivant :

a) L'ensemble  $X$  des éléments du circuit est partitionné en deux sous-ensembles  $X_1$  et  $X_2$  selon l'heuristique de Kernighan présentée ci-dessus, de manière à minimiser le nombre d'équipotentiels communes aux deux sous-ensembles. La bi-partition est effectuée sans contrainte de taille sur les deux sous-ensembles  $X_1$  et  $X_2$ . La surface  $S$  du réseau hôte est ensuite partagée par une ligne de partition en deux surfaces  $S_1$  et  $S_2$ . Cette ligne est déterminée par le concepteur et tient compte de la taille des deux sous-ensembles logiques. Chacun des deux sous-ensembles  $X_1$  et  $X_2$  est affecté à une des deux surfaces  $S_1$  et  $S_2$  (fig 18).

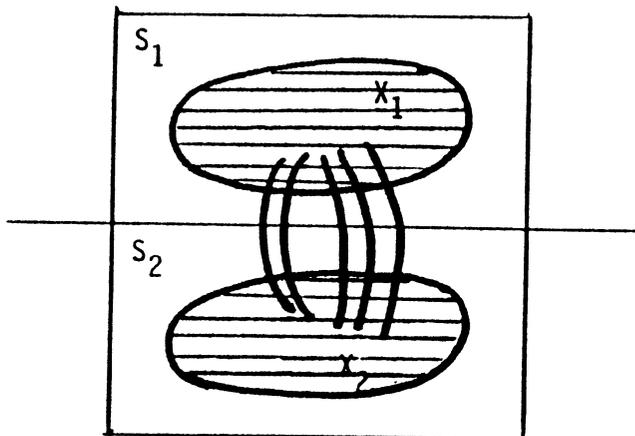


figure 18

b)  $(X_1, S_1)$ ,  $(X_2, S_2)$  sont à leur tour partitionnés suivant le même principe. Le traitement est poursuivi jusqu'à ce que chaque sous-ensemble contienne un seul élément.

Ces méthodes connues dans la littérature sous le nom "block-oriented min-cut" sont employées généralement pour le placement de blocs de tailles et de formes variables [LAU 79]. Elles conviennent particulièrement bien au placement hiérarchisé (guidé par la structure architecturale du circuit) des macro-éléments sur des structures sans contraintes (Préas et Gwyn [PRE 78]).

La figure (MAS) suivante montre le résultat d'un placement de 9 macro-éléments par une méthode "block-oriented min-cut". Le traitement a nécessité 6 itérations soit six lignes de partitions.

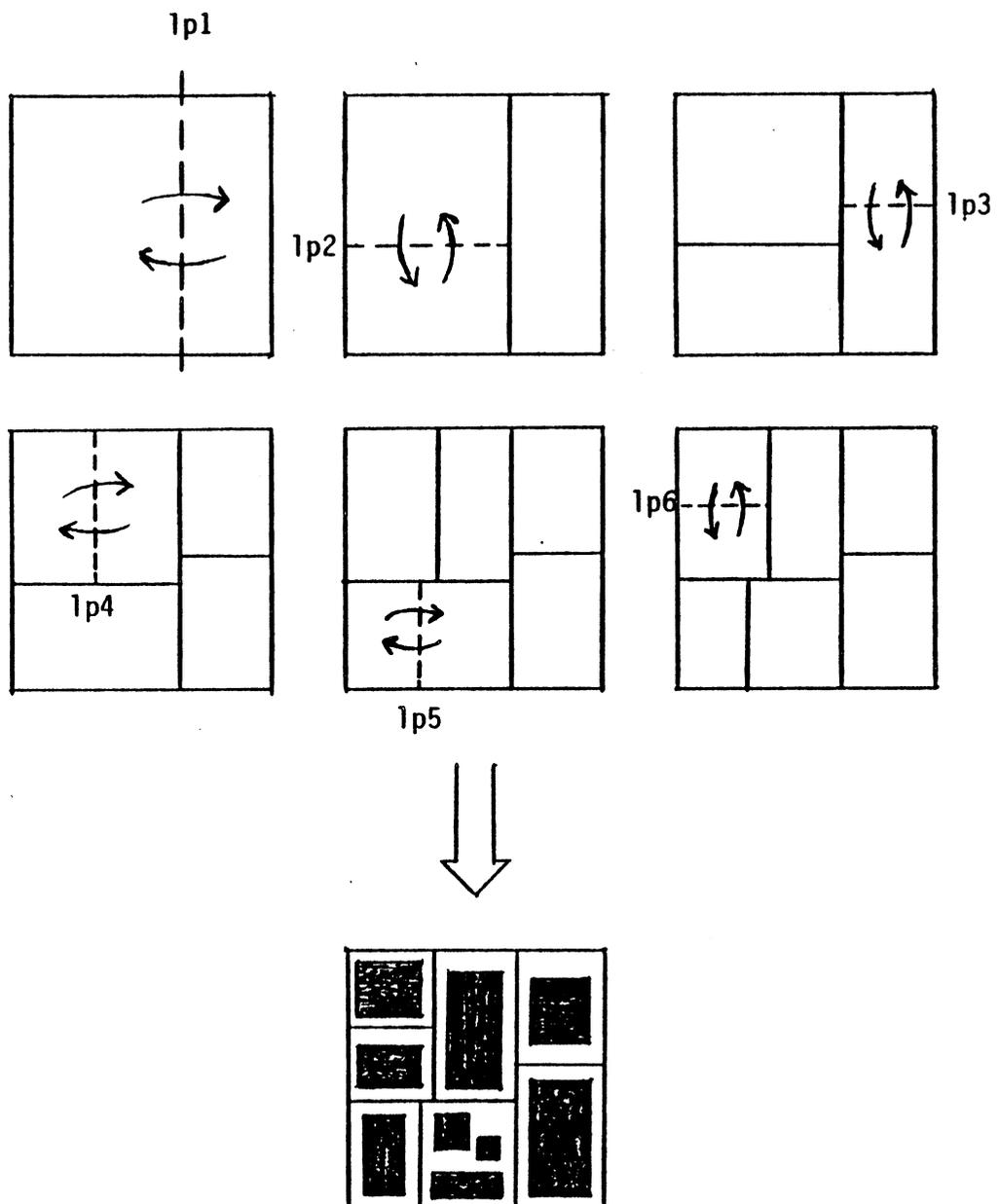


figure 19

Remarquons quand même que cette méthode présente l'ambiguïté suivante :

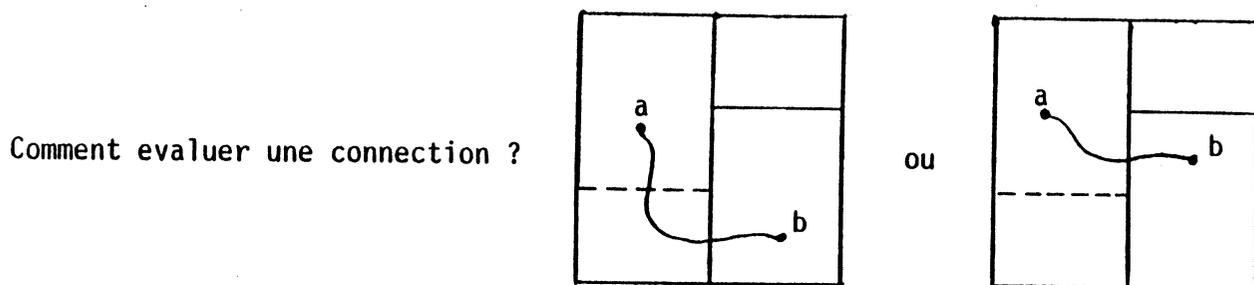


figure 20

#### 3.2.2.2.2. Méthodes à priorité topologique

Pour ce type de méthode , le partitionnement de la structure hôte en macroplaces est réalisé en premier lieu. Il est matérialisé, là aussi, par des lignes de partition. La position de l'ensemble de ces lignes peut être prédéterminée. Ceci, par exemple, est le cas pour le réseau prédiffusé ACE étudié dans la thèse [BEL 84] et pour le réseau prédiffusé Ferranti étudié dans la thèse [TSI 84]. Cette position peut aussi être définie par le concepteur. Ce dernier poursuit le traitement en réalisant séquentiellement les bipartitions logiques.

Ces bipartitions doivent d'une part, respecter les contraintes de taille des macroplaces décidées, d'autre part minimiser la densité des connexions sur les lignes de partition définies.

Cette stratégie est généralement utilisée sur les réseaux dont la structure topologique est connue (réseaux structurés). Ces méthodes sont caractérisées par la séquence de traitement des lignes de partition. Parmi les différentes séquences expérimentées dans la littérature technique, nous allons citer les deux exemples suivants.

- placement par quadrature : séquence alternée de lignes de partition verticales et horizontales (fig 21).

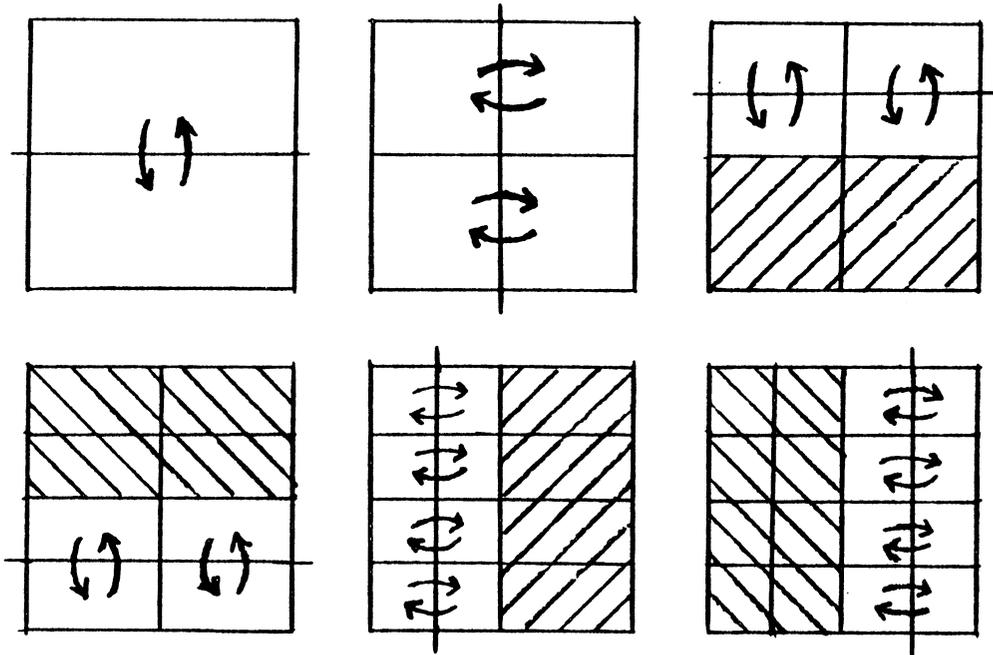


figure 21

Cette méthode n'a pas d'ambiguïté pour évaluer une connexion (fig 20) mais son processus d'optimisation séquentielle a appauvri la qualité du résultat global. De plus la méthode ne doit pas s'adapter à un placement de blocs de taille variable. Elle est particulièrement adaptée aux réseaux non structurés ou de structure matricielle. Une application de cette méthode est illustrée dans [BUR 83].

- placement par bisection: le réseau est partitionné verticalement (resp horizontalement) en deux blocs de taille égale ; puis chaque bloc est partitionné en deux parties égales suivant la même direction. Quand toutes les lignes de partition verticales (resp horizontales) ont été traitées, le même processus est appliquée aux lignes de partition perpendiculaire. Cette méthode est appliquée plus particulièrement aux réseaux structurés en bandes (fig 22).

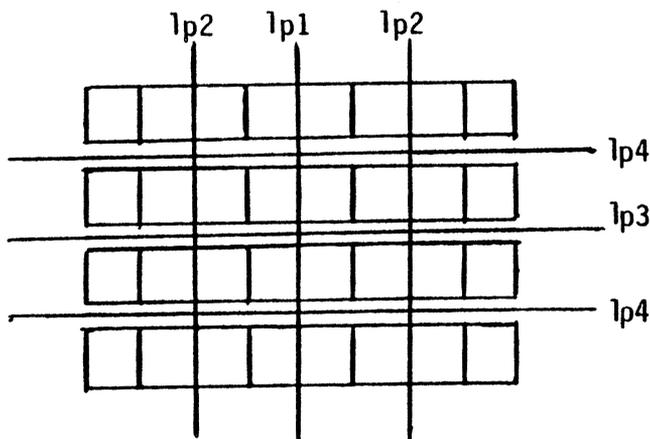
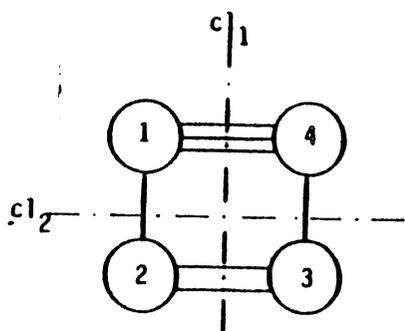


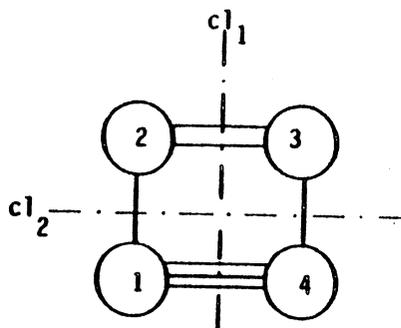
figure 22: Bisection

### 3.2.2.2.3. Discussion des méthodes descendantes

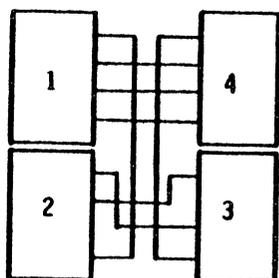
Il est vrai que ces méthodes permettent une vision globale du placement. Les relations de connectivité entre les éléments, les contraintes de faisabilité du placement, prises en compte globalement sur les réseaux logiques et topologiques, peuvent favoriser la minimisation et l'uniformisation des densités de connexions sur le réseau hôte. Cependant elles présentent un inconvénient de base qui est l'indétermination sur la position des groupes d'éléments équivalents dans chaque partition. Cette figure tirée de la thèse de [BEL 84] illustre cet inconvénient.



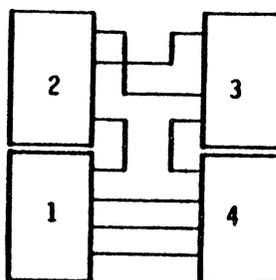
a) partition du graphe  
suivant  $c1_2$  puis  $c1_1$



b) partition équivalente  
à la partition a



a') résultat topologique de la  
partition a  
densité max = 4  
longueur des conn. = L



b') résultat topologique de la  
partition b  
densité max = 2  
longueur des conn. = L-4l

figure 23 : Indétermination sur une partition

Les deux partitions des figures 23a et 23b sont parfaitement équivalentes au sens de l'optimisation définie au cours de ce paragraphe mais les résultats des figures 23a' et 23b' sont très différents à cause de la position des contacts sur les éléments partitionnés : il existe une indétermination de la partition suivant  $c1_2$  sur l'affectation aux sous-blocs "haut" et "bas" des ensembles équivalents (2,3) et (1,4).

Cet inconvénient est particulièrement gênant car rencontré, avec une forte probabilité au début du placement, quelque soit la méthode descendante choisie.

De plus l'efficacité d'une méthode descendante dépend essentiellement du choix des lignes de partition et de la séquence de traitement de ces lignes.

Comme le montre, par exemple, la figure suivante [BEL 84] :

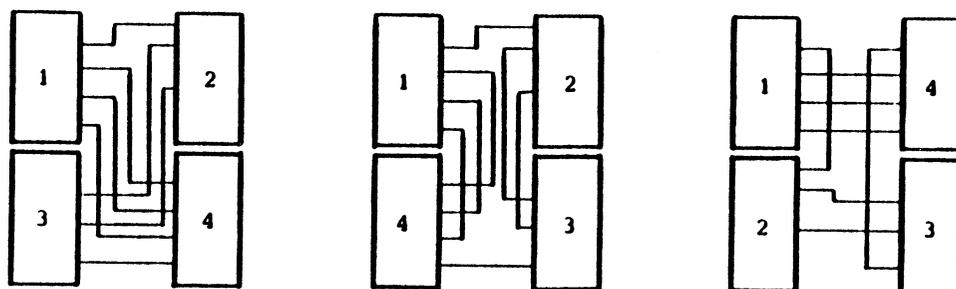
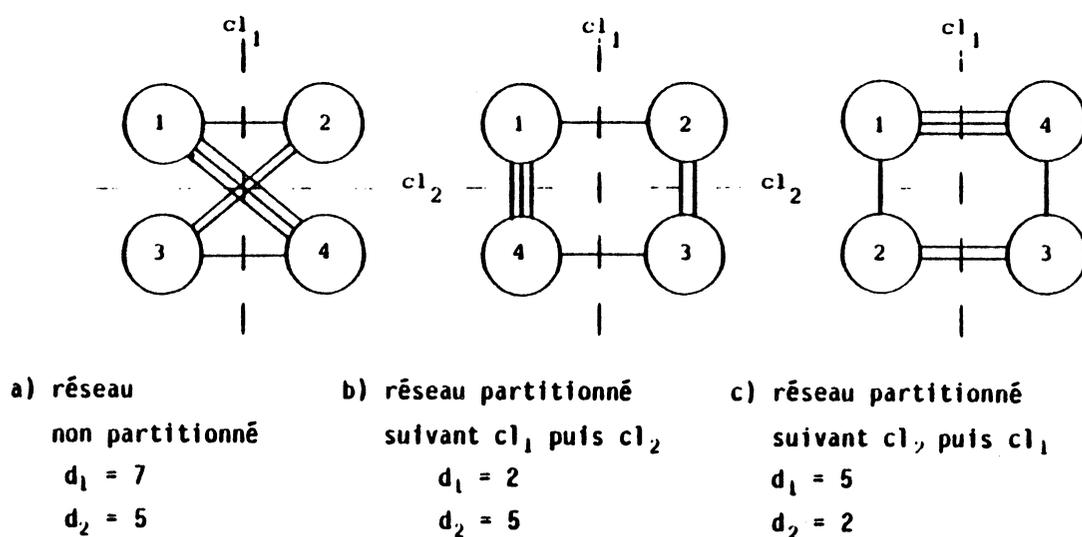


Figure 24 : Influence de l'ordre de traitement des lignes de partition

Dans un paragraphe intitulé "Etude des méthodes de placement par bi-partitionnements" [BEL 84], A. BELLON s'est intéressé de près à ce problème. Il a ainsi dégagé des règles d'ordonnement des lignes de partition. Ces règles qui ont pour but de réaliser effectivement l'optimisation des critères de densité de connexions sur le réseau hôte, ont conduit à définir pour chaque structure, une évaluation précise de l'ordre de traitement des lignes de partition, donc, à définir une méthode par bi-partitionnement optimale pour ce type de structure.

#### IV - METHODES ITERATIVES D'AMELIORATION

La plupart des méthodes de placement exposées dans la littérature, calculant une solution non optimale et néanmoins acceptable, sont basées sur le principe suivant : l'utilisation d'une méthode constructive de placement initial suivie d'une méthode d'amélioration itérative.

Dans le paragraphe précédent, nous avons présenté les tendances les plus significatives dans le domaine des méthodes de placement initial. Nous avons aussi proposé quelques moyens pour affiner les heuristiques correspondantes.

Dans ce paragraphe nous allons nous intéresser aux méthodes d'amélioration itératives. Ces méthodes interviendront dans le cas où la stratégie constructive n'aurait pas pu délivrer un placement initial satisfaisant. Elles sont basées sur le principe d'itérations consistant à faire évoluer un placement en échangeant certains éléments ou en appliquant des opérations de rotation ou de symétrie à chacun des éléments, jusqu'à ce qu'un placement satisfaisant soit obtenu. On distingue deux types de méthodes d'améliorations itératives : d'une part les méthodes stochastiques qui à chaque itération,

appliquent au placement initial les transformations ci-dessus d'une manière aléatoire et à des éléments choisis au hasard, d'autre part les méthodes dirigées dont ces transformations sont effectuées selon des procédures prédéfinies.

#### IV - 1. Méthodes stochastiques

Ces méthodes particulièrement longues n'étaient pas utilisées dans le problème du placement. Néanmoins, une heuristique récemment développée par Kirkpatrick [KIR 83] est appliquée avec succès dans des problèmes où le temps de calcul n'est pas critique. Cette méthode intitulée "Amélioration du placement par la méthode des recuits successifs", a la particularité d'accentuer un nouveau placement, même s'il est moins bon, mais avec une probabilité qui dépend de la différence des coûts entre les deux placements. Elle permet ainsi de rompre des minima locaux de la fonction coût.

##### 4.1.1. *Amélioration du placement par la méthode des recuits successifs (Simulated Annealing)*

La méthode du recuit est un procédé expérimental destiné à trouver l'état stable d'un système composé de particules en interaction. Cette technique issue de la mécanique statistique, peut-être dérivée à bon escient au problème de l'amélioration itérative d'un placement. Cette dérivation peut se justifier au moyen des différentes analogies suivantes établies entre les particules matérielles et les blocs à placer :

- la présence ou l'absence de connexion entre des blocs permettent de définir respectivement des notions de forces d'attraction ou de répulsion interagissant entre ces différents blocs.

- l'énergie interne E du système est assimilée au coût du placement,
- les variations de la configuration du système dépendent d'un coefficient T dit de "température".
- pour chaque modification du placement :
  - . si l'énergie du système (le coût du placement) est améliorée, la transformation est acceptée,
  - . si l'énergie du système est dégradée, la transformation est acceptée avec une probabilité de  $e^{\frac{-\Delta E}{kT}}$  (K = constante de Boltzmann).

Le système est tout d'abord porté à une haute température. La "température" est alors abaissée progressivement. La recherche est arrêtée lorsque le système se retrouve dans une configuration pratiquement identique durant trois baisses successives de la température.

Pour plus de détails concernant cette approche nous renvoyons le lecteur à [TSI 84] où cette méthode a été appliquée avec succès à des problèmes où peu de critères doivent être simultanément optimisés. Pour le cas général cette méthode reste longue (baisse progressive du coefficient de température, calcul expérimental des coefficients T, k).

#### IV - 2. Méthodes dirigées

Dans cette approche, une procédure d'échange entre les éléments est prédéfinie. Elle est appliquée à des éléments vérifiant certaines conditions. Un échange n'est accepté que si le coût du placement est amélioré. La transformation est effectuée jusqu'à ce que le placement ne soit plus amélioré.

#### 4.2.1. Echange de $n$ éléments à la fois

A chaque itération, un nombre  $n$  des éléments sont permutés, et les critères sont évalués. Les éléments sont traités par ordre d'importance de leur connectivité aux éléments placés.

La solution est meilleure quand  $n$  est grand, mais le temps de calcul augmente avec  $n$ . De plus, quand les éléments n'ont pas la même géométrie et les mêmes dimensions il devient difficile d'effectuer l'échange des  $n$  éléments à la fois. En général, la valeur de  $n$  est choisie égale à 4 ou 5 pour avoir un compromis entre la qualité de la solution et le temps de calcul.

Parmi ces méthodes on peut citer la méthode FDR (Force Directed Relaxation) décrite dans [HAN 73]. Puis son amélioration GFDR (General Force Directed Relaxation Method) introduite par GOTO [GOT 79].

#### 4.2.2. Echange par paire

A chaque itération, le placement est modifié par l'échange de deux éléments. Deux algorithmes d'itération sont possibles :

- toutes les configurations permises par la procédure d'échanges sont générées, et la solution au meilleur coût est conservée,
- on arrête le processus d'échange à la première solution qui améliore le coût.

Le choix des éléments à échanger peut-être réalisé de diverses manières :

- échange systématique des  $n(n-1)/2$  paires obtenues lors d'un placement de  $n$  éléments [SE 80],
- échange par paires systématiques des  $K$  éléments compris dans une fenêtre de dimension fixe ( $K.(K-1)/2$  échanges) [HAN 72].

- énumération des  $2^K$  configurations de placement par échange vertical (resp horizontal) de paires d'éléments sur des lignes (resp colonnes) adjacentes,

- échange par paires d'éléments fortement connectés [GLA 59],

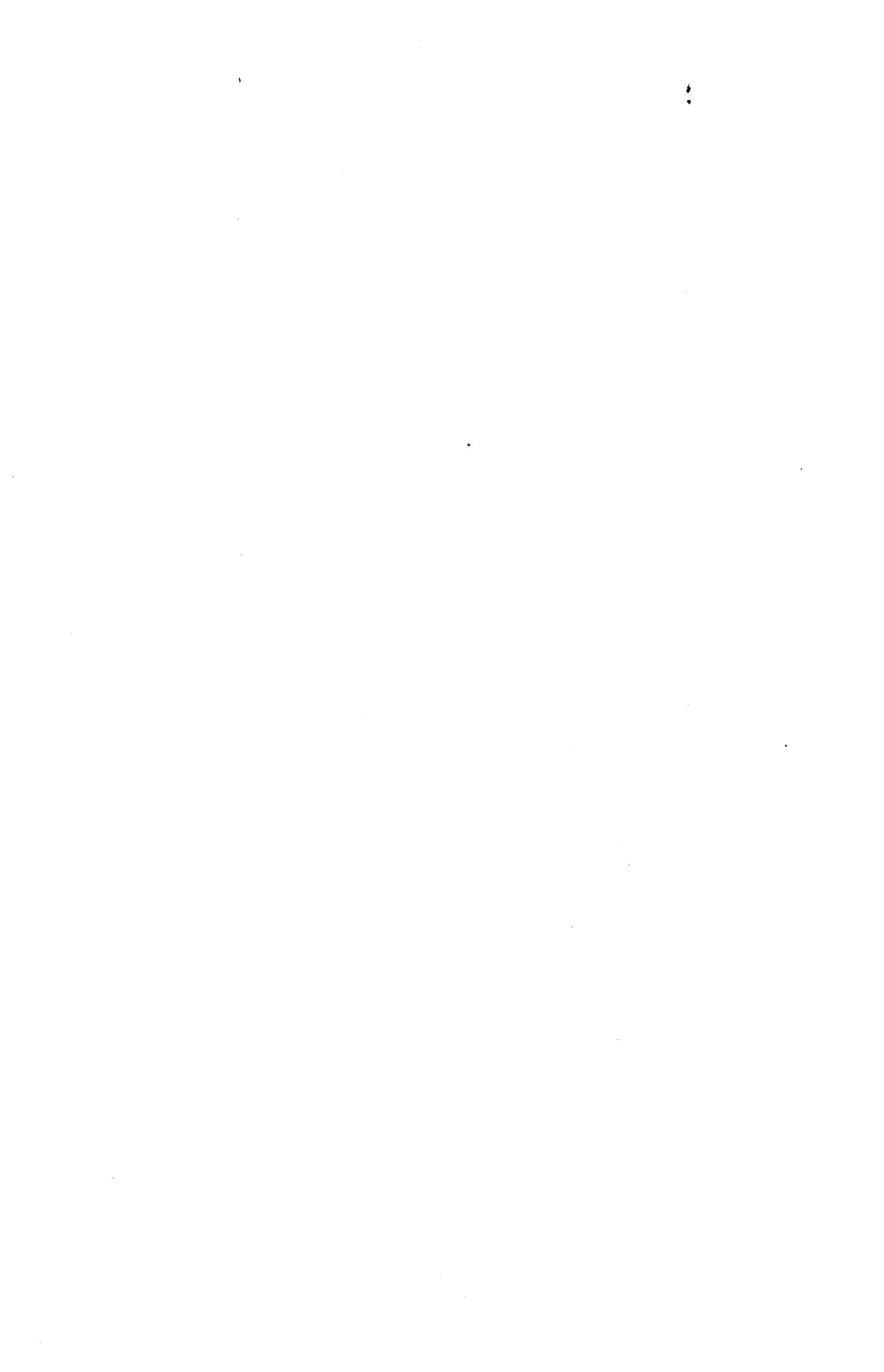
- échange de l'élément candidat avec un élément proche de sa position de relaxation. La position de relaxation d'un élément est une place où le vecteur force de cet élément est nul. Le vecteur de force associé à un élément représente la résultante des forces d'attraction exercées sur lui par les éléments placés auxquels il est connecté. Les règles de calcul du vecteur de force et de la position de relaxation permettent de définir plusieurs méthodes de ce type. A. BELLON dans un paragraphe de sa thèse intitulé "Méthodes par relaxation" a donné les principales méthodes par relaxation qu'on rencontre dans la littérature technique [BEL 84].

Pour conclure sur les méthodes dirigées, nous dirons que l'amélioration itérative d'un placement par des procédures d'échange limitées entre 2 éléments est la plus utilisée dans la pratique. Lorsque la taille des éléments est très différente, celle-ci constitue une véritable limitation aux possibilités d'améliorations du placement par des procédures d'échange. Enfin pour le placement de blocs rectangulaires où cette limitation devient vraiment un obstacle on applique à chacun des éléments des opérations de rotation ou de symétrie [LAN 79][SER 81 ][PRE 78].

## CONCLUSION

La complexité croissante des circuits rend les problèmes de placement de plus en plus critiques. La méthode la plus couramment utilisée dans la pratique consiste à construire un placement initial (par une méthode ascendante ou une méthode descendante) et de l'améliorer par une méthode itérative. Les algorithmes de placement constructif, bien que du même ordre de complexité que ceux d'amélioration itérative, traitent plus rapidement les problèmes où le nombre d'éléments à placer est grand.

Une étude préalable et rapide des caractéristiques logiques du réseau à immerger (prestructuration logique des sommets du réseau, prédiction - pas trop coûteuse - de la connectique dès le placement initial ....), une prise en compte élégante des caractéristiques topologiques de la structure hôte (adaptation des règles de sélection des éléments et des places, placement sur structure hôte virtuelle ...), permettent de mieux appréhender le problème i.e de sélectionner et d'affiner les méthodes de placement initial les plus adaptées à chaque structure afin d'éviter dans beaucoup de cas l'amélioration ou au moins de diminuer le temps de calcul requis par celle-ci.



**PARTIE II**

**L'IMPLANTATION AUTOMATISEE DE CIRCUITS  
SUR RESEAUX PREDIFFUSES CMOS**



## INTRODUCTION

Cette deuxième partie présente une application des principes définis précédemment à la conception d'un système d'implantation automatisée sur réseaux prédiffusés CMOS (Complementary Metal Oxyde Semiconductor) : le système IACMOS. Ce système a été développé pour s'intégrer dans un ensemble de programmes CAO déjà existant à la société CIT-ALCATEL de Paris. C'est pourquoi en plus des problèmes généraux posés par les réseaux prédiffusés CMOS. On insistera sur les problèmes spécifiques du réseau CIT.

Dans le premier chapitre, nous décrirons le réseau CIT-ALCATEL sur lequel porte cette étude. Dans le second, nous présenterons l'organisation générale du système IACMOS.

Les chapitres III et IV seront consacrés à la présentation détaillée des méthodes utilisées dans le système IACMOS pour le placement des modules d'une part et le tracé des connexions inter-modules d'autre part.

## I - PRESENTATION DU RESEAU CIT-ALCATEL

I - 1. Description de la structure prédiffusée

Le réseau prédiffusé de CIT-ALCATEL est réalisé en technologie CMOS. IL se présente sous la forme de bandes horizontales (zones actives) composées de places élémentaires ou cellules de base. Ces bandes accueillent les différents éléments disponibles (dans une bibliothèque de cellules I.2) qui utilisent une ou plusieurs places élémentaires. Elles sont séparées par des canaux permettant de tracer les connexions entre éléments logiques. De plus, la périphérie du circuit est réservée aux cellules d'entrée/sortie. Deux types de réseaux, similaires mais de tailles différentes, sont disponibles :

- UCI 400 portes (1 porte = 1 place élémentaire)

39 cellules de bases	351
. 9 rangées de	
2 cellules de passages périphériques	18
. 4 x 9 Pad d'entrée:sortie	<u>36</u>
	405

- UCI 1200 portes

70 cellules de base	1 120
. 16 rangées de	
2 cellules de passages périphériques	32
. 4 x 16 Pad d'entrée/sortie	<u>64</u>
	1 216

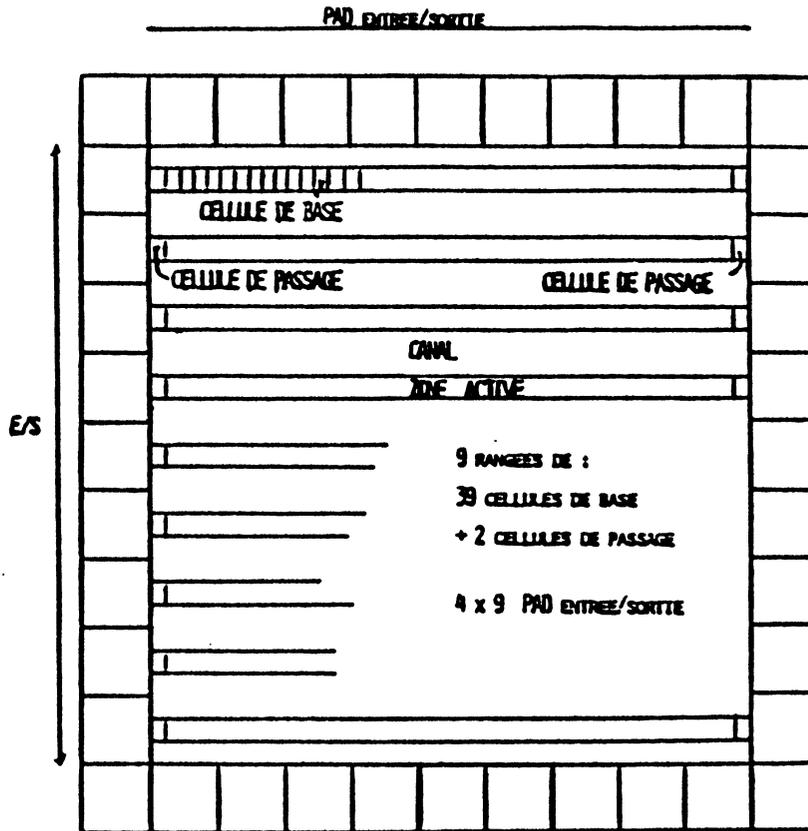


Figure 25 : Matrice UCI 400

- Les canaux d'interconnexion inter-zones actives sont constitués de deux rangées de "barettes" en Silicium Polycristallin possédant chacune trois contacts (figure 26).

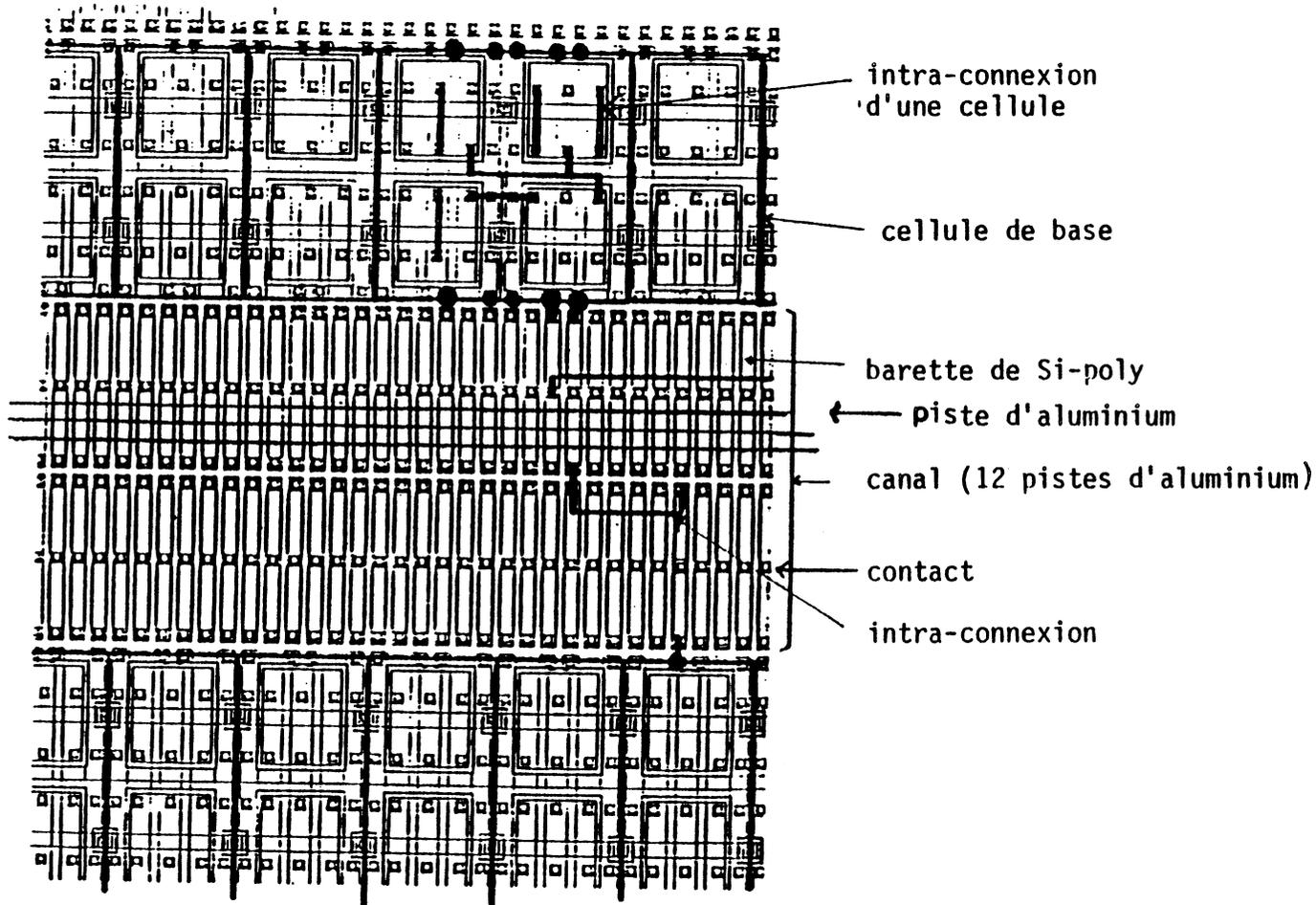
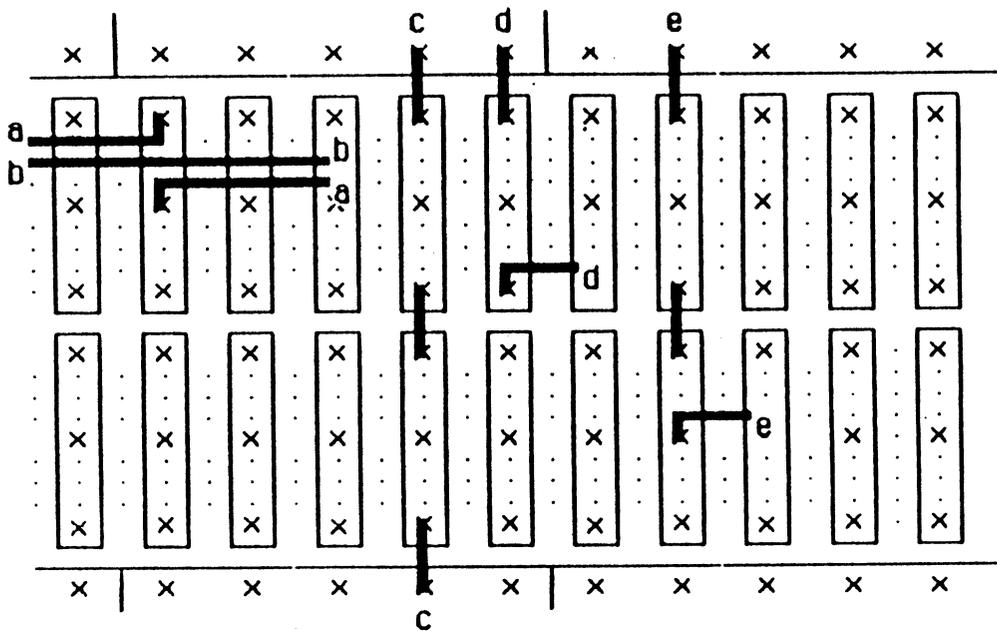


Figure 26 : CMOS CIT-ALCATEL

Les canaux d'interconnexion entre cellules internes et cellules périphériques contiennent une seule rangée de ces barettes.

Entre deux contacts d'une même barette peuvent passer trois connexions en aluminium. Ce qui permet de faire passer jusqu'à 12 fils (resp. 6) dans la largeur d'un canal inter-zones actives (resp. d'un canal entre cellules internes et cellules périphériques). Tout fil d'aluminium passant sur un des contacts est en contact électrique avec la barette correspondante. Le croisement de deux connexions a et b s'effectue aisément (figure 27).



a-b croisement

c traversée directe du canal

d connexion sur piste (partie haut du canal)

e connexion sur piste (partie basse du canal)

Figure 27

La cellule de base est constituée d'éléments actifs non connectés qui sont pré-implantés sur la plaquette. Elle est symétrique et comprend six points d'entrée/sortie de chaque côté de la bande de façon à tomber en face des barettes en polysilicium. Elle comporte également deux passages verticaux, c'est à dire des liaisons électriques entre les deux bords de la cellule n'intervenant en rien sur son fonctionnement et permettant ainsi de faire passer une connexion d'un canal à l'autre.

### I - 2. Description de la bibliothèque de cellules

La conception de circuits en réseaux de cellules utilisent des fonctions logiques prédéterminées dont la complexité peut varier du simple opérateur logique (NOR, NAND, Inverseur) aux macro-fonctions logiques plus complexes telles que batch-bascules-comparateur (voir même ALU, RAM ou ROM dans le cas de réseaux précaractérisés). L'implantation fine de ces fonctions est préalablement optimisée électriquement et topologiquement. Les cellules ainsi définies sont réunies dans une bibliothèque, de cellules où elles sont décrites par leurs fonctions logiques, topologiques et leurs performances électriques. Chaque fonction logique peut posséder plusieurs solutions d'implantation différentes (taille, forme, position des points de contact).

Dans le cas étudié, on dispose d'une bibliothèque de 50 éléments logiques et de 15 cellules d'entrée/sortie (cellules périphériques) prédéfinies. Toute cellule de bibliothèque est symbolisée par un rectangle (ensemble de cellules de base) dans lequel apparaissent les informations suivantes :

- le nom de la cellule donné par l'utilisateur,

- l'emplacement et le nom de référence des entrées/sorties de la cellule. Pour les éléments logiques les passages verticaux sont repérés par le symbole § suivi d'un chiffre permettant de les différencier.

Les éléments logiques tiennent sur une bande de cellules de base. Leur taille varie de une à six places élémentaires. On trouvera à la suite (figure 28) la liste de ces éléments ainsi que leurs tailles respectives. On trouvera aussi aux figures (29,30,31) quelques exemples de ces cellules avec leur représentation symbolique.

Les cellules périphériques sont caractérisées électriquement et topologiquement (un exemple de telles cellules est donné (figure 32)).

A cette liste s'ajoute une cellule fictive appelée FT (pour Feed Through) qui ne possède pas d'intraconnexion et laisse libres deux passages à travers la zone active.

L'implantation d'une cellule de bibliothèque est réalisée par le placement des masques d'intraconnexion prédéfinis de cette cellule sur une ou plusieurs cellules de base disponibles. La personnalisation du réseau en un circuit particulier est réalisé en définissant uniquement les interconnexions d'aluminium entre les éléments actifs.

nom	fonction	taille
INV	1 inverseur	1
INVS	2 inverseurs en série	1
INVP	2 inverseurs en parallèle	1
INVD	2 inverseurs indépendants	1
NAND2	1 nand 2 entrées	1
NAND3	1 nand 3 entrées	2
NANDI3	1 nand 3 entrées + 1 inverseur	2
NAND4	1 nand 4 entrées	2
AND2	1 and + nand 2 entrées	2
ANDI2	1 and + nand 2 entrées + 1 inv.	2
AND3	1 and + nand 3 entrées	2
AND4	1 and + nand 4 entrées	3
ANDI4	1 and + nand 4 entrées + 1 inv.	3
NOR2	1 nor 2 entrées	1
NOR3	1 nor 3 entrées	2
NORI3	1 nor 3 entrées + 1 inverseur	2
NOR4	1 nor 4 entrées	2
OR2	1 or + nor 2 entrées	2
ORI2	1 or + nor 2 entrées + 1 inv.	2
OR3	1 or + nor 3 entrées	2
OR4	1 or + nor 4 entrées	3
ORI4	1 or + nor 4 entrées + 1 inv.	3
EXOR	1 or exclusif + nor 2 entrées	3
EXNOR	1 nor exclusif + nand 2 entrées	3
TGATE	1 porte de transfert	1
MUX	1 multiplexeur 2 entrées	1
TRISTA	1 buffer 3 états	3
TRISTA*	1 buffer 3 états inverseur	2
RSNOR	1 bascule RS à portes nor	2
RSNAND	1 bascule RS à portes nand	2
COMTRI	commande 3 états avec buffer	5
COMTRI1	commande 3 états	3
COMTRIX	commande 3 états inversée	2
TRIGGER	1 trigger de schmitt	2
LATCH	1 bascule D	2
LATCHR	1 bascule D avec reset	3
LATCHS	1 bascule D avec set	3
LATCHRX	1 bascule D avec reset inversé	3
LATCHSX	1 bascule D avec set inversé	3
BD	1 bascule D maître/esclave	4
BDR	1 bascule D m/s avec reset	5
BDS	1 bascule D m/s avec set	5
BDRS	1 bascule D m/s avec set et reset	6
BDRX	1 bascule D m/s avec reset inv.	5
BDSX	1 bascule D m/s avec set inversé	5
BDRSX	1 basc. D m/s avec set et res. inv.	6
VDD	niveau logique 1	1
VSS	niveau logique 0	1
VDDVSS	niveaux logiques 1 et 0	1

Figure 28 : Catalogue des cellules CIT-ALCATEL

2 INVERSEURS  
Indépendants

INVD

INVERSEURS  
Indépendants

INVD

CARACTERISTIQUES DYNAMIQUES

CAPACITE D'ENTREE : CA = 0.1PF  
CB = 0.1PF

TEMPS DE PROPAGATION : t<sub>PH</sub> OU t<sub>PL</sub>  
 t<sub>PHL</sub> < t<sub>PH</sub> - 1.6NS/PF  
 t<sub>PLH</sub> < t<sub>PL</sub> - 1.6NS/PF  
 TEMPS DE TRANSITION : t<sub>THL</sub> < t<sub>TH</sub> - 1.6NS/PF  
 t<sub>TLH</sub> < t<sub>TL</sub> - 1.6NS/PF

CELLULE



TABLE DE VERITE

A	A*	B	B*
H	L	H	L
L	H	L	H

LISTE D'APPEL SPLICE

Nom A\* A B\* B INVD

SCHEMA LOGIQUE

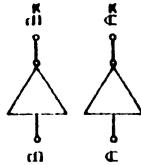


Figure 29

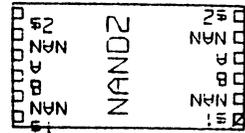
# NAND2

NAND  
2 entrees

CARACTERISTIQUES DYNAMIQUES:

- CAPACITE D'ENTREE : CA = 0.3PF  
CB = 0.1PF
- TEMPS DE PROPAGATION MAX/A ou MAX/B : TPLH < 3NS + 7NS/PF  
TPLH < 4NS + 7NS/PF
- TEMPS DE TRANSITION : TTLH < 4NS + 14NS/PF  
TTLH < 5NS + 16NS/PF

CELLULE



# NAND2

NAND  
2 entrees

TABLE DE VERITE

A	B	NA	NB
H	H	L	L
H	L	L	H
L	H	H	L
L	L	H	H

LISTE D'APPEL SPICE

Nom NAR A B NAND2

SCHEMA LOGIQUE

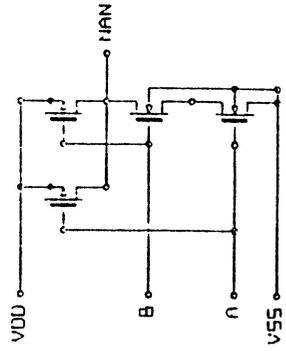


Figure 30

# BD BASCULE D

## CARACTERISTIQUES DYNAMIQUES

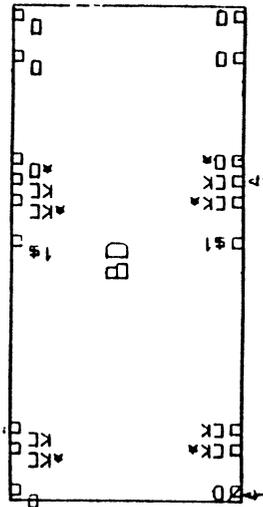
CAPACITES D'ENTREE  
 : CD = 0.4PF  
 CCK = 0.2PF  
 CCK\* = 0.2PF

TEMPS DE PROPAGATION  
 O/CK : TPHL < 8.5NS + 4NS/PF  
 TPLH < 6.5NS + 7NS/PF  
 O\*/O : TPHL < 2NS + 3.5NS/PF  
 TPLH < 2.5NS + 7NS/PF

TEMPS DE TRANSITION  
 O : TTHL < 6NS + 7NS/PF  
 TTLH < 8NS + 16NS/PF  
 O\* : TTHL < 3NS + 7NS/PF  
 TTLH < 2NS + 16NS/PF

TEMPS PREPOSITIONNEMENT : TSU > 7.5NS  
 TEMPS DE MAINTIEN : TMO > 2NS  
 TEMPS PROPAGATION CK/CK\* : THHB < 13NS  
 TEMPS TRANSITION CK,CK\* : TTH < 35NS

## CELLULE



# BDR BASCULE D AVEC RESET

## TABLE DE VERITE

D	CK	R	O
X	X	H	L
H	/	L	H
L	/	L	L
X	L	L	O

## LISTE D'APPEL SPICE

Nom O G\* D CK CK\* R BDR

## SCHEMA LOGIQUE

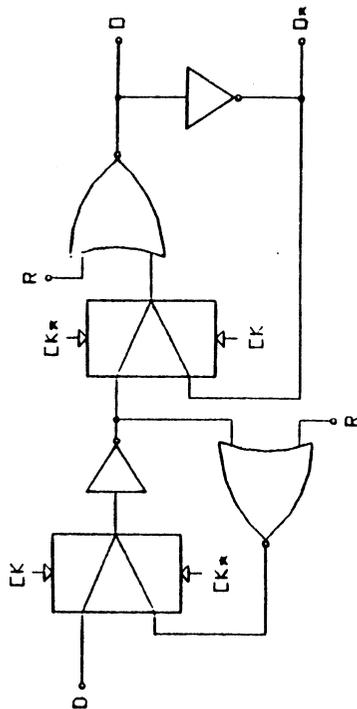


Figure 31



# BUF100DIR E/S DIRECTE

# BUF100DIR E/S DIRECTE

## CARACTERISTIQUES DYNAMIQUES

CAPACITE D'ENTREE

: CTP = 1PF  
CTN = 3PF

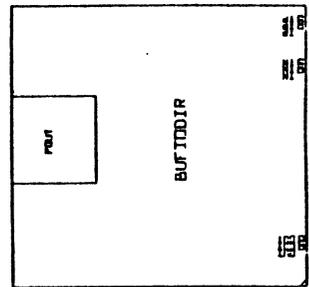
TEMPS DE PROPAGATION OUT/POUT

: TP<sub>HL</sub> < 0NS + NS/PF  
: TP<sub>LH</sub> < 0NS + NS/PF  
: TP<sub>HL</sub> < 7.5NS + 0.5NS/PF  
: TP<sub>LH</sub> < 6NS + 0.5NS/PF

TEMPS DE TRANSITION OUT

: TT<sub>HL</sub> < 0NS + NS/PF  
: TT<sub>LH</sub> < 0NS + NS/PF  
: TT<sub>HL</sub> < 21.5NS + 0.5NS/PF  
: TT<sub>LH</sub> < 10NS + 0.5NS/PF

CELLULE



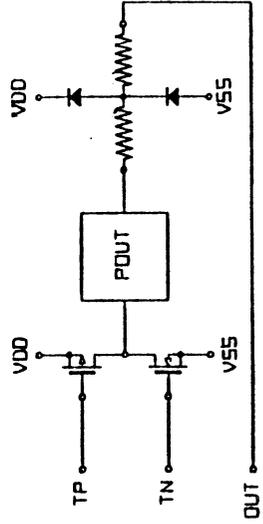
## TABLE DE VERITE

TP	TN	POUT	OUT
H	H	L	L
H	L	Z	X
L	H	X	X
L	L	H	H

## LISTE D'APPEL SPLICE

Nom POUT OUT TP TN BUF100DIR

## SCHEMA LOGIQUE



### I - 3. Problèmes spécifiques

L'implantation automatisée sur réseaux prédiffusés CMOS (une couche d'aluminium de métallisation) se heurte à des problèmes spécifiques :

#### - dus à la structure des réseaux

- . la position et le nombre des cellules de base sont fixés,
- . la capacité des canaux d'interconnexion est fixée,
- . la traversée des bandes se fait par des emplacements précis (feed through) en nombre réduit,
- . les contacts entre la couche d'aluminium et les barettes de silicium polycristallin sont prédéfinis en nombre et en emplacement.

#### - dus à la technologie :

- . l'emploi de silicium polycristallin induit un ralentissement du signal, dû à l'effet capacitif et resistif important de celui-ci. On va donc avoir besoin de minimiser l'emploi des barettes de silicium pour conserver des vitesses acceptables.

Les deux méthodes (placement + routage) qui seront présentées dans les chapitres (III et IV) décriront une manière de faire face à cette limitation des ressources, en recherchant comme objectif principal le tracé de toutes les connexions.

### I - 4. Données du problème

Le système IACMOS étant destiné à s'intégrer dans un ensemble de programmes de CAO déjà existant chez CIT. Les formats de données en entrée et des résultats en sortie du programme sont fixés au départ par l'utilisateur.

On dispose en entrée de 4 fichiers :

- une description du réseau employé,
- une description de la bibliothèque de cellules,
- une net-list (liste des équipotentiels), une équipotentielle est une connexion électrique connectant deux ou plusieurs éléments,
- un fichier de codage/decodage du nom des cellules de bibliothèque.

On devra fournir en sortie :

- après placement, un fichier "résultat-pour-routage" indiquant pour chaque élément du circuit à implanter, son emplacement sur une zone active (bande) bien déterminée. Ce fichier servira aussi d'entrée pour le programme de tracé.
- après routage, un fichier de segments décrit par couple de coordonnées (x,y) réelles, classés par canaux d'interconnexion.

## II - PRESENTATION GENERALE DU SYSTEME IACMOS

### II - 1. Description du système

Le système d'implantation automatique IACMOS réalise le placement et le tracé des connexions sur les deux types de réseaux prédiffusés CMOS (une couche de métallisation) décrits précédemment. Il se divise en trois parties ;

- l'acquisition des données
- le placement des modules effectué par le programme PLACMOS (placement automatisé en CMOS),
- le tracé des connexions inter-modules effectué par le programme TRACMOS (tracé automatisé en CMOS). Ce programme a été réalisé principalement par B. MARTINET (L.C.S).

. La mise en place de la structure de données, formée essentiellement de listes de cellules et d'équipotentiels, s'effectue à partir de la lecture des 4 fichiers d'entrée cités précédemment dans I - 4. Une description détaillée de la structure mise en place serait assez lourde et de peu d'intérêt pour le lecteur. Il est à noter qu'elle comporte à peu près 300 lignes de Pascal.

. La méthode de placement consiste à construire (selon une approche structurée) un placement initial et de l'améliorer en cas de besoin par une méthode itérative.

. La méthode de routage comporte deux phases :

- une phase de routage global où, après éclatement des équipotentiels en segments, on affecte ces segments d'une part aux canaux en gérant la densité d'occupation de ceux-ci et d'autre part aux "feed-throughs",
- une phase de tracé fin par "greedy channel router" [RIV 82] dans les canaux pour résoudre les problèmes de contact aluminium/Si Poly et affecter les équipotentiels aux pistes.

## II - 2. Domaine d'application du système IACMOS, description d'une structure proposée pour 2 couches en AL

Le système IACMOS a été développé principalement pour l'implantation automatique de circuits logiques sur le réseau hôte à fortes contraintes de la société CIT-ALCATEL (réseau prédiffusé en technologie CMOS à une couche d'aluminium). Il peut aussi être appliqué sur d'autres structures qui présentent les mêmes caractéristiques.

Le programme PLACMOS par exemple peut facilement être appliqué sur une structure à deux couches d'interconnexion en AL. En particulier, sur celles proposées par le laboratoire "Circuits et Systèmes" dont voici un exemple.

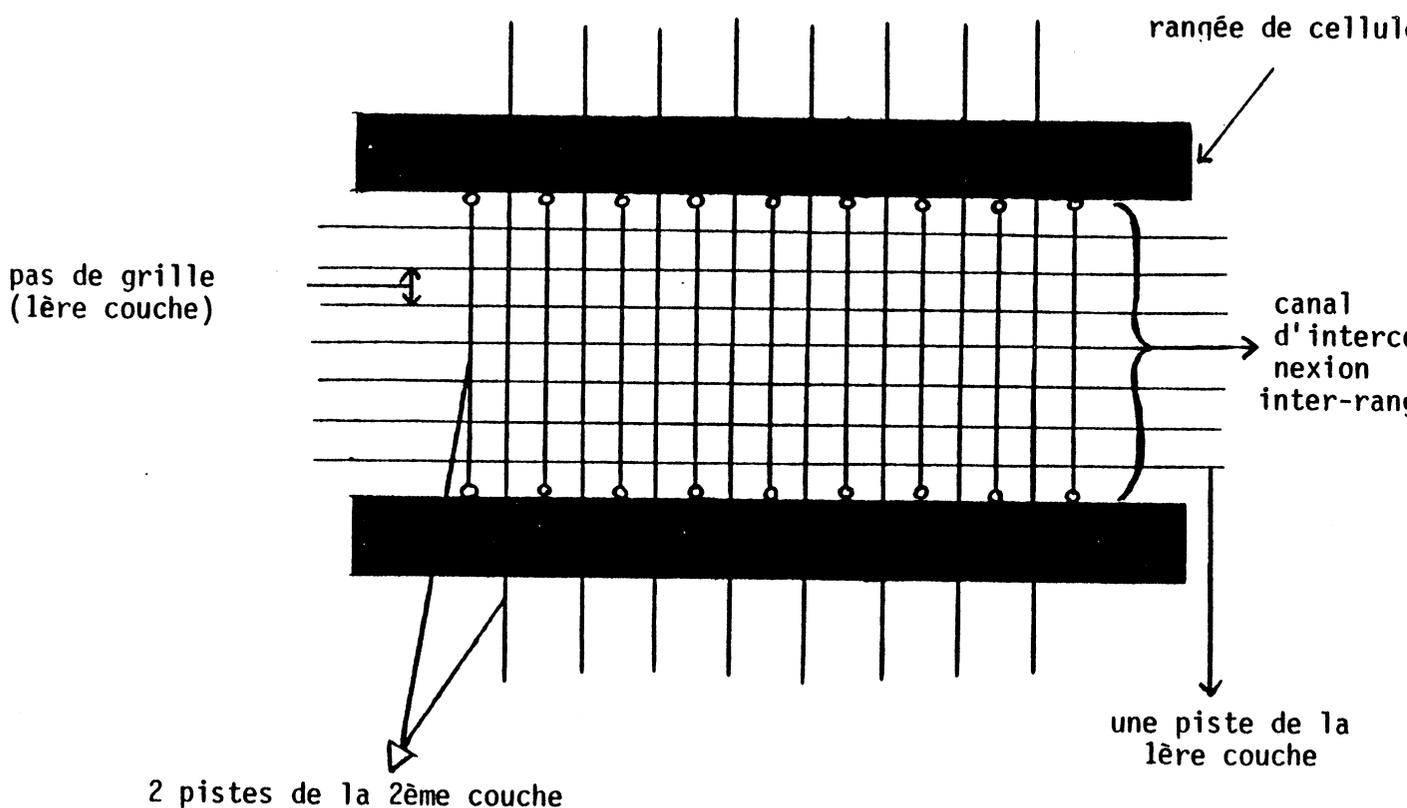


Figure 33

- la première couche est réservée pour les connexions horizontales,
- la deuxième couche est réservée pour les connexions verticales qui sont de deux types : connexions entre deux rangées de cellules adjacentes, connexions pour traverser les rangées de cellules,
- 0 : représente contact entrée/sortie sur bloc (fig 33).

Nous pouvons remarquer figure 33 que :

- sur les rangées de cellules, les connexions utiliseront forcément la deuxième couche,
- le pas de grille de la deuxième couche sur les rangées de cellules est le double du pas de grille dans les canaux.

#### Avantages d'une telle solution

- possibilité d'utiliser un "channel router" utilisant des structures semblables,
- utilisation de toutes les pistes horizontales de la première couche à l'inverse d'autres solutions qui n'interdisent pas un changement de couche pour une même direction (verticale, horizontale), et qui nécessitent de nombreux contacts.

### III - PLACEMENT

#### III - 1. Description générale de la méthode utilisée

La grande taille des circuits à implanter et des réseaux prédéfinis proposés par CIT-ALCATEL condamne l'utilisation des méthodes "exactes" de résolution (Partie A). La méthode retenue est une méthode approchée conduisant à une solution acceptable dans un temps de calcul relativement court. Elle comporte donc deux phases :

- une phase de placement initial constructif,
- une phase d'amélioration itérative.

La première phase de cette méthode est obligatoire. Elle est employée quelque soit le circuit à implanter. Pendant cette phase les éléments sont placés séquentiellement sur le réseau. Le passage à la phase suivante dépend du succès/échec de la première phase. Pour certains circuits (circuits très denses), ce passage sera jugé nécessaire. Certains éléments seront alors déplacés de façon à améliorer le placement.

L'originalité de la méthode relève de la première phase. La construction du placement initial se fait selon une version améliorée de "cluster development method". Cette amélioration repose sur l'application de quelques idées nouvelles émises dans la partie A de cette thèse (III.2.1.4). La méthode utilise une préstructuration logique du circuit à implanter et travaille dans un premier temps, sur une structure hôte virtuelle. Cette approche permet de guider la recherche d'un emplacement optimisé pour les éléments logiques du circuit. Cette optimisation concerne les critères de placement liés au tracé : implantation de toutes les connexions, minimisation et

uniformisation des densités estimées de connexions dans les canaux. De plus cette méthode, par sa forte prédiction de connectique va diminuer la nécessité ou la longueur d'une phase d'amélioration itérative.

L'organigramme général de la phase du placement peut être représenté par la figure suivante :

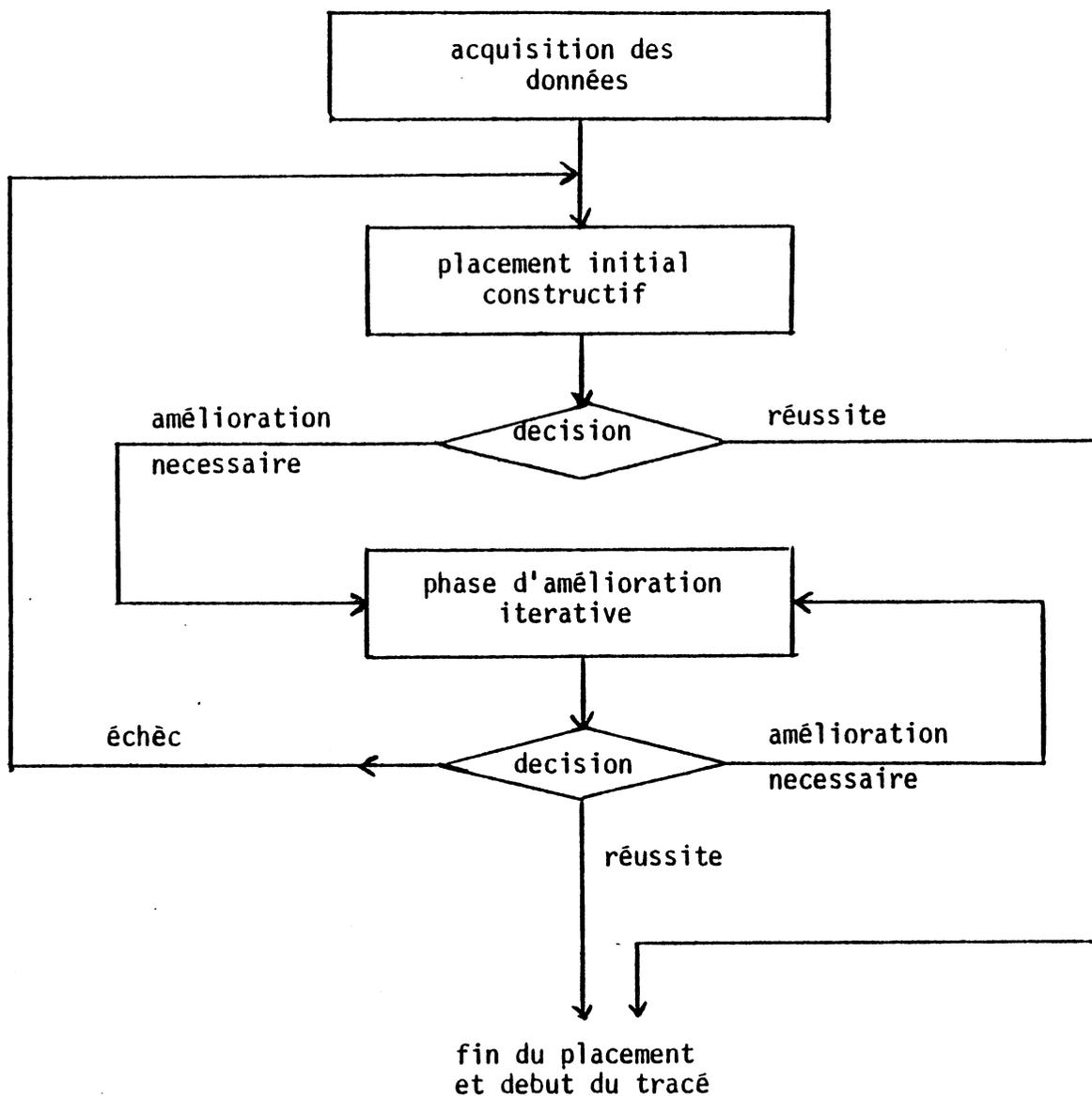


Figure 34

### III - 2. Placement initial constructif

Cette phase comprend trois parties essentielles. Son organigramme général est représenté par la figure suivante :

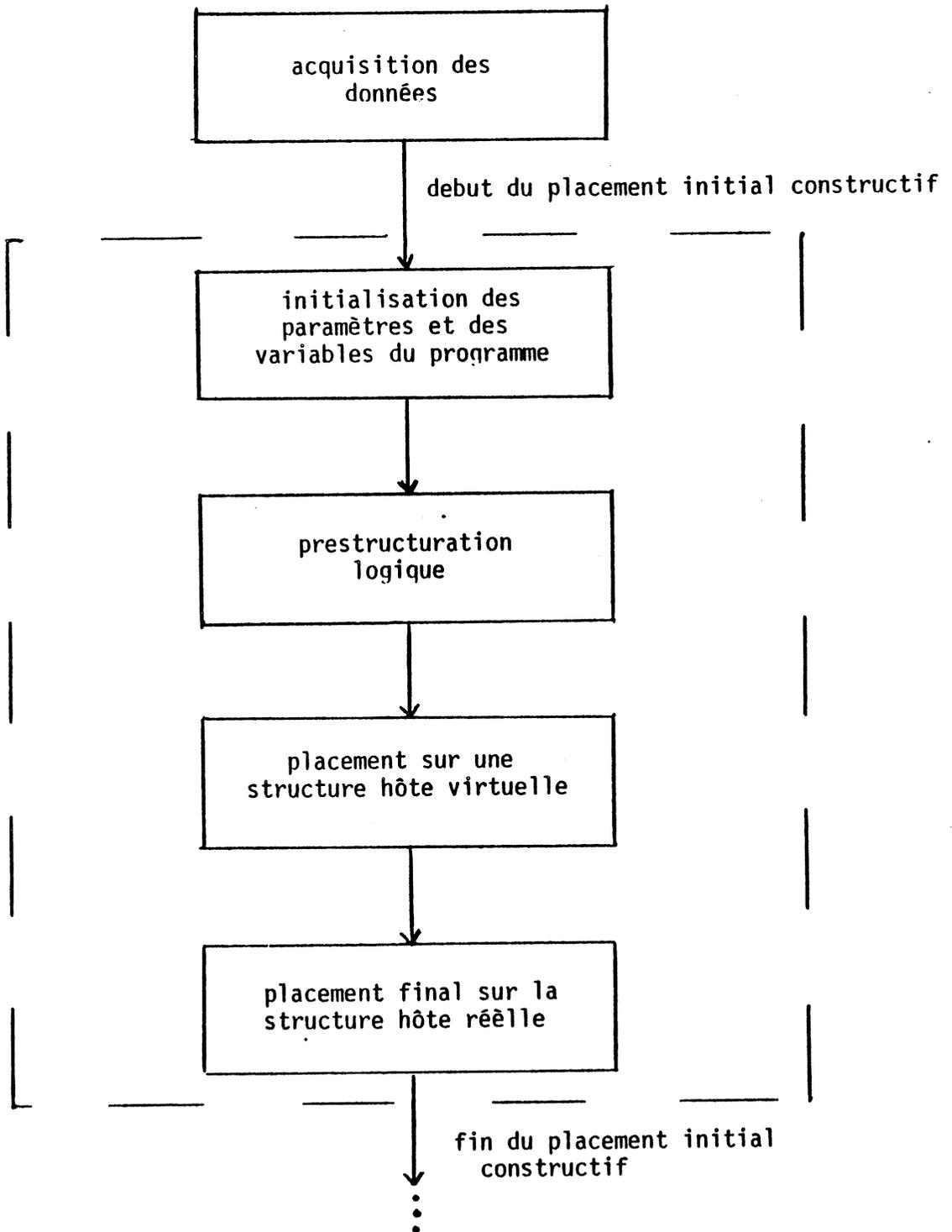


Figure 35

Nous allons dans ce qui suit détailler les 3 parties essentielles de cet organigramme.

### §.2.1. Préstructuration logique

Nous avons vu dans la première partie de cette thèse que pour les méthodes constructives d'un placement initial : le placement est formé en joignant des modules à un sous-ensemble (noyau) d'éléments déjà placés, que ces méthodes opèrent itérativement sur l'ensemble des éléments non placés en sélectionnant un de ces éléments et en le positionnant sur une place non occupée, et qu'enfin les règles de sélection du module d'une étape donnée sont basées essentiellement sur des critères de connectivité entre ce module et l'ensemble des modules déjà placés indépendamment de leur position. Les modules peuvent donc être totalement ordonnés avant toute application d'une règle de positionnement.

Pour cela, on affecte d'abord tout élément à place imposée à sa place (ceci étant fait lors de l'initialisation de la structure de données utilisée) ; puis, pendant cette phase de préstructuration logique, notre méthode ordonne totalement tous les éléments restants (à place non imposée) à partir d'un élément central choisi parmi eux selon la procédure suivante :

- l'élément central est l'élément dont l'indice de profondeur est maximum et qui est le plus connecté,
- à chaque étape de l'ordonnement, l'élément sélectionné est l'élément le plus connecté aux éléments déjà ordonnés et dont l'indice de profondeur est maximum.

Le critère "indice de profondeur" a été défini dans la première partie de cette thèse (III.2.1.4.1.3.). Il permet ici d'ajouter une indispensable vision à deux dimensions aux règles de sélection des éléments pour mieux choisir l'élément central et pour discriminer efficacement les éléments équivalents (au sens du critère de connectivité) dans les phases extrêmes de cet ordonnancement (au début du traitement, peu d'éléments sont ordonnés donc le critère de sélection basé sur la connectivité est évalué très partiellement. En fin de traitement, la majorité des éléments sont déjà traités, ce critère est équivalent pour un grand nombre d'éléments).

### 3.2.1.1. Discussion

a) L'ordonnancement est fait selon maximisation du vecteur coût composé des deux critères :

$C_1$  : nombre de connexions aux éléments déjà ordonnés

$C_2$  : indice de profondeur

La comparaison entre deux vecteurs coûts  $C : \begin{vmatrix} C_1 \\ C_2 \end{vmatrix}$  et  $C' : \begin{vmatrix} C'_1 \\ C'_2 \end{vmatrix}$  est faite selon la relation d'ordre totale définie dans la partie A de cette thèse

$$(II.2) : C < C' \Leftrightarrow \begin{cases} C \neq C' \\ \text{et } (C_1 < C'_1) \text{ ou } ((C_1 = C'_1) \text{ et } (C_2 = C'_2)) \end{cases}$$

Nous avons essayé sur plusieurs exemples de modifier l'ordre de priorité de ces deux critères selon différentes alternatives : au début de l'ordonnancement après le traitement d'un certain pourcentage des éléments du circuit ....

Nous n'avons en aucun cas amélioré les résultats ceci nous a permis de conclure que pour notre méthode de développement de noyau le critère de connectivité demeure prioritaire sur le critère "indice de profondeur".

b - L'algorithme place aussi les cellules périphériques dont les places ne sont pas imposées. Au début du travail nous avons au cours de cet ordonnancement négligé le type de l'élément. Nous avons donc confondu les cellules internes et les cellules périphériques. Une cellule périphérique pouvait ainsi être sélectionnée et (ultérieurement placée) avant une cellule interne si le nombre de ses connexions aux éléments déjà ordonnés était plus important. Cependant, l'essai d'une stratégie qui consistait à ordonnancer tout d'abord toutes les cellules internes et puis ensuite toutes les cellules périphériques a permis d'obtenir de meilleurs résultats. Nous avons donc retenu cette dernière solution.

c - Nous avons aussi été confrontés à un problème au cours de la première étape qui est le choix de l'élément central (noyau). En effet, dans tous les exemples de circuit traités, plusieurs éléments (au moins 2) avaient le même indice de profondeur (le maximum) et possédaient le même nombre de connexions avec le reste du circuit. Devant ce conflit nous avons envisagé deux alternatives :

#### Première alternative

Considérer tous les  $l$  éléments candidats pour le noyau de circuit comme un seul élément (en supposant qu'il y a  $l$  tels éléments) ce qui permet de passer après directement à la  $(l + 1)^{\text{ième}}$  étape de l'ordonnancement.

#### Deuxième alternative

On choisira aléatoirement un de ces  $l$  éléments comme noyau du circuit.

La première alternative semble être plus raisonnable à première vue puisqu'elle affecte les éléments les plus profonds du circuit logique au centre du réseau hôte. Cependant elle présente des inconvénients constatés au cours de l'expérimentation.

Nous allons, pour des raisons de simplicité, expliciter cet inconvénient pour le cas  $l = 2$ . Supposons que ces 2 éléments (" $x_1$ " et " $x_2$ ") n'aient pas un même élément auquel ils sont liés tous les deux (ceci est d'ailleurs souvent le cas). A la troisième étape l'élément " $x_3$ " sélectionné sera lié à un et un seul de ces deux éléments soit par exemple à  $x_1$ .

L'élément de la quatrième étape a plus de chances de provenir de  $\{x_1, x_3\}$  et ainsi de suite ... avec cette alternative, la méthode va donc ordonnancer et dans la suite placer tous les éléments de la suite provenant de  $x_1$  avant de pouvoir reconsidérer l'élément  $x_2$ .

En fait, le problème soulevé au début de cette remarque (c) peut être rencontré à n'importe quelle étape de l'ordonnancement et non seulement à la première. La maximisation du vecteur coût défini dans la remarque (a) peut être réalisée à n'importe quel moment par plusieurs éléments distincts. Face à de tels conflits, l'expérimentation nous a conduit à adopter la deuxième alternative pour résoudre de tels problèmes.

d - Le critère de connectivité entre un élément non-traité et un ensemble d'éléments traités étant le critère primordial des règles de sélection des éléments dans cette phase de préstructuration logique, nous avons essayé les deux principales mesures de la connectivité utilisées dans la littérature technique :

1 - La connectivité entre un élément candidat et un ensemble d'éléments traités est mesurée comme la somme des pondérations des arcs entre ces ensembles dans le graphe de connectivité du circuit. Ce graphe est défini comme suit :

Les sommets du graphe correspondent aux éléments du réseau logique. Une arête du graphe représente l'existence d'au moins une connexion commune aux deux éléments représentés par les sommets de cette arête. Chaque arête est pondérée par le nombre de connexions qu'elle présente.

2 - La mesure de connectivité est évaluée simplement par le nombre d'équipotentiels qui relient l'élément candidat à l'ensemble déjà traité. On a pu constater, au cours de l'expérimentation encore, que la deuxième mesure est plus réaliste que la première. Une variante de cette deuxième mesure aurait pu être employée : la connectivité est évaluée dans ce cas [BEL 84] par la différence entre le nombre de connexions de l'élément candidat liées aux éléments traités et le nombre total de ses connexions. Nous appellerons cette valeur connectivité relative d'un élément. Le choix d'un élément dont la connectivité relative est maximum permet de définir l'élément qui est le plus connecté aux éléments traités et le moins connecté aux éléments non-traités mais présente malheureusement l'inconvénient suivant ; il favorise les éléments qui possèdent peu de connexions par rapport à ceux qui en possèdent beaucoup.

### 3.2.2. Placement sur une structure hôte virtuelle

Après la préstructuration logique, la seconde idée de base de notre méthode est le placement sur une structure hôte virtuelle. Nous avons vu dans les paragraphes III.2.1.3 et III.2.1.4.2. de la première partie de cette thèse comment, dans le but de construire un placement bien optimisé, nous étions amenés dans un premier temps à affecter les éléments du circuit sur une structure hôte virtuelle puis à immerger ensuite le placement trouvé sur la vraie structure hôte.

Dans notre cas d'étude nous avons pris la structure hôte virtuelle quatre fois plus grande que l'initiale

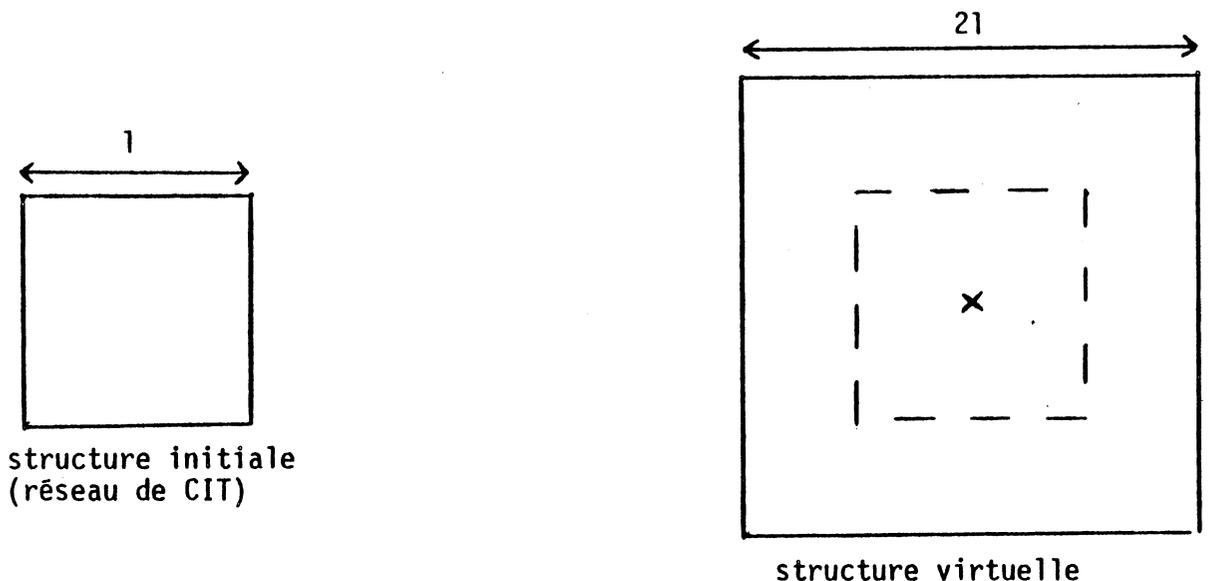


Figure 36

Cette solution a été choisie au départ grâce aux facilités qu'elle présente pour passer d'une structure à l'autre. En effet, le passage de la structure initiale à la structure virtuelle nécessite juste la multiplication par deux

de certains paramètres dans les données du programme. Le passage dans le sens inverse qui consiste à immerger sur la structure hôte initiale, le placement trouvé sur la surface virtuelle se fait simplement en divisant certains paramètres par 2 et en appliquant au résultat trouvé quelques translations dans les deux directions (horizontale + verticale). Nous essayerons d'explicitier ceci dans la suite de ce paragraphe.

Cette solution malgré sa simplicité a permis au placement virtuel d'atteindre facilement ses objectifs.

La structure hôte virtuelle étant déterminée, la méthode enchaîne en lui affectant un par un tous les éléments du circuit selon l'ordre total décrit dans le paragraphe précédent "préstructuration logique".

Au commencement le noyau est affecté au centre de cette surface virtuelle. A partir de là, nous plaçons de façon optimale les éléments du circuit en respectant une évolution vers la figure géométrique de la structure hôte réelle. Pour cela, suivant que l'élément est une cellule logique ou une cellule périphérique, l'algorithme utilise une des deux procédures suivantes : "placer-élément-logique" et "placer-élément -périphérique".

#### 3.2.2.1. Procédure placer-élément-logique

Cette procédure a pour but d'affecter une place de la structure hôte virtuelle à l'élément logique sélectionné au cours d'une certaine étape de l'ordonnancement. Elle comprend les modules suivants :

## 3.2.2.1.1. Choix de l'ensemble des places candidates

On définit l'ensemble des places de la surface virtuelle auxquelles peut-être affecté l'élément d'une étape donnée, en considérant les emplacements encadrant l'ensemble déjà placé (figure 37) et respectant la figure géométrique de la structure hôte réelle (longueur d'une zone active, nombre de zones actives). On n'essaie donc pas n'importe quelle place demeurant libre. Ceci permet de restreindre la complexité de la méthode tout en l'orientant vers la minimisation de la longueur des connexions.

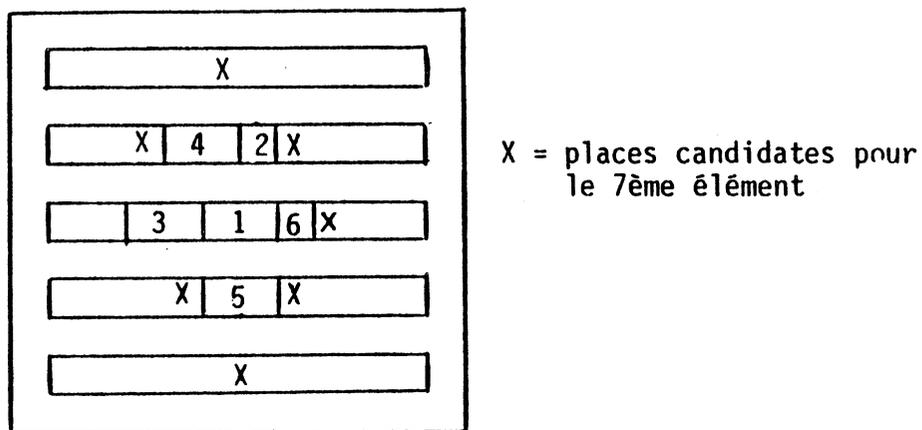


Figure 37

### 3.2.2.1.2. Discrimination de l'ensemble des places candidates

On réduit cet ensemble de places candidates selon des prédictions faites sur les accroissements des densités d'occupation (canaux et "feed-throughs") provoqués par l'affectation fictive de ce nouvel élément à une des places de cet ensemble. Le but de cette discrimination est la recherche de la faisabilité ultérieure d'un tracé à 100 %. Elle est faite selon le critère suivant :

#### Saturation des "canaux et passages"

Une place demeure candidate pour l'élément de l'étape "i" si toute connexion la liant avec un élément déjà placé peut : passer par n'importe quelle région d'un canal se trouvant à l'intérieur du rectangle englobant cette connexion sans provoquer un dépassement de capacité de celui-ci, et sans rencontrer une zone active de ce rectangle où trop de passages sont occupés. Notons qu'à ce stade lorsqu'un élément est lié à n éléments déjà placés par une même équipotentielle, on le considère comme connecté à l'élément le plus proche de la place candidate essayée. Cette hypothèse qui permet de restreindre la complexité de la méthode doit être prise en compte lors de la phase du routage ("global router") (cf. Chapitre IV) sans quoi les prédictions qui vont suivre n'auraient plus de sens.

L'évaluation d'une place candidate et donc la densité dans les canaux d'interconnexion et la probabilité d'utiliser un passage est effectuée comme suit :

Supposons qu'une place de coordonnées (a,b) est une place candidate pour un élément donné. Si cet élément est connecté (au sens de l'hypothèse ci-dessus)

à un élément déjà affecté à une place de coordonnées  $(c,d)$ , les densités de toutes les lignes de coupe verticales traversant le rectangle délimité par les quatre places de coordonnées  $(a,b)$ ,  $(c,b)$ ,  $(a,d)$ ,  $(c,d)$  (fig 38) seront supposées incrémentées de 1 dans chaque canal d'interconnexion. Les passages non saturés de chaque zone active se trouvant à l'intérieur de ce rectangle verront leur facteur d'occupation, supposé plus grand de  $1/K$  ; où  $K$  est le nombre de passages non encore saturés sur cette zone active. On considère qu'un passage est saturé lorsque son facteur d'occupation tend vers 1.

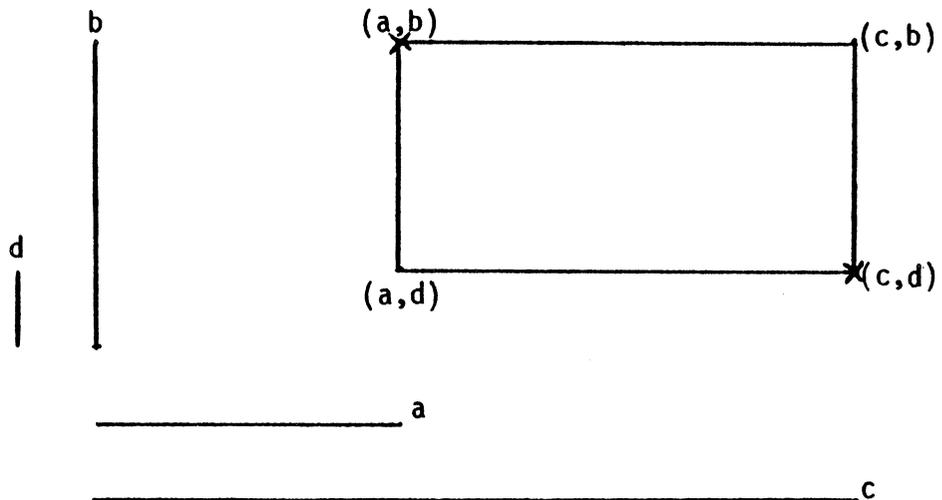


Figure 38

Rectangle englobant les 2 places  $(a,b)$  et  $(c,d)$

Une place candidate au III.2.2.1.1. sera rejetée ici si l'évaluation prédictive de la densité dans les canaux proposée ci-dessus dépasse en un point d'un canal quelconque un nombre fixé qui est la capacité de ce canal. Cette place sera aussi écartée s'il existe une rangée de cellules à l'intérieur du rectangle délimitant la connexion où trop de passages sont saturés.

### Remarques

1- Afin de permettre à cette discrimination d'être efficace (en termes de densité) dès les premières étapes du placement, nous avons choisi de modifier automatiquement la capacité des canaux, après le traitement d'un certain pourcentage des éléments du circuit. Par exemple, on l'initialise à 4 et on lui rajoute 2 chaque fois qu'on a traité 25 % des éléments du circuit. Ceci permettra aussi d'uniformiser la densité des connexions sur l'ensemble du réseau hôte.

2- Dans certains cas, on rejette toutes les places possibles au III.2.2.1.1. pendant la phase de discrimination. On les reprendra toutes et on choisira parmi elles la moins mauvaise au sens du vecteur coût qui sera définie dans la procédure suivante : "choix de la meilleure place". On résoudra au cours de la phase d'amélioration itérative, les problèmes que pourraient soulever cette stratégie.

#### 3.2.2.1.3. Choix de la meilleure place

Pour choisir la meilleure place parmi l'ensemble des places sélectionnées dans 3.2.2.1.2, nous avons défini un vecteur coût composé de 3 critères. Dans notre cas d'étude, les circuits à implanter étant difficiles à tracer, nous avons choisi l'ordre de priorité suivant pour ces différents critères.

##### 1 - Critères de densité des connexions

On estime les densités de connexions sur le circuit en définissant des lignes de coupes verticales (toutes les cellules de base) qui traversent tout le réseau

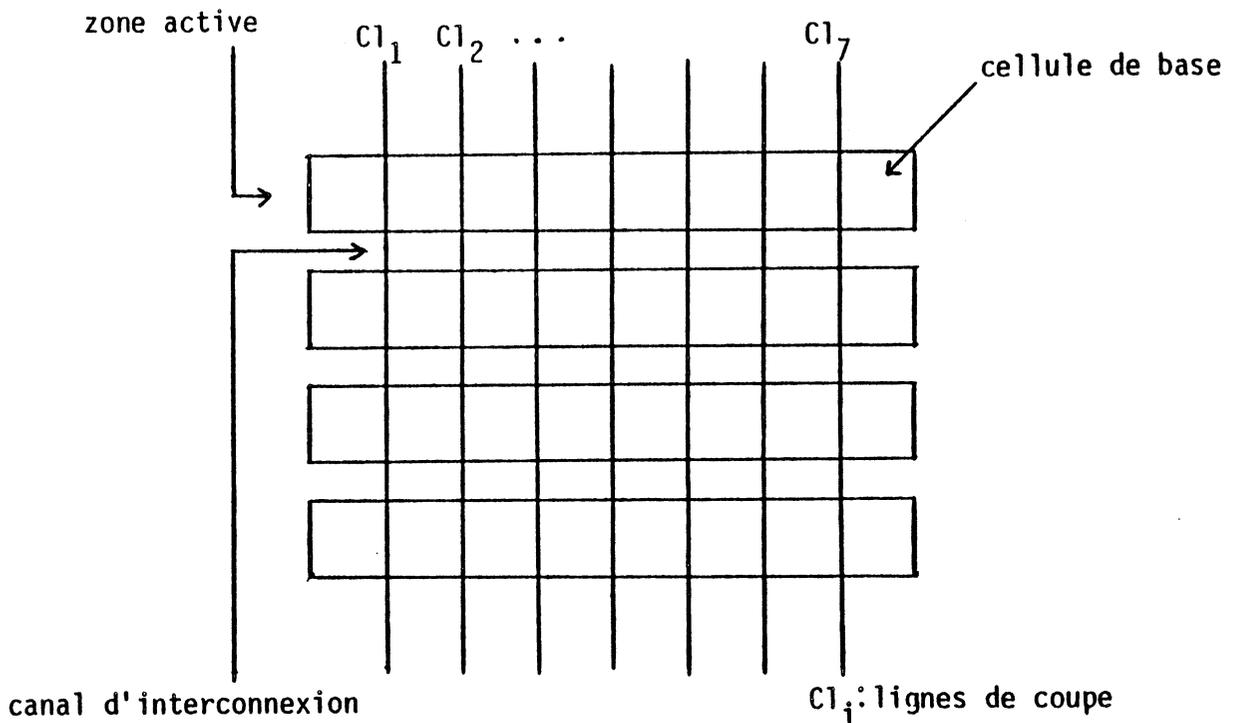


Figure 39

Une valeur utile de densité est ensuite affectée dans chaque canal à chaque ligne de coupe. Cette valeur est donnée par la relation suivante :

$$V_{ij} = d_{ij} - cap_i$$

où  $d_{ij}$  représente le nombre de connexions qui traversent la ligne de coupe  $j$  dans le canal  $i$  et  $cap_i$  désigne la capacité du canal  $i$ .

Il est facile de voir que les  $v_{ij}$  prennent leur valeur dans  $[-12; +\infty[$ .

Remarquons que la procédure "discrimination de l'ensemble des places candidates" (III.2.2.1.2) est introduite pour jouer, pendant toute la phase du placement, le rôle d'une contrainte sévère afin de garder toutes les valeurs  $v_{ij}$  dans  $[-12; 0]$ . Cette contrainte est condition minimale de faisabilité du tracé mais elle peut dans certains cas être insuffisante. C'est pour cela que nous allons

évaluer pour chaque place candidate : la valeur maximale  $v_{ij}$  enregistrée sur tout le réseau et provoquée par l'affectation fictive de l'élément de l'étape courante à cette place. La minimisation de cette valeur dès les premières étapes du placement et jusqu'à la fin de celui-ci permettra de minimiser la densité des connexions sur l'ensemble des canaux du réseau hôte. Ainsi, à chaque étape la place retenue réalise la plus petite valeur de  $\max_{i,j}(v_{ij})$ . En cas de conflit où plusieurs places réalisent ce minimum, nous garderons celle pour laquelle  $\max_{i,j}(v_{ij})$  a été atteint un plus petit nombre de fois.

## 2 - Critère lié aux passages au dessus des zones actives

Ce critère est évalué par une quantité booléenne qui indique pour chaque place restante s'il y a suffisamment ou non de passages non saturés au-dessus des zones actives pour connecter l'ensemble des éléments déjà placés à l'unique élément de cette étape supposé affecté à cette place (3.2.2.1.2).

### Remarque

Les circuits proposés correspondant en moyenne à un remplissage à 70 %, la méthode crée périodiquement une cellule de base fictive (réservée aux passages) sur chaque zone active chaque fois qu'on a rempli dessus un certain nombre de cellules de base. C'est pour cela que nous n'avons pas estimé les passages plus finement comme ci-dessus à travers des lignes de coupes horizontales définies toutes les zones actives.

### 3 - Critère de longueur des connexions

Il est basé sur les deux considérations suivantes :

a) La longueur d'une connexion biterminales (j) est estimée par la mesure (Lj) du demi-périmètre du rectangle qui englobe ses points de contact.

b) Si l'élément de l'étape "i" est relié par une équipotentielle à plusieurs modules déjà placés, on le supposera comme connecté à l'élément de cette équipotentielle occupant la place la plus proche de celle où on essaie l'élément de l'étape "i".

Par suite, si l'élément de l'étape "i" possède n connexions avec les (i-1) déjà placés la quantité  $\sum_{i=1}^j L_j$  est le critère de longueur des connexions à minimiser.

#### 3.2.2.1.4. Mise à jour des ressources

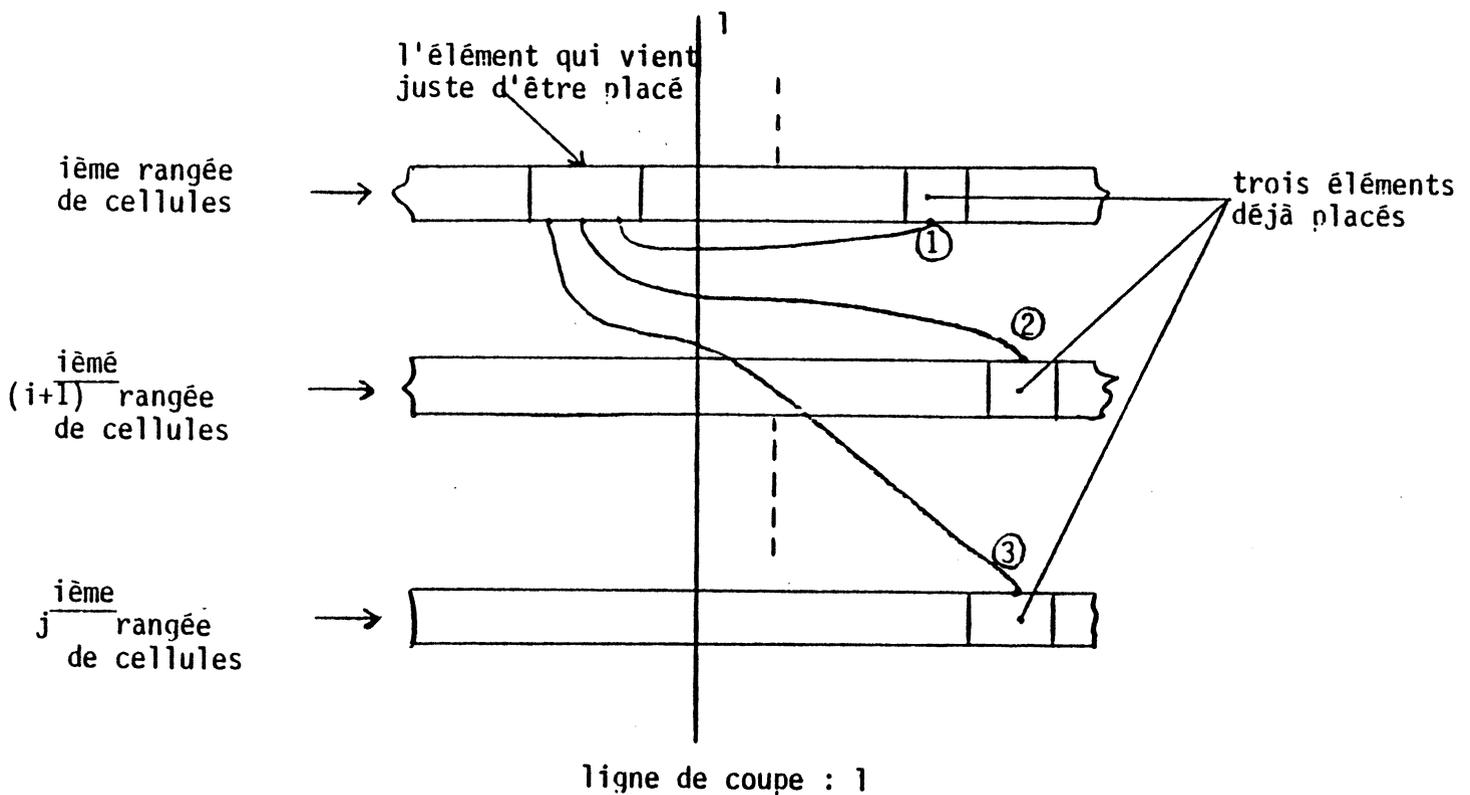
Quand la meilleure place optimisant le vecteur coût défini au 3.2.2.1.3. est sélectionnée et donc affectée définitivement à l'élément de l'étape "i", une mise à jour de la densité et des passages est faite comme suit :

##### a) Mise à jour du facteur d'occupation de chaque passage

Au paragraphe 3.2.2.1.2, on a donné une évaluation du facteur d'occupation de chaque passage quand on place l'élément "i" à une place candidate. Ces valeurs ont contribué au choix de la meilleure place. Une fois la meilleure place choisie, on met à jour le facteur d'occupation des passages avec les valeurs correspondantes à cette meilleure place seulement.

b) Mise à jour de la densité

Les valeurs de densité  $d_{ij}$  et  $v_{ij}$  définies au 3.2.2.1.3. et associées aux lignes de coupes verticales relativement à chaque canal d'interconnexions sont mises à jour selon l'illustration présentée au schéma suivant [SHI] :



- connexion ① : on ajoute  $1/2$  à la composante (correspondante au canal  $[i, i + 1]$ ) du vecteur de densité associé à la ligne de coupe :  $l$  puisque le routage utilisera le canal  $[i, i + 1]$  ou le canal  $[i - 1, i]$
- connexion ② :  $1$ , à la composante associée au canal  $[i, i + 1]$ , puisque le routage utilisera le canal  $[i, i + 1]$
- connexion ③ :  $1/(j - i)$ , aux composantes associées aux  $(j - i)$  puisque le routage empruntera tous ces canaux avec la même probabilité.

Figure 40

Ainsi se termine la procédure qui est utilisée itérativement par le programme pour placer toutes les cellules logiques du circuit à implanter. Nous passons maintenant à celle qui traite les cellules périphériques.

### 3.2.2.2. Procédure placer-élément-périphérique

A son tour cette procédure a pour but d'affecter une place périphérique de la structure hôte virtuelle à la cellule périphérique "E" sélectionnée au cours d'une certaine étape de l'ordonnancement. Elle procède comme suit :  
 Considérons l'ensemble CT des éléments de coordonnées  $(X_i, Y_i)$ , déjà placés et reliés à E. Ces éléments peuvent être logiques ou périphériques.  
 On appelle médian de l'élément E, un pseudo-barycentre  $B_E$  des coordonnées des éléments de CT dont les coordonnées sont calculés ainsi :

$$X_{B_E} = \frac{|CT|}{\sum_{i=1}^{|CT|} \lambda_i X_i} \quad , \quad Y_{B_E} = \frac{|CT|}{\sum_{i=1}^{|CT|} \lambda_i Y_i}$$

où  $\lambda_i$  est un coefficient de pondération traduisant le nombre d'équipotentiels différentes qui lient les deux éléments  $X_i$  et E.

E est alors affecté à une place périphérique qui est encore libre et qui est la plus proche de ce médian.

#### Remarque

On peut considérer comme places candidates du placement de la cellule périphérique E, les E places périphériques disponibles les plus proches du médian (E étant un paramètre du programme) et choisir parmi elles celle qui optimise les critères liés aux passages et à la densité des connexions.

Le placement sur une structure hôte virtuelle s'achève après l'affectation sur cette surface de la dernière cellule périphérique. Nous passons maintenant à la dernière phase de ce placement initial constructif qui est "le placement final sur la structure hôte réelle".

### 3.2.3. Placement final sur la structure hôte réelle

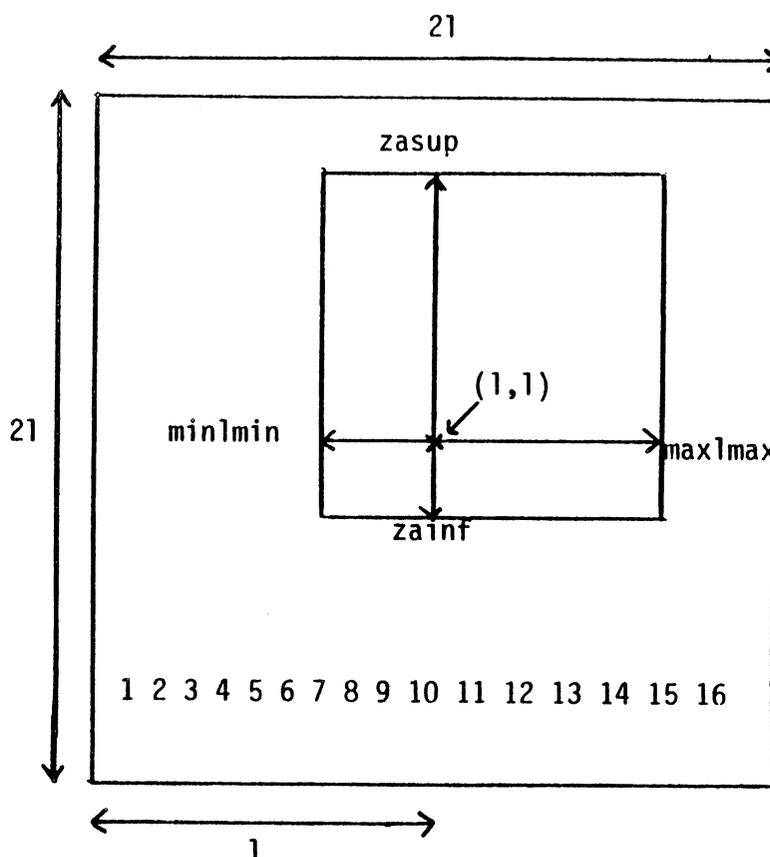
Pour ramener le placement trouvé sur la vraie structure hôte, nous utilisons les valeurs des 4 paramètres suivants, enregistrées au cours du placement virtuel :

minlmin : désigne le numéro de la cellule de base la plus à gauche remplie au cours du placement virtuel (figure 41)

maxlmax : numéro de la cellule de base la plus à droite remplie au cours du placement virtuel

zainf : désigne le numéro de la zone active (logique) la plus basse, utilisée au cours du placement virtuel

zasup : numéro de la zone active (logique) la plus haute utilisée au cours du placement virtuel.



l : largeur de la structure hôte réelle

Figure 41

Chaque élément logique affecté à la place de coordonnées  $(x,y)$  sur la structure virtuelle sera affecté sur la structure réelle à la place de coordonnées  $(x',y')$  définie comme suit :

si  $zasup > l$  alors  $y' = y - (zasup - l)$  sinon  $y' = y$

si  $maxlmax > l$  alors  $x' = x - (maxlmax - l)$  sinon  $x' = x$

Pour une cellule périphérique l'affectation sur la structure hôte réelle ne modifiera pas la zone sur laquelle elle aurait été affectée au cours du placement virtuel (elle restera donc la zone du : haut, bas, droite ou gauche). Par contre le numéro de sa place sur cette zone sera divisé par 2 car au cours du placement virtuel on aurait décidé de placer toutes les cellules périphériques sur des places de numéro pair.

Ainsi s'achève le placement initial constructif.

### III - 3. Amélioration itérative

Au cours de la phase précédente i.e. placement initial constructif nous avons utilisé une prédiction assez forte de connectique dans le but de diminuer l'impact de cette phase d'amélioration itérative. L'amélioration itérative ne sera ici indispensable que pour supprimer des points de saturation éventuels afin d'assurer la traçabilité. Elle pourra aussi dans certains cas être effectuée pour réduire encore plus la longueur totale des connexions. Le placement initial peut toutefois dans beaucoup de cas donner directement un placement acceptable.

A l'heure actuelle, une amélioration simple effectuant les échanges par paires de cellules de taille équivalente pour réduire la longueur des connexions a été développée chez CIT-ALCATEL. Elle donne des résultats qui semblent rester insuffisant pour les circuits très denses.

Une autre méthode [SHI 80] qui consiste à échanger les éléments ou déplacer et insérer des cellules à l'intérieur de chaque rangée de cellules pourrait être suffisante. Cette méthode a été développée avec succès à un réseau hôte semblable à celui proposé pour cette étude pour atteindre le même objectif à savoir l'élimination des endroits de congestion de densité. Cette méthode s'est avérée plus efficace que les méthodes conventionnelles car les cellules bougent avec une perspective d'amélioration réaliste.

## CONCLUSION

D'autres stratégies de placement constructif ont été développées pour des structures en technologie CMOS (une ou deux couches d'interconnexions en AL) semblables à celles proposées pour cette étude. Par exemple une méthode par bipartitionnement successifs a été décrite dans [SHI 80], une méthode constructive par front mobile a été aussi développée dans le Laboratoire "LCS" de Grenoble. On appelle front la délimitation physique entre la partie gauche du réseau hôte où on a déjà placé des cellules et la partie droite qu'on remplira dans les étapes suivantes... Il est difficile de comparer ces méthodes d'une manière absolue. Les paramètres de chaque réseau à implanter (facteur de remplissage, nombre de connexions, rôle privilégié de certaines cellules...) contribuent à ce qu'une approche soit meilleure dans un cas précis et moins bonne dans un autre.

Cependant la méthode présentée en III possède de nombreux avantages par exemple :

- . La méthode crée dynamiquement des cellules logiques vides. De plus elle place aussi les cellules périphériques dans le cas où leurs places ne sont pas imposées.

- . La méthode utilise une forte prédiction de connectique ce qui diminue l'impact d'une phase d'amélioration itérative.

- . La méthode a une appréhension globale du problème. La préstructuration logique + le pré-placement sur une structure hôte virtuelle permettront de guider l'optimisation locale, sur le sous-réseau partiel correspondant à une étape donnée de cette méthode ascendante vers une optimisation globale sur le réseau final.

Les résultats obtenus sur la base des essais effectués (présentés au paragraphe V) confortent nos affirmations. En effet, l'idée de base étant

de réduire au maximum la nécessité d'une amélioration ultérieure et de faciliter le travail du router ; nous pensons qu'on pourra encore gagner dans ce sens si on peut apporter à cette méthode les améliorations suivantes :

1 - En ce qui concerne la prédiction de la connectique

- Le caractère prédictif de la méthode proposée se trouverait d'autant plus efficace qu'on pourrait mémoriser l'éclatement des équipotentielles en biterminals. En effet, ceci faciliterait de façon substantielle le travail ultérieure du traceur. De plus cette façon de procéder serait compatibles avec les exigences du traceur.

- Une prise en compte des points exacts d'entrée/sortie sur chaque cellule permettrait d'affiner la prédiction des ressources de connectique.

2 - A propos de l'appréhension globale du problème nous pouvons proposer deux approches complémentaires

- L'analyse factorielle des données [BAR 84a] : elle apporte une autre façon de déterminer de façon globale et économique un placement initial. L'avantage d'obtenir ce placement très rapidement est freiné par le critère de faisabilité du tracé. Il reste donc à voir comment tirer profit de cet apport vis à vis de la méthode prédictive utilisée.

- Les méthodes de classification automatique dont nous citerons quelques applications dans la troisième partie de cette thèse permettent de traiter des exemples dans lesquels il y a beaucoup plus de blocs. Elles essayent de regrouper de façon homogène et d'ordonner des cellules sur la base d'un critère global d'agrégation. Cela permet aussi de réduire la complexité du problème à traiter, dans la mesure où le problème initial se trouve décomposé en un nombre réduit de petits problèmes relatifs aux classes obtenues.

#### IV - LE TRACE DE CONNEXIONS [MAR 84]

##### IV - 1. Introduction

Le tracé de connexions sur réseaux prédiffusés, en général, se heurte à un problème capital : la capacité fixe des canaux d'interconnexion.

Alors que dans les problèmes généraux ("full custom"...), les algorithmes vont tendre à réduire le nombre de pistes employées dans chaque canal, en vue d'une compaction ultérieure du circuit, afin de gagner en surface ; sur réseaux prédiffusés, la surface étant fixée, le but est de tracer toutes les connexions dans la limite impartie, en minimisant la longueur des connexions.

Dans le cas de réseau à une seule couche de métallisation s'ajoute le problème posé par l'emploi de barette en Silicium Polycristallin. Celui-ci étant très capacitif et resistif, on va essayer de minimiser son emploi. Le nombre de contact entre la couche métal et la couche Si poly est fixé, de même que le nombre de "feed through" (passage au travers des cellules). Ces contraintes supplémentaires vont imposer une gestion sévère des ressources critiques.

## IV - 2. Divers types de stratégies

Deux grandes familles de stratégies s'opposent à ce niveau.

### Première famille

Une première famille où on va ranger les algorithmes réalisant un tracé fin immédiat sur l'ensemble du réseau. Ce sont essentiellement les algorithmes de type Lee ou de tracé par ligne.

### Algorithmes de Lee

Ils nécessitent une grande place mémoire (bit map de l'ensemble du réseau) et un temps d'exécution très long. La recherche des chemins s'exécutant séquentiellement, l'ordre de traitement va être très important. De plus la taille des réseaux allant croissant, ce type de solution devient de moins en moins applicable. Les meilleurs résultats sont obtenus sur des structures de petites tailles à faibles contraintes. Ils restent néanmoins très souvent appliqués localement pour résoudre des points critiques.

### Recherche par ligne - Expension par ligne

Mieux adaptés à la structure par canaux, que l'algorithme de Lee, dont ils dérivent, ils sont généralement plus rapide et restent assez généraux. Cependant ils restent encore assez lent, et n'obtiennent pas de bons résultats dans les cas à fortes contraintes.

### Deuxième famille

La seconde famille est représentée par les algorithmes de types "channel router" et "global router".

Le tracé s'effectue en 2 phases :

- 1 phase d'affectation des équipotentiels (ou morceaux d'équipotentiels) aux canaux : tracé global ("global ou topological router").
- 1 phase de tracé fin des équipotentiels dans les canaux ("channel router").

Il existe de nombreux algorithmes de type channel router mais les plus couramment développés le sont pour des circuits à la demande et généralement pour deux couches de métallisation.

Les avantages sont importants.

#### La phase d'affectation (routage global)

Elle permet de se rendre compte très rapidement des cas d'impossibilité de tracé pour cause de manque de ressources. Dans ce cas on peut revenir à une remise en cause du placement. Cette phase s'effectue généralement en 2 parties :

- une affectation séquentielle
- une phase d'amélioration qui va résoudre les problèmes locaux de surcharge et qui va tendre à uniformiser la densité dans les canaux.

#### Le channel router

On peut choisir soit des algorithmes généraux, comme ceux de rechercher par lignes, soit beaucoup plus adaptés aux structures. Plus le réseau aura des contraintes fortes et moins les algorithmes généraux seront efficaces.

Il conviendra donc pour une application particulière de développer des outils collant de très près à la structure, en prenant le risque de perdre une certaine transportabilité.

### Choix de la stratégie

L'emploi avec un taux de réussite acceptable de l'une ou l'autre classe de stratégie va dépendre en fait de la qualité du placement et de la quantité d'information qu'il peut fournir.

Avec un placement à faible taux de prédiction de densité dans les canaux, une stratégie avec routage global puis par canal s'imposera. Dans le cas de placements à forts taux de prédictions avec amélioration du placement jusqu'à obtenir une uniformisation de la densité des connexions.

La phase de routage global ne s'imposera plus et on pourra envisager un tracé fin immédiat sur l'ensemble du circuit. Mais il faut garder en tête le fait que les méthodes classiques restent assez lourdes. Si on veut obtenir un résultat acceptable dans un temps raisonnable, il faudra peut être adapter l'outil pour le faire coller à la structure et perdre de sa généralité.

### Solution retenue

Pour l'application décrite après, nous avons retenu une stratégie avec global et channel router.

Le tracé global restant assez général, le tracé fin collant de très près au type de réseaux employés en vue d'un maximum d'efficacité.

Les Dogleg channel router donnent de bons résultats mais ils sont malheureusement inapplicables ici, à cause du peu de liberté, que présentent les réseaux à une couche de métallisation, au niveau des changements de couche d'interconnexions.

On a donc développé un algorithme nouveau sur les prédiffusés une couche, à savoir une adaptation de "greedy-channel-router" au problème actuel.

#### IV - 3. Description de la méthode utilisée

##### Le "global router"

C'est la phase d'application des équipotentielles aux canaux. On va chercher à uniformiser la densité dans les canaux de façon à rendre le circuit entièrement traçable.

Il se décompose en deux parties :

- une partie "constructive" dans laquelle les équipotentielles vont être traitées séquentiellement,
- une partie d'amélioration qui va remettre en cause certaines affectation de la première phase pour cause de saturation.

## 1 - Phase constructive

Cette phase se décompose en quatre modules qui vont s'enchaîner séquentiellement.

### 1. Module 1 : éclatement en biterminales

Le traitement des équipotentielles multiterminales apportant une complication au problème, on va dans ce premier module, éclater chaque connexion multiterminale en un ensemble de biterminales qui respectent un arbre de connexion minimum (figure 42). Les cellules sont reliées 2 à 2 suivant les plus courtes distances calculées en longueur de Manhattan.

Ces distances étant pondérées, pour tenir compte de la recherche de minimisation d'emploi de Sipoly, et du nombre restreint de passage à travers les cellules.

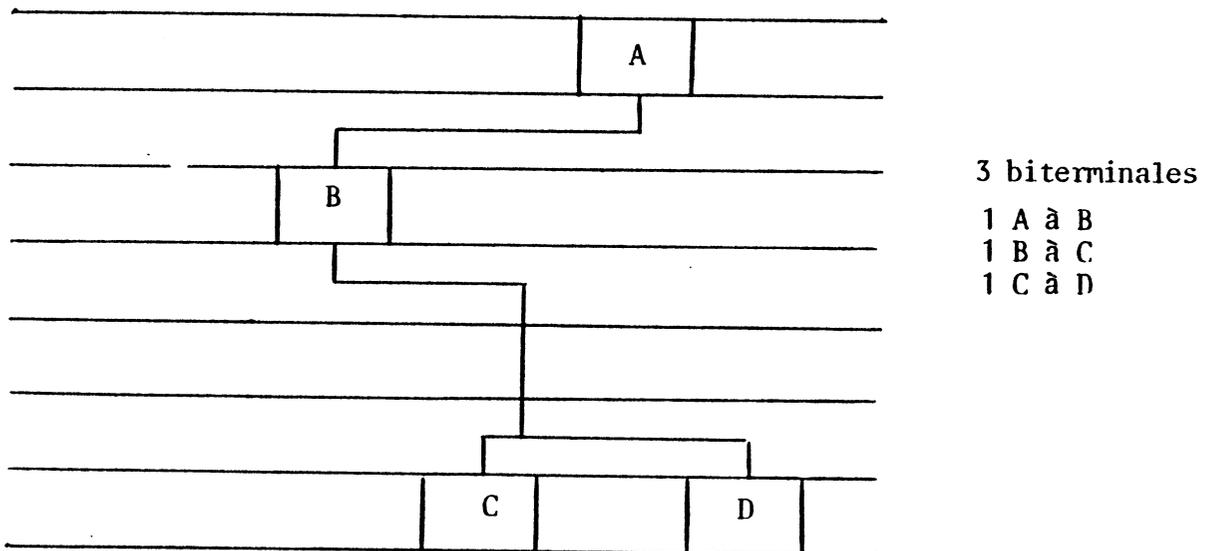


Figure 42

Afin de minimiser l'emploi de Sipoly la composante verticale de la distance de Manhattan va être multipliée par un coefficient de façon à privilégier les pistes horizontales en aluminium. Ce coefficient est en rapport avec le taux de remplissage du réseau et du nombre de passage disponible à travers les cellules. Ce coefficient constitue un paramètre du programme, et peut être modulé par l'utilisateur (ou fixé pour un type de réseau donné, après une série d'expérimentation).

L'éclatement en biterminale se fait en conservant une information sur l'équipotentielle "mère" de façon à pouvoir concaténer deux biterminales "soeur" empruntant la même portion de canal lors du tracé fin (tendance à se rapprocher d'un arbre de Steiner).

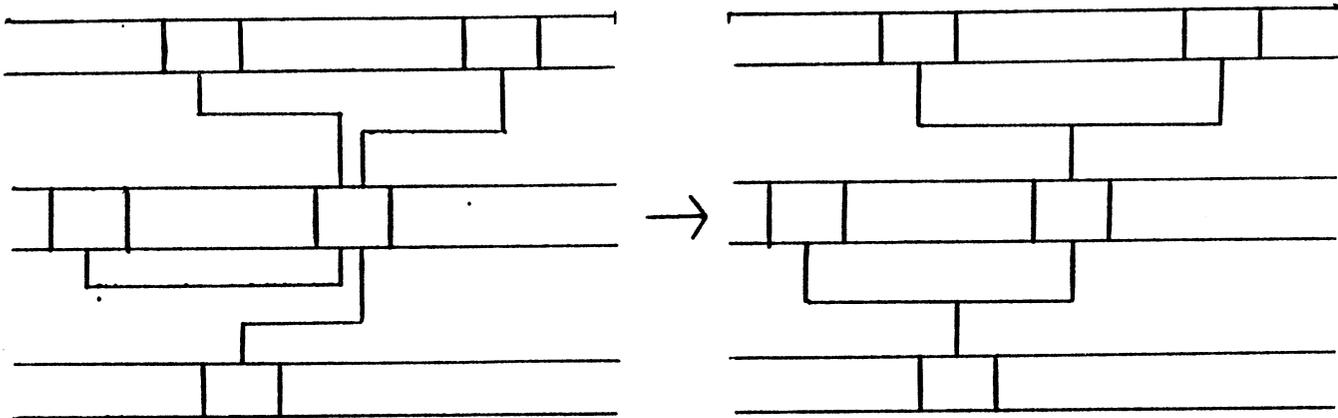


Figure 43

## 2. Module 2 : traitement des biterminales simples

### a) Coût d'un chemin

Le coût d'un chemin reliant deux cellules se mesure en fonction de plusieurs paramètres. Il peut être représenté par un vecteur à trois composantes :

$$C = \begin{pmatrix} C_1 \\ C_2 \\ C_3 \end{pmatrix} \quad \begin{array}{l} C_1 : \text{composante booléenne de traçabilité} \\ C_2 : \text{composante densité dans les canaux empruntés par le} \\ \quad \text{chemin} \\ C_3 : \text{longueur du chemin} \end{array}$$

Dans la phase de routage global on ne tiendra pas compte de la composante  $C_3$  celle-ci ne nous servant qu'au moment du fractionnement des équipotentiels en biterminales. Tant que le tracé reste à l'intérieur du rectangle englobant les cellules à connecter la longueur de Manhattan des biterminales reste la même.

On part de l'hypothèse que le placement a cherché à minimiser les longueurs d-s équipotentiels, on va donc chercher à minimiser uniquement la densité dans les canaux afin d'assurer une traçabilité complète.

La densité est mesurée suivant des lignes de coupe perpendiculaires aux canaux. On dispose une ligne de coupe entre chaque cellules de bases (fig. 44).

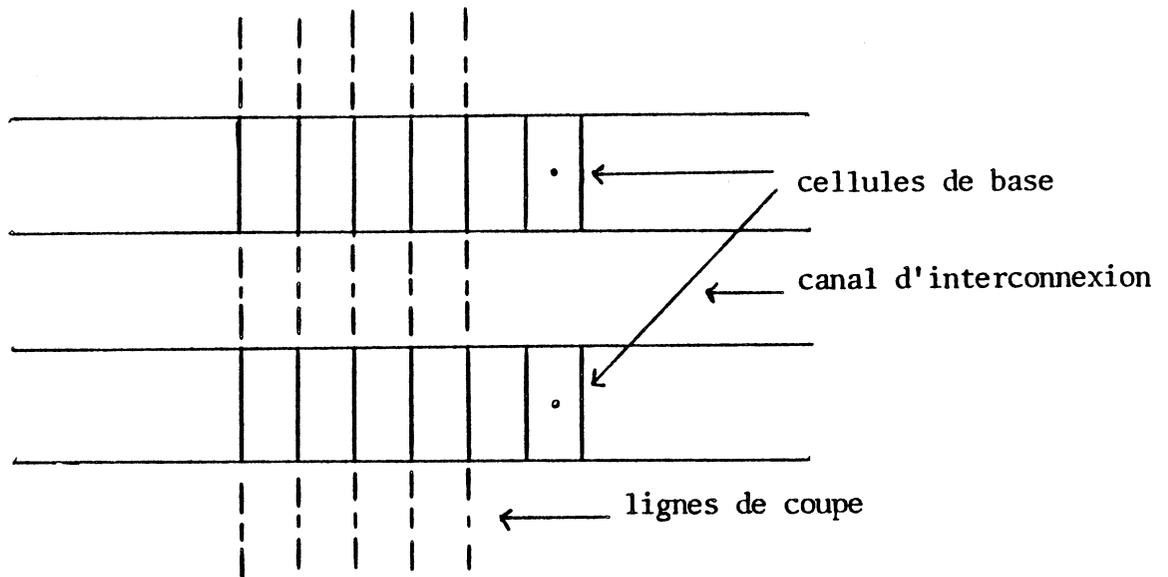


Figure 44

La densité dans les canaux est mesurée sur chaque ligne de coupe. Le coût se réduit donc à la composante  $C_2$  et à la composante  $C_1$  de traçabilité (disponibilité des feed through).

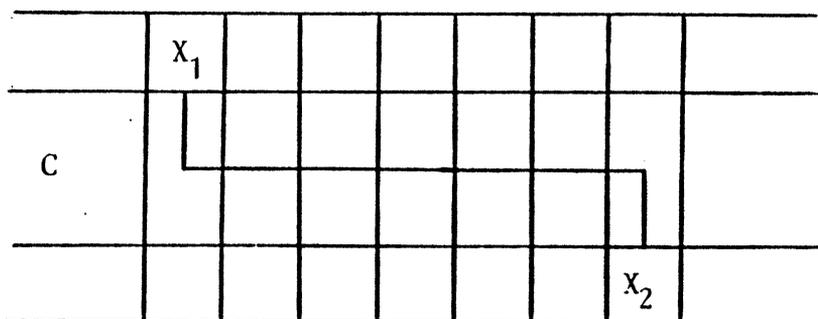
Pour tenir compte des pics de densité le coût d'un chemin est calculé par la somme des carrés des densités.

#### b) Traitement des biterminales simples

Est appelée simple une biterminale qui se trace entièrement dans un canal.

2 cas

b1. Les deux extrémités sont sur deux zones actives différentes.



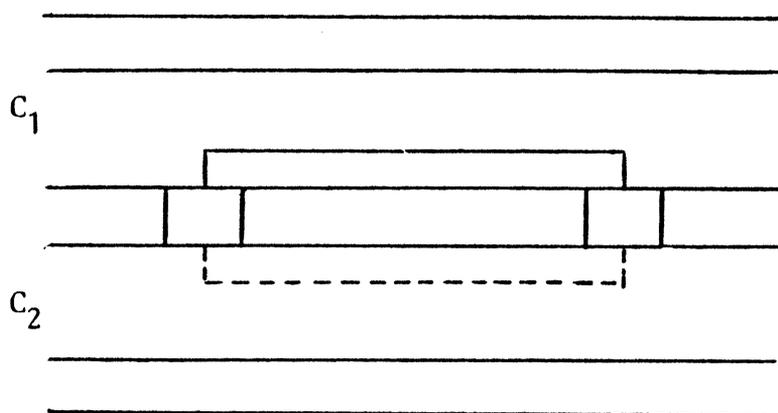
La biterminale est affectée au canal situé entre les deux zones actives

Figure 45

La matrice de densité est augmentée de un entre  $X_1$  et  $X_{2-1}$  pour le canal C. On affecte la biterminale à cette portion de canal même si l'augmentation de densité conduit à un dépassement de capacité du canal.

Les seules ressources prises en compte pour la traçabilité sont les feeds through (inutile ici) et les barettes de Sipoly pour les canaux limités à une seule rangée de barettes.

b2. Les extrémités sont sur la même zone active



2 possibilités

Figure 46



Une matrice d'occupation des passages est donc tenue à jour. Le chemin de coût minimum est calculé en employant l'algorithme de Dijkstra de recherche du plus court chemin dans un graphe à arcs valués. S'il n'y a pas de solution possible par manque de ressource dans le rectangle englobant on augmente le rectangle de recherche jusqu'à trouver une solution ou une impossibilité totale.

L'augmentation du rectangle se faisant uniquement sur les X de façon à ce que les augmentations de longueur n'aient lieu qu'en aluminium.

#### 4. Module 4 : module d'amélioration

Les deux modules précédents trouvent la meilleure solution à un instant donné. Cette solution pouvant être remise en cause par les biterminales tracées après.

L'uniformisation de la densité et la décongestion des zones saturées va donc être réalisée au cours d'un module d'amélioration.

Plusieurs procédures simples sont employées.

##### 1 - Déplacement sans accroissement de longueur

###### a) biterminales simples

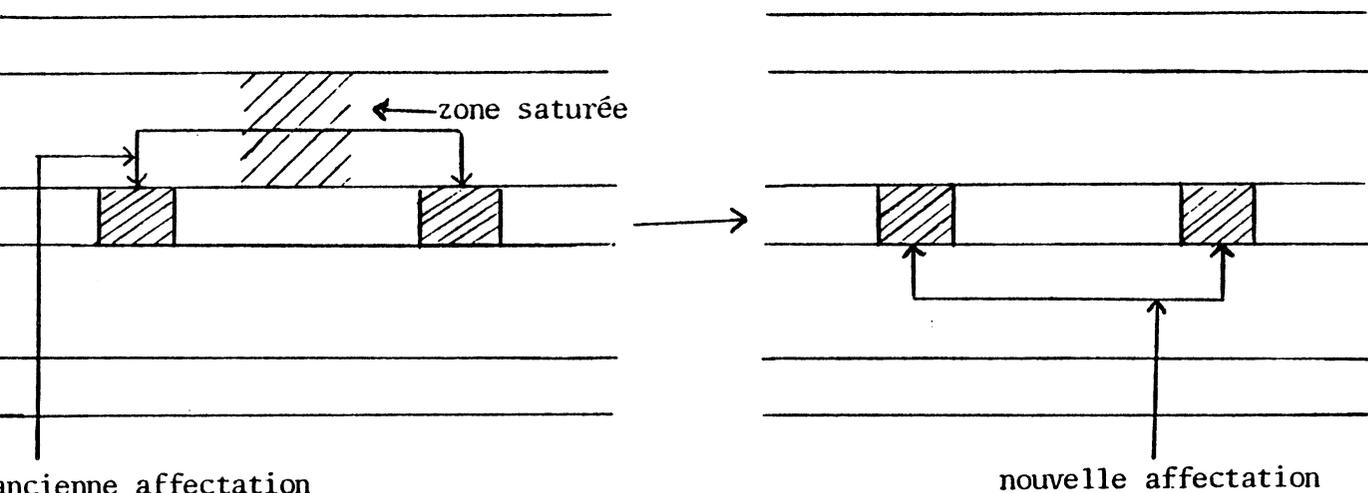


Figure 48

Une biterminale simple empruntant une portion de canal saturée est basculée dans le canal supérieur (ou inférieur suivant le cas) si le coût dans celui-ci est moindre que le coût dans le canal courant.

### b) biterminales complexes

On va rechercher une solution qui évite la portion de canal saturée par l'emploi de feed through en deça ou au delà de la saturation.

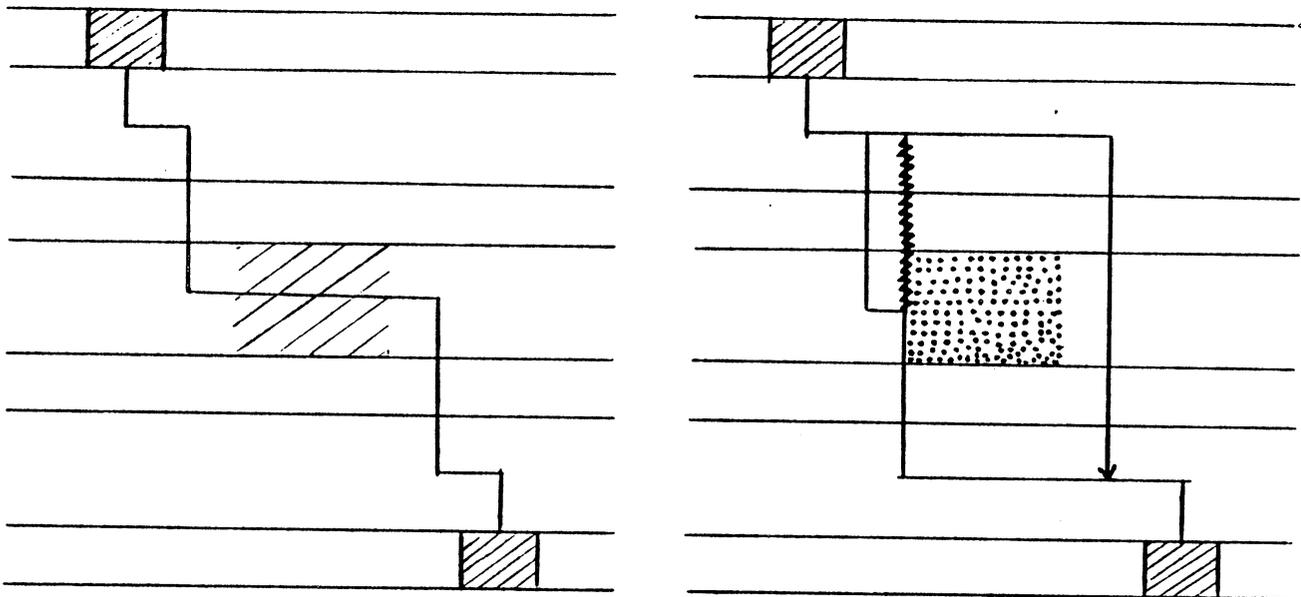


Figure 49

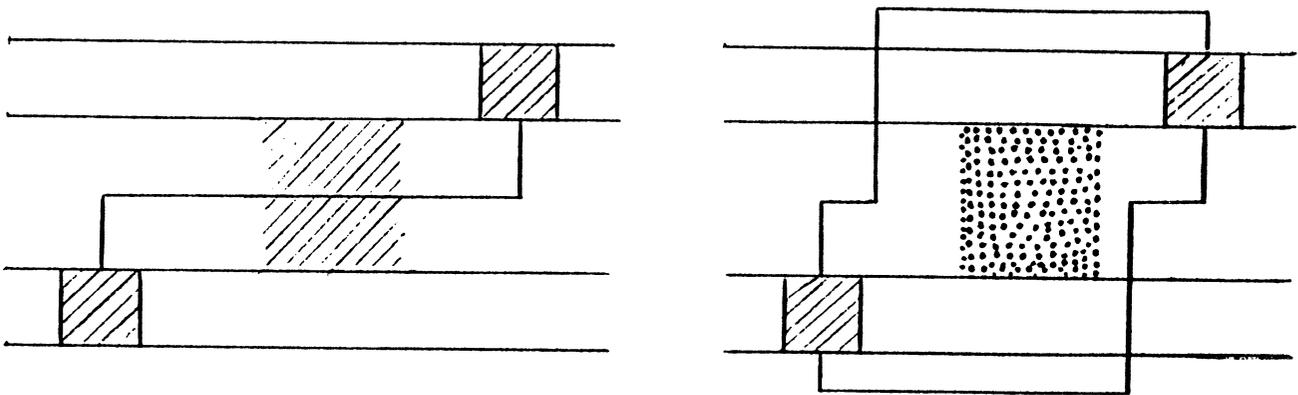
Il peut exister plusieurs solutions possibles, afin d'éviter d'en recalculer un trop grand nombre on conserve celle qui "colle" la plus à la zone saturée pour ne pas déplacer une trop grande longueur de biterminale.

### 2 - Déplacements avec modification de longueur

Dans le cas où les procédures ci-dessus échouent on va essayer de décongestionner les zones saturées en permettant un accroissement minime de la longueur des connexions.

Les accroissements se faisant soit en aluminium soit par l'emploi de feed through supplémentaires.

a) biterminales simples



2 solutions possibles. On conserve celle de cout minimum.

Figure 50

b) biterminales complexes

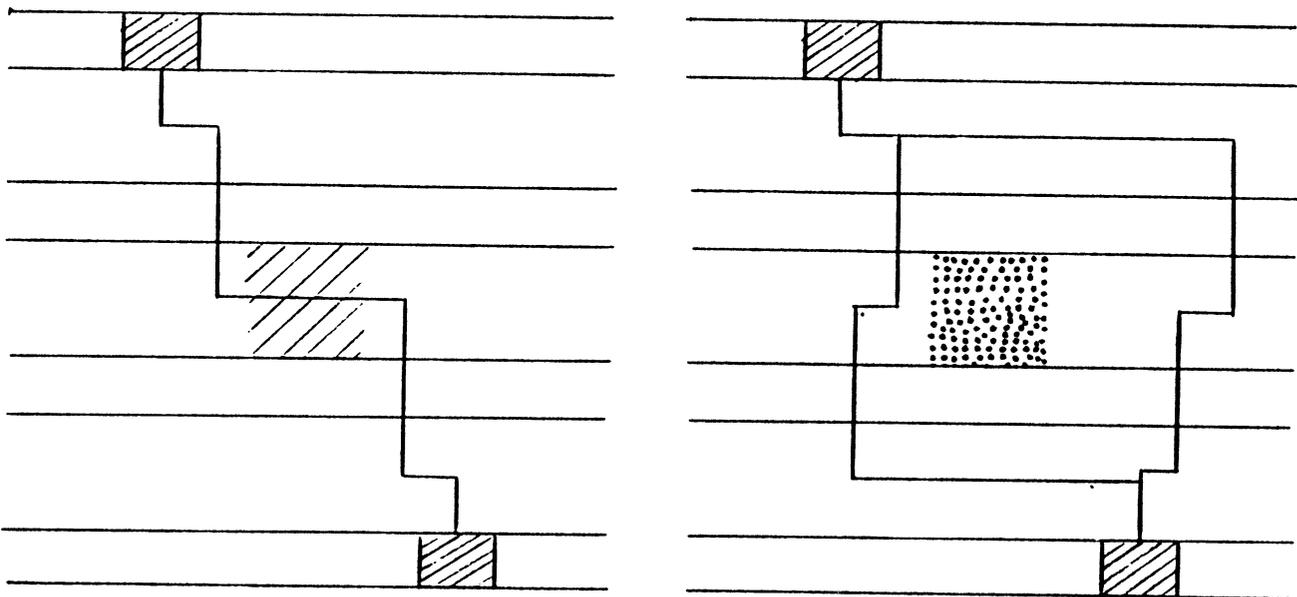


Figure 51

L'accroissement se fait par débordement du rectangle englobant pour la recherche de feed through.

Plusieurs solutions, celle qui nécessite la moins grande augmentation est conservée.

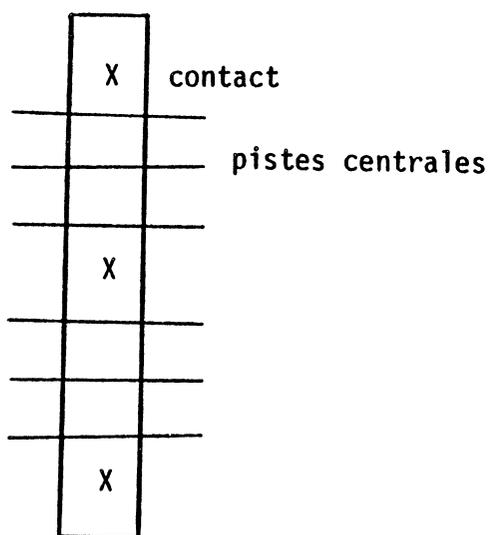
### 3 - Effacé-retracé de biterminales complexes

Lorsque les procédures précédentes ont échouées, on n'essaie plus des solutions locales mais on efface et retrace à l'aide du module 3 les biterminales empruntant la zone de canal congestionné. Dans ce cas on interdira au chemin trouvé de passer par un point de saturation s'il n'y a pas de solution possible on remplace la biterminale sur son ancien chemin et on en essaie une autre.

#### Remarque

Au delà de la simple décongestion des canaux, on va essayer de diminuer la densité en dessous de la capacité du canal pour plusieurs raisons.

- Cas de pistes centrales



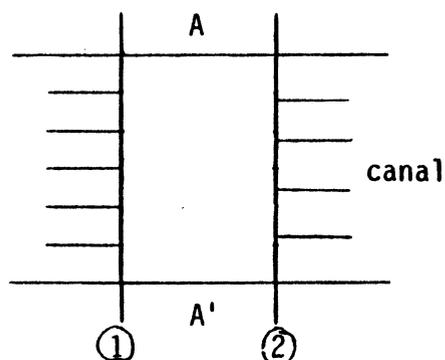
- Lorsque les barettes de Si poly se présentent sous cette forme, on remarque que pour entrer ou sortir sur une piste centrale, il est nécessaire qu'une des deux pistes encadrantes soit libre.

Figure 52

Localement on dispose donc d'une capacité plus faible lors d'une connexion à une cellule.

- Limites liées à la stratégie

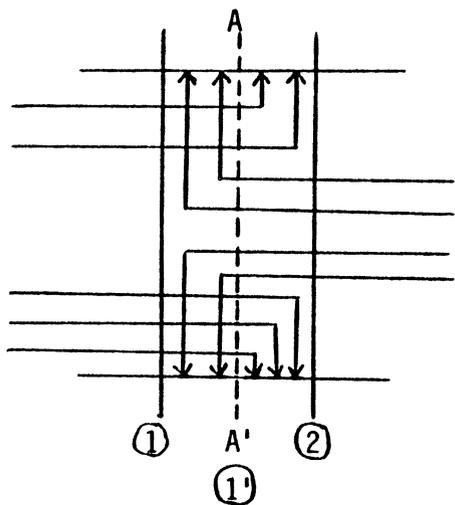
Afin d'éviter de faire un bit-map total du circuit, les lignes de coupes sont placées au niveau de chaque cellule de bases et non à chaque pas de grille. Les densités correspondent donc à un flux entrant et un flux sortant au niveau d'une cellule de base.



Une densité de 5 sur la ligne 1 et de 4 sur la ligne 2 ne veulent pas forcément dire qu'une biterminale s'arrête sur A ou A' et qu'aucune n'en part.

Figure 53

On ne sait pas ce qui se passe entre deux lignes de coupe et une vision pessimiste pourrait donner ceci.



Ce qui implique qu'en ① on aurait besoin de neuf pistes pour tout tracer.

Figure 54

Donc on a une sous évaluation de la densité vraie. On aura par conséquent intérêt à descendre la densité dans les canaux, tout en gardant à l'esprit que certaines procédures rallongent le chemin.

### Le "Channel router"

La stratégie Greedy channel router développée par Rivest et Fiduccia [RIV 82] adaptée au problème présent, à savoir, capacité fixe des canaux, et changements de pistes limites dans leur facilité. L'algorithme se déroule en  $n$  pas si  $n$  est le nombre de pas de grille du réseau.

A chaque pas  $i$ , on va retenir ce qui existait au pas  $i-1$  et prendre en compte les nouvelles entrées et sorties du canal.

Au pas  $i$  :

- réaliser les sorties
- réaliser les entrées
- prolonger au pas  $i+1$  les biterminales non finies en  $i$ .

Ce schéma simple doit être complexifié pour éviter les conflits à un même pas deux prises de contact d'équipotentiels différentes ne peuvent avoir lieu sur la même barette pour cause de court-circuit.

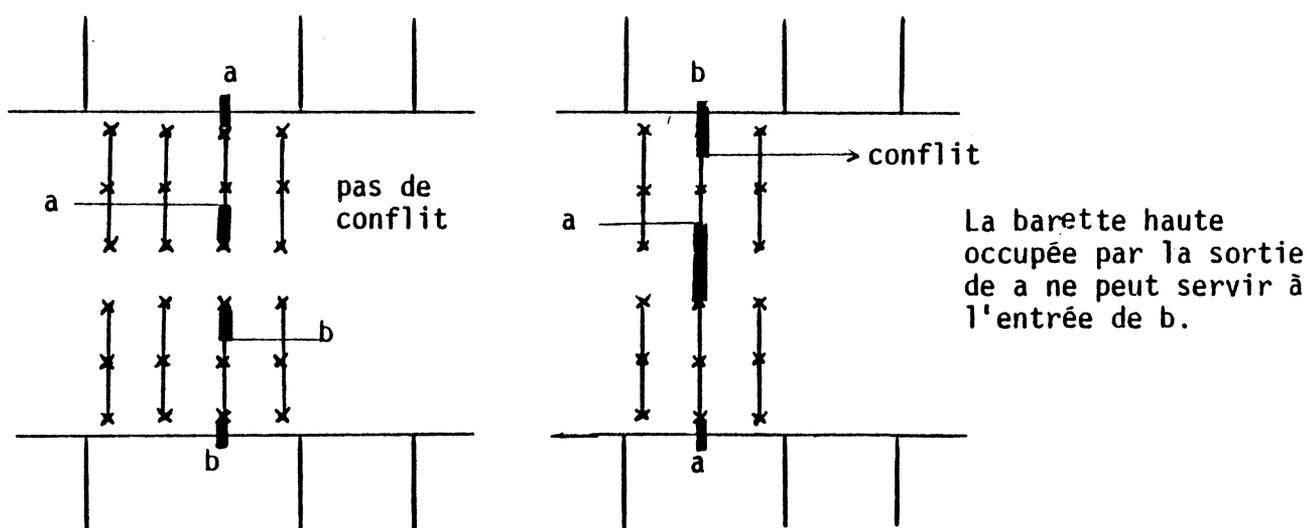


Figure 55

Il est donc nécessaire qu'une biterminale se situe sur la portion de canal qui jouxte sa sortie, afin de ne pas gêner un mouvement sur la barette opposée.

Lors d'une entrée, si la sortie de la biterminale se situe sur la barette du 1/2 canal opposée, on se trouve devant deux cas :

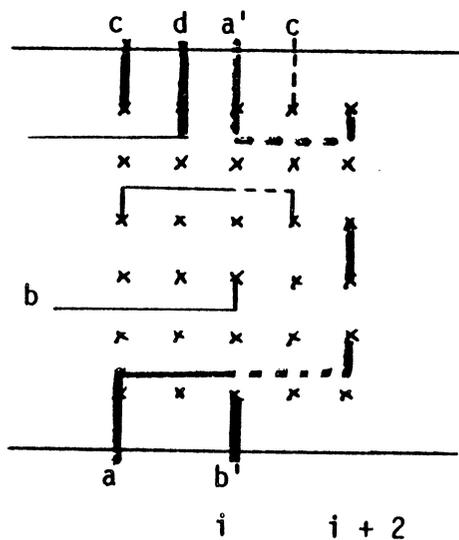
- soit la barette opposée est libre dans ce cas l'entrée se fera directement sur la barette opposée,
- soit elle est occupée pour une autre sortie ou une autre entrée dans ce cas l'entrée se fera sur la barette adjacente et la biterminale sera placée dans une liste gérant les demandes de changement de 1/2 canal.

Une phase supplémentaire a donc lieu à chaque pas  $i$ , dans le cas où il n'y a ni entrée ni sortie à ce pas.

Dans les cas où on traite des canaux à "pistes centrales" une dernière phase s'exécute pour faciliter les sorties des pistes centrales. Les biterminales empruntant ces pistes sont ramenées sur des pistes latérales dès que possible vu l'encombrement du canal.

Pour chacune des quatre phases de tracé, des procédures de back tracking interviennent pour résoudre les problèmes d'impossibilité qui pourraient apparaître.

Dans le cas des sorties, si aucune solution ne peut être trouvée même pas un back tracking, on crée une biterminale "éclatée" qui sera refermée dès que possible.



La biterminale entrée en a ne trouve pas dans son 1/2 canal de sortie et ne peut y aller pour raison de court-circuit avec la biterminale bb' si elle prend la barette basse.

On va donc créer un autre bout de biterminale au niveau de la sortie a' et on va refermer cette biterminale dès que possible.  
(au pas  $i + 2$  ici)

Figure 56

A chaque pas on exécute donc :

- réaliser les sorties
- réaliser les entrées
- concaténer les biterminales "éclatées"
- effectuer les changements de 1/2 canal
- effectuer les changements de pistes.

## V - PERFORMANCES

Les deux programmes (PLACMOS + TRACMOS) du système IACMOS ont été développés par le LCS sur le CII-HB 68 de l'IMAG.

Ils représentent 12 000 lignes de programmes en PASCAL et utilisent 300 Koctets de mémoire (dont 47 Koctets pour PLACMOS et 355 Koctets pour TRACMOS).

Ce système a été implanté en PASCAL sur VAX (11/780) dans le systèmes CAO de la Société "CIT-ALCATEL" de Paris où il est actuellement opérationnel.

Le tableau suivant représente les résultats trouvés lors de l'implantation de 5 circuits à la CIT par le systèmes IACMOS.

Circuit	1	2	3	4	5
Nombre d'éléments	174	378	358	384	434
Nombre de cellules logique	147	339	297	344	376
Nombre de plots	27	39	61	40	58
Nombre de cellules de base	317	639	677	802	722
Remplissage	28,3 %	57,0 %	60,4 %	71,6 %	64,4 %
Nombre de feed-troughs créés	47	150	154	142	162
Surface employée (feed-through)	32,5 %	70,4 %	74,2 %	84,3 %	79,3 %
Nombre d'équipotentielles	233	470	506	559	645
Nombre de biterminales	353	850	741	1065	997
Nombre de segments	498	1262	1192	1658	1624
Nombre de segments non tracés	0	10	11	25	43
Pourcentage de segments tracés	100 %	99,2 %	99,0 %	98,4 %	97,3 %
Nombre de passages utilisés (théorique)	145 (145)	412 (412)	451 (451)	593 (593)	627 (627)
Longueur de métal utilisé (théorique)	18658 (18318)	53848 (52636)	55256 (53032)	72400 (69760)	72822 (67952)
Nombre de barettes utilisées	1214	3267	3078	4377	4246
CPU (VAX II/780) PLACMOS		9'20"	6'26"	9'47"	12'29"
CPU (VAX II/780) TRACMOS		3'55"	25'58"	15'33"	18'06"
CPU (VAX II/780) IACMOS		13'15"	32'24"	25'20"	30'35"

Figure 57

### Interprétation

La qualité des résultats varie en opposition avec le coefficient de remplissage du circuit. Mais celui-ci n'est pas le seul en cause. Le nombre d'équipotentiels joue un rôle important à même taux de remplissage. L'efficacité diminue si le nombre d'équipotentiels augmente.

### Remarque

Le nombre de Feed-throughs créé pour réaliser un tracé représente environ 15 % de la surface totale employée, ce qui indique qu'un taux de remplissage de 80 % est la limite à ne pas dépasser si on veut avoir la place physique pour réaliser l'implantation.

Pour une qualité de résultats acceptable la borne de 75 % de remplissage représente sûrement une valeur plus juste.

De plus, cette méthode trouve sûrement des limites au voisinage de circuits de cette taille (1200 portes). Pour des circuits plus importants une division du problème en sous-problèmes s'impose pour continuer à utiliser ce type de méthodes.

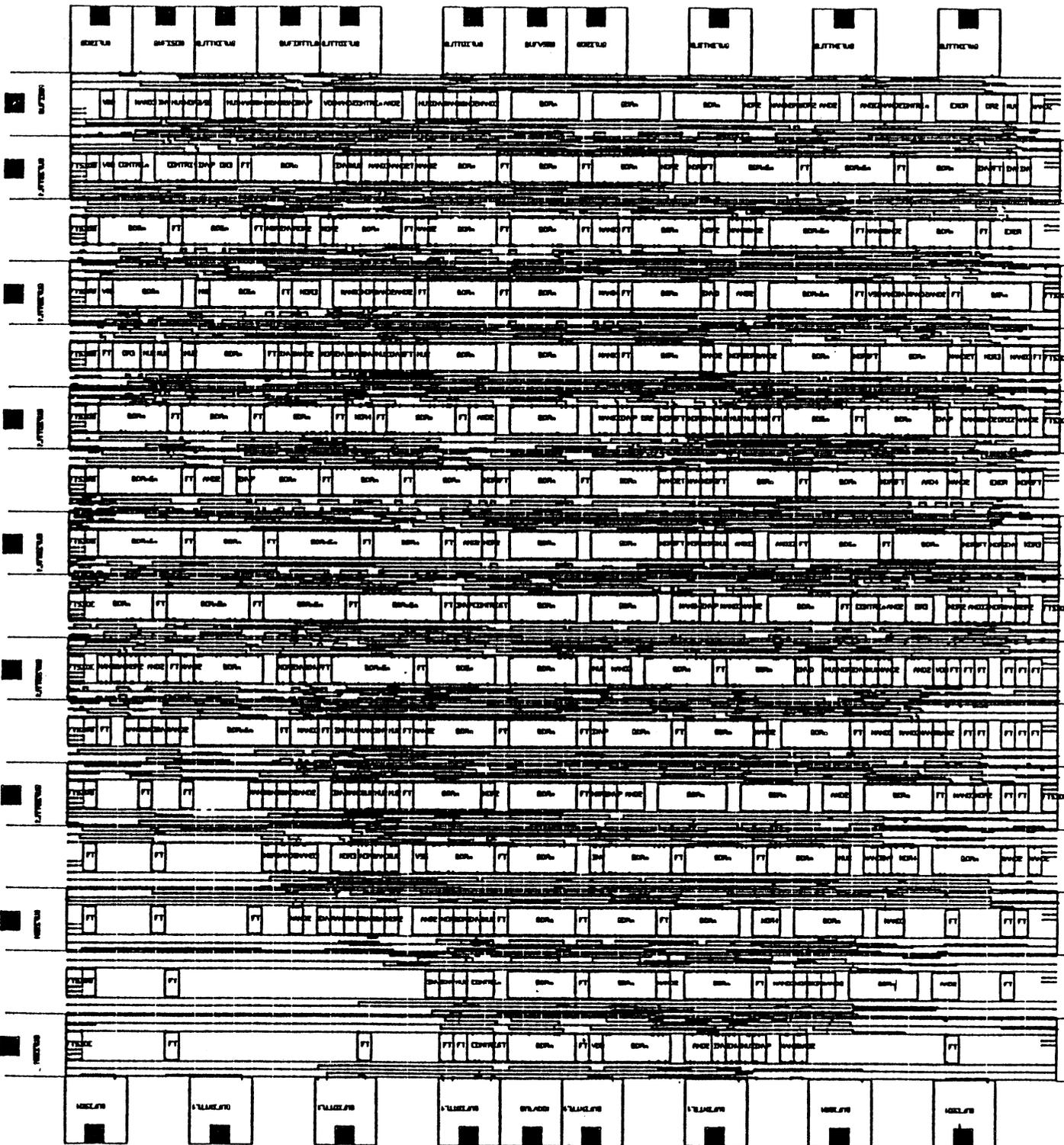


Figure 58

Implantation du circuit 4 sur réseau UCI 1200 par le système IACMOS

PARTIE III

VERS L'UTILISATION DES METHODES DE CLASSIFICATION



## I - LE PROBLEME A RESOUDRE

Nous avons déjà évoqué au cours des deux premières parties de cette thèse, l'intérêt de recourir aux deux grandes familles des méthodes statistiques multidimensionnelles (ou techniques d'analyse de données) : les méthodes factorielles et les méthodes de classification automatique pour résoudre un problème de placement [BAR 85].

Dans cette partie nous présentons une étude qui va dans ce sens. Il s'agit de remplacer un problème de placement complexe dans lequel il y a un grand nombre de modules à placer par des sous problèmes plus petits et donc plus faciles à traiter.

Un problème de placement de modules d'un circuit peut, à certains égards, se décrire par le tableau suivant :

- le tableau de description logique (Net-list) croissant les modules et les équipotentiels :

$$X_{ij} = 1 \quad \text{si l'équipotentielle numérotée } j \text{ rencontre le module numéroté } i$$

$$X_{ij} = 0 \quad \text{sinon}$$

Cette étude, partant de ce tableau de description logique est donc destinée à :

- a) Produire des regroupements de ces modules en blocs de modules (classes) : c'est l'objectif des méthodes de classification automatique.
- b) Résoudre le problème pour chacun des petits problèmes relatifs aux classes obtenues.

## II - RAPPEL SUR LES METHODES DE CLASSIFICATION AUTOMATIQUE

Pour atteindre l'objectif 1.a (mise en évidence d'aggrégats possibles entre les modules) une méthode de classification automatique s'appuie essentiellement sur la définition d'une mesure de proximité (pseudo distance) entre les modules. On ne confond pas distance au sens mathématique du terme avec proximité au sens statistique (laquelle ne satisfait pas l'inégalité triangulaire) mais cet usage de proximité à la place de distance qui serait illicite en statistique exploratoire parce qu'il n'y a pas de modèle sous-jacent est rendu possible par la notion de modèle virtuel [BAR 84a] .

La définition de cette pseudo- distance est assez importante car elle oriente nettement les résultats. En effet, un agrégat (permettant de définir un sous problème) est formé de modules entre lesquels les pseudo distances sont faibles. Entre deux éléments appartenant à deux agrégats différents la pseudo distance sera assez grande. Plusieurs définitions (de cette pseudo distance) basées essentiellement sur des analogies et des intuitions, sont possibles.

Le choix de la meilleure définition est laissée au gré de l'utilisateur. L'usage seul peut le lui indiquer.

### Exemple de mesures de proximité

a) A partir du tableau X présenté ci-dessus dans "I" on peut définir la mesure de proximité suivante entre les modules i et j :

$$\Delta I = \sum_{k=1}^P (X_{ik} - X_{jk})^2 \quad \text{où } p \text{ désigne le nombre de colonnes de } X.$$

Dans ce cas particulier, cette définition rejoint celle de la distance euclidienne dans  $R^P$ .

b) On peut aussi définir la proximité  $\Delta_2$  comme suit :

Pour tout couple de modules  $i, j$ , on regarde  $n_{ij}$  le nombre d'équipotentiels communes. On pose :

$$\Delta_2(i,j) = \frac{1}{1 + n_{ij}}$$

Contrairement à la première, cette proximité ne définit pas une distance car elle ne vérifie pas l'inégalité triangulaire.

c) En utilisant le même tableau qu'en 2.a on peut aussi poser pour tout couple de module  $i, j$  :

$$\Delta_3(i,j) = \frac{1}{1 + S_{ij}} \quad \text{où } S_{ij} = \frac{\prod_k x_{ik} x_{jk}}{(\sum_k x_{ik})(\sum_k x_{jk})}$$

$S_{ij}$  est une sorte de corrélation entre les modules  $i$  et  $j$ .

d) A partir du même tableau  $X$  défini dans I, on peut construire le tableau  $Y = X^t X$  et choisir la mesure de proximité suivante entre deux module  $i, j$  :

$$\Delta_4(i,j) = \sum_k (Y_{ik} - Y_{jk})^2$$

Ceci traduit pour tout couple de modules  $(i,j)$  avec  $i \neq j$ , le nombre de modules distincts auxquels ils sont liés tous les deux.

Nous expérimenterons plus loin chacune de ces quatre pseudo distances sur la base d'une méthode de classification qui sera présentée au paragraphe suivant.

### III - STRATEGIE DE RESOLUTION

Pour résoudre le problème posé au 1.a nous proposons de procéder comme suit :

a) Appliquer une méthode de classification automatique pour définir les classes de modules (aggrégats).

b) Discuter la qualité du résultat obtenu par cette méthode en rapport avec le but escompté par l'implanteur. Pour cela nous proposons un procédé mathématique original qui aboutit à la réduction du problème initial en un nombre restreint de sous-problèmes.

En ce qui nous concerne, M. TRICOT [BAR 84b] nous a proposé une méthode de classification automatique. Nous indiquerons, dans ce qui suit, brièvement le mécanisme utilisé.

#### III - 1. Une méthode de classification automatique (selon M. TRICOT (LCS))

l'algorithme mathématique de cette méthode se décompose comme suit :

1 - On suppose donnée entre les modules une mesure de proximité  $\Delta$

2 - On définit la proximité entre deux classes  $C_i$  et  $C_j$  par :

$$D(C_i, C_j) = \min \{ \Delta(x, y) / x \in C_i, y \in C_j \}$$

On pose  $\alpha = \max_{i, j} \Delta(C_i, C_j)$

On calcule une estimation de  $\alpha$  en déterminant l'arbre minimal associé au graphe pondéré des modules [BAR 84b].

3 - Pour tout module  $x$  on note  $V_x = \{ y / \Delta(x, y) \leq \alpha \}$ . On établit une relation  $R$ , sur l'ensemble des modules, réflexive et symétrique :

$$xRy \iff \text{Min} \left\{ \frac{|V_x \cap V_y|}{|V_x|}, \frac{|V_x \cap V_y|}{|V_y|} \right\} \geq \beta$$

La relation  $R$  dépend donc de  $\beta : \beta \in [0,1]$ .

4 - La fermeture transitive de  $R$ ,  $R$ , forme une relation d'équivalence sur l'ensemble des modules dont les classes constituent la classification recherchée.

On aura donc :

$$xRy \Leftrightarrow \exists z_1, z_2, \dots, z_k \text{ tels que } xRz_1, z_1Rz_2, \dots, z_kRy$$

$R$  est une relation d'équivalence.

Remarquons que l'on aurait pu choisir un autre critère d'aggrégation pour  $R$ , par exemple :

$$\frac{|V_x \cap V_y|}{\sqrt{|V_x| \cdot |V_y|}}$$

Ceci est une sorte de corrélation. On prendra le premier pour l'étude qui suit.

En résumé, l'étude statistique présentée, la classification automatique est fonction arbitraire :

- de la mesure de proximité  $\Delta$
- du critère  $R$  d'aggrégation choisi comme relation entre les modules

On le voit, le critère est ici composé de :

- deux paramètres  $\alpha$  et  $\beta$
- une relation sur l'ensemble des modules.

### III - 2. Qualité du résultat obtenu et proposition d'une amélioration complémentaire

Nous avons déjà évoqué l'aspect exploratoire des méthodes de classification automatique. Celles-ci ne se prétendent nullement inférentielles et il sera souvent important (voir même indispensable) d'analyser la structure en classes du système en question que ces méthodes suggèrent. Pour cela nous proposons le procédé mathématique original suivant :

1 - On suppose trouvées  $n$  classes distinctes par la méthode de classification automatique choisie.

2 - On définit une pseudo corrélation entre deux classes  $C_i$  et  $C_j$  par :

$$\rho_{ij} = \frac{G_{ij}}{\sqrt{G_i \times G_j}} \quad \text{où}$$

a -  $G_{ij}$  désigne le nombre de "1" du tableau logique " $X_{ij}$ " croisant les modules de la classe  $C_i$  et ceux de la classe  $C_j$  et construit comme suit :

Soit  $\ell$  un module dans  $C_i$  et  $m$  un module dans  $C_j$ , on pose :

$X_{ij}(\ell m) = 1$  s'il existe une équipotentielle liant le module  $\ell$  au module  $m$ .

b -  $G_i$  désigne le nombre de "1" du tableau logique " $X_i$ " défini de la même façon que ci-dessus et croisant entre eux les modules de la classe  $C_i$ .

3 - On forme la matrice de pseudo corrélation  $\rho$  entre les différentes classes :

$$\rho = [\rho_{ij}]$$

$$1 \leq i \leq n$$

$$1 \leq j \leq n$$

4 - Dialogue avec l'implantateur.

L'implantateur peut s'appuyer sur cette matrice  $\rho$  pour juger de la qualité de représentation des résultats fournis par l'algorithme de classification initial. Certaines classes peuvent être à nouveau regroupées entre elles à la lumière de ce nouveau critère d'aggrégation. Il peut se mettre à jour un grand nombre de classes à petit effectif (par exemple les singletons) auquel cas l'implantateur serait amené à réaffecter ces dernières aux classes les plus importantes et les plus "proches d'elles".

5 - Procédé de résolution proprement dit

Pour atteindre ces deux objectifs nous proposons une méthode qui s'appuie sur l'analyse en composantes principales (ACP) d'un tableau de corrélation avec éléments supplémentaires [LEB 79].

Le tableau de corrélation utilisé est établi ici à partir du tableau  $\rho$  défini ci-dessus. Les axes factoriels définis par cette ACP ne sont fonction que des classes dites "actives" : les classes retenues par l'implantateur. Les autres jouant le rôle d'éléments supplémentaires sont projetées sur ce premier plan factoriel. Le but de ce procédé est d'affecter chaque élément supplémentaire à la classe proche à fort effectif dans le graphique obtenu. Dans ce même graphique on peut regrouper aussi deux classes à fort effectif (actives) qui sont très proches l'une de l'autre.

## 6 - Placement des classes obtenues

Une fois le regroupement des modules en classes, effectué suivant la méthode citée précédemment, étant jugé acceptable, il reste à résoudre le problème du placement à l'intérieur de chaque classe et le placement des classes entres-elles.

Nous proposons tout d'abord de déterminer l'emplacement de chaque classe par une méthode de placement de blocs "mous" (TSI 84) ; ensuite l'intérieur de chaque classe par :

- une méthode classique telle que celle décrite dans la deuxième partie de cette thèse
- par analyse factorielle si le nombre d'éléments de cette classe est inférieur à 100 [TSI 84]

#### IV - EXPERIMENTATION

Nous allons dans ce paragraphe présenter quelques résultats de l'expérimentation de la stratégie de résolution décrite dans 3 à une étude de cas consistant au circuit correspondant à la partie opérative du multiplieur de deux nombres de 4 bits présenté dans (A.3.2.1.4.1) et dont la description logique (description de la net-list) est donnée dans les deux pages qui vont suivre.

De plus pour voir ce que représentent les résultats trouvés en termes d'architecture de circuits, nous reprenons le schéma architectural qui a été conçu par l'utilisateur pour ce circuit ainsi que le schéma logique de cette partie opérative.

L'expérimentation consistera ici à appliquer plusieurs fois l'algorithme décrit dans 3.1. à cette étude de cas pour différentes mesures de proximité. Nous avons retenu pour chacune de ces mesures les seuils  $\alpha$  et  $\beta$  qui ont donné les meilleurs résultats lors de l'expérimentation. On fera intervenir pour chaque mesure de proximité la matrice de pseudo corrélation définie dans 3.2.3 qui permet de discuter la qualité des résultats trouvés. On appliquera le procédé de résolution décrit dans III.5 sur un de ces résultats à titre d'illustration et on donnera une interprétation architecturale à la solution correspondante proposée.

Fig 1. Description de la Net list du circuit traité

Fig. 1a

EQUIPOTENTIELLES		ELEMENTS				
1	1	63	55	51		
2	2	64	56	52		
3	3	65	57	53		
4	4	66	58	54		
5	1	2	3	4		
6	63	64	65	66		
7	5	6	7	8	9	
8	5	6	7	8	9	
9	59	60	61	62		
10	59	60	61	62		
11	10	11	12	13	14	55 56 57 58
12	10	11	12	13	14	55 56 57 58
13	51	52	53	54		
14	51	52	53	54		
15	5	11				
16	6	12	1	15	23	
17	7	13	2	16	24	
18	8	14	3	17	25	
19	9	55	4	18	26	
20	59	56	63			
21	60	57	64			
22	61	58	65			
23	62	66				
24	67	10	38	42		
25	11	43				
26	12	44				
27	13	45				
28	14	46				
29	51	15	19			
30	52	16	20			
31	53	17	21			
32	54	18	22			
33	15	19	23	47		
34	16	20	24	48		
35	17	21	25	49		
36	18	22	26	50		
37	19	27				
38	20	28				
39	21	29				
40	22	30				
41	23	27				
42	24	28				
43	25	29				
44	26	30				
45	27	31	35			
46	28	32	36			
47	29	33	37			
48	30	34	38			
49	35	31	39	47		
50	36	32	40	48		
51	37	33	41	49		
52	38	34	42	50		
53	31	43				
54	39	43				
55	32	44				
56	40	44				
57	33	45				
58	41	45				
59	34	46				
60	42	46				
61	47	10				
62	48	35	39			
63	50	37	41			
64	10	5				
65	11	6				
66	12	7				
67	13	8				
68	14	9				
69	55	59				
70	56	60				
71	57	61				
72	58	62				
73	5	6	7	8	9	
74	49	36	40			

fig. 1b

ELEMENTS	EQUIPOTENTIELLES				
1	1	5	16		
2	2	5	17		
3	3	5	18		
4	4	5	19		
5	7	8	15	64	73
6	7	8	16	65	73
7	7	8	17	66	73
8	7	8	18	67	73
9	7	8	19	68	73
10	11	12	24	61	64
11	11	12	15	25	65
12	11	12	16	26	66
13	11	12	17	27	67
14	11	12	18	28	68
15	29	33	16		
16	17	30	34		
17	18	31	35		
18	19	32	36		
19	29	33	37		
20	30	34	38		
21	31	35	39		
22	32	36	40		
23	16	33	41		
24	17	34	42		
25	18	35	43		
26	19	36	44		
27	37	41	45		
28	38	42	46		
29	39	43	47		
30	40	44	48		
31	45	49	53		
32	46	50	55		
33	47	51	57		
34	48	52	59		
35	45	49	62		
36	46	50	74		
37	47	51	63		
38	24	48	52		
39	49	54	62		
40	50	56	74		
41	51	58	63		
42	24	52	60		
43	25	53	54		
44	26	55	56		
45	27	57	58		
46	28	59	60		
47	33	49	61		
48	34	50	62		
49	35	51	74		
50	36	52	63		
51	1	13	14	29	
52	2	13	14	30	
53	3	13	14	31	
54	4	13	14	32	
55	1	11	12	19	69
56	2	11	12	20	70
57	3	11	12	21	71
58	4	11	12	22	72
59	9	10	20	69	
60	9	10	21	70	
61	9	10	22	71	
62	9	10	23	72	
63	1	6	20		
64	2	6	21		
65	3	6	22		
66	4	6	23		
67	24				

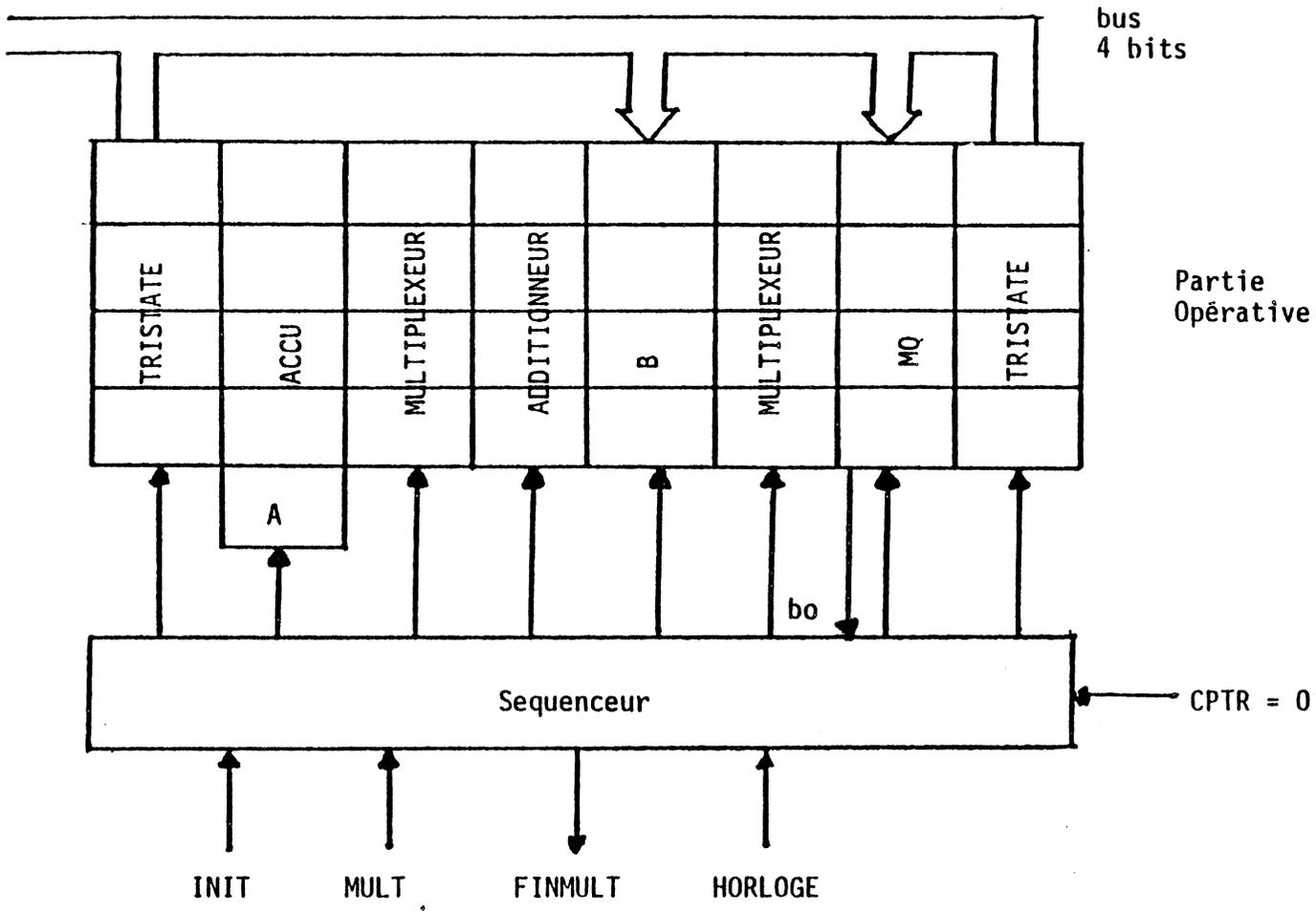


Fig. 2. Plan de masse du circuit traité

Fig. 3. Schéma logique de la partie opérative du circuit traité

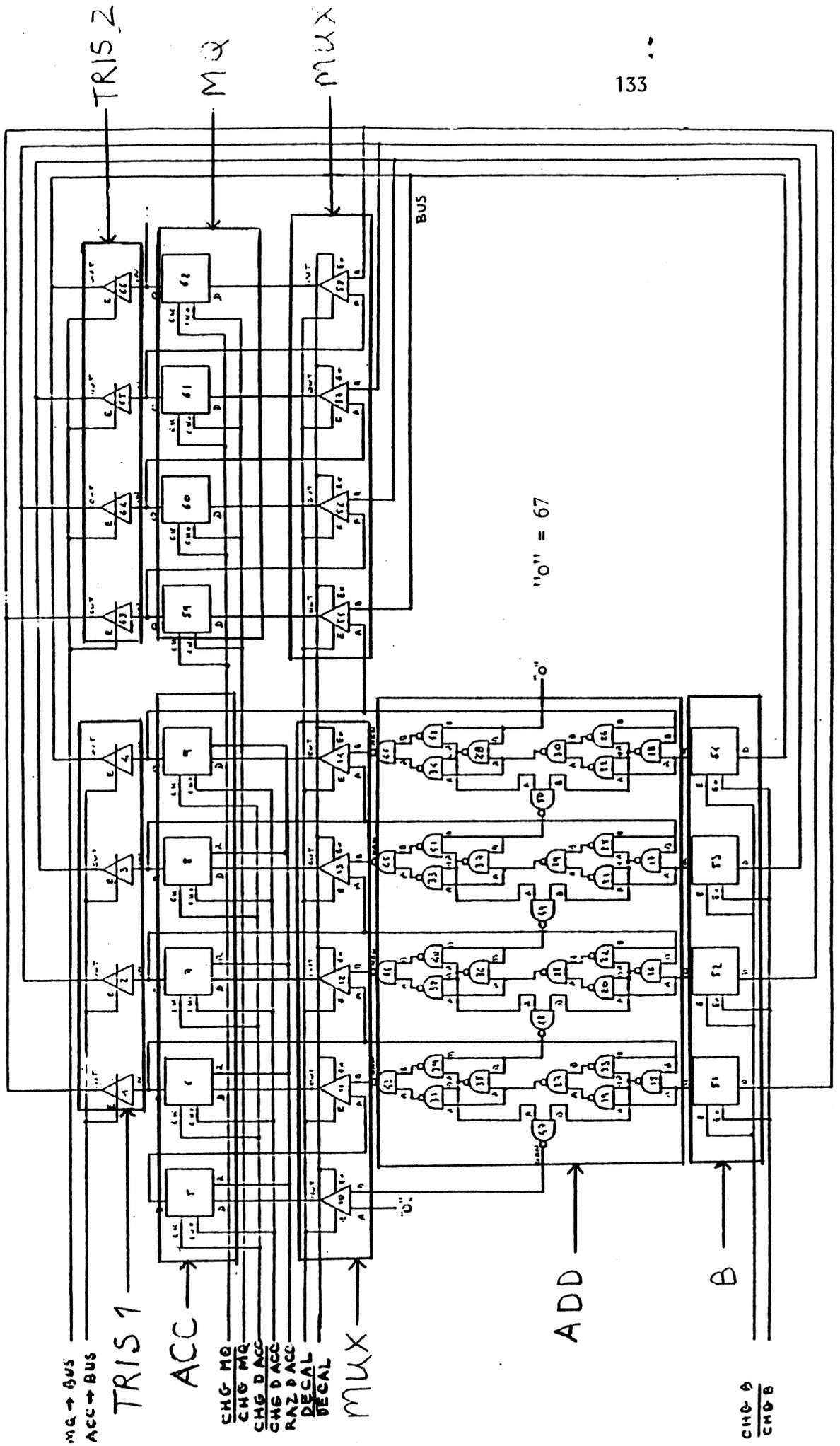


Fig. 4. Resultat de la classificacion  
(cas:  $\Delta_1$ ,  $\alpha = 3.1$ ,  $\beta = 0.45$ )

CLASSES	
1	5
2	15
3	16
4	17
5	18
6	59
7	64
8	1
9	2
10	3
11	4
12	6
13	9
14	10
15	11
16	12
17	13
18	14
19	51
20	52
21	53
22	54
23	55
24	56
25	57
26	58
27	63
28	67
	9
	23
	20
	21
	22
	61
	19
	48
	49
	50
	65
	35
	36
	37
	38
	60
	31
	32
	33
	34
	30
	27
	28
	29
	30
	43
	44
	45
	46
	39
	40
	41
	42

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	
1																													
2																													
3																													
4																													
5																													
6																													
7																													
8																													
9																													
10																													
11																													
12																													
13																													
14																													
15																													
16																													
17																													
18																													
19																													
20																													
21																													
22																													
23																													
24																													
25																													
26																													
27																													
28																													

Fig. 5. Presentation matricielle des resultats de classification

(cas:  $\Delta_1$ ,  $\alpha = 3.1$ ,  $\beta = 0.45$ )

Remarque : Un coefficient  $a_j$  de cette matrice est égal à 1 si l'élément

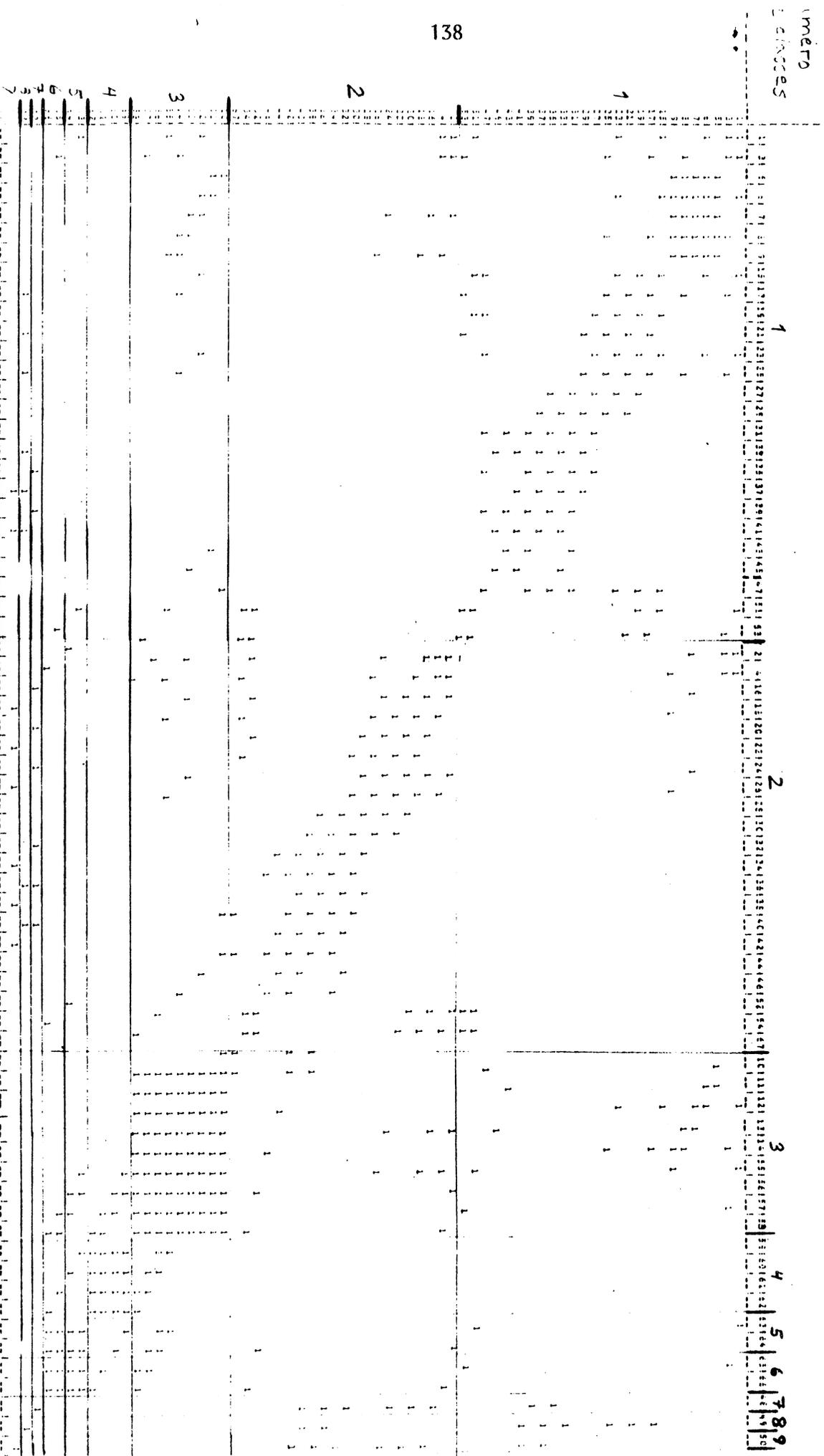


Fig. 7. Resultat de la classificació

(cas:  $\Delta_2$ ,  $\alpha = 0.5$ ,  $\beta = 0.38$ )

Classes	1	2	3	4	5	6	7	8	9
1	1	2	10	59	63	65	48	49	50
2	51	2	40	67	4	42	58	57	56
3	23	4	54	18	26	22	30	38	34
4	15	67	16	24	20	28	36	32	24
5	9	52	54	18	26	22	30	38	34
6	7	41	8	5	3	53	25	17	21
7	47	24	20	28	36	32	24	44	44
8	19	30	38	34	46	43	43	43	43
9	27	35	31	43	43	43	43	43	43

Fig. 8. Presentation matricielle des resultats de classification  
 (cas:  $\Delta_2$ ,  $\alpha = 0.5$ ,  $\beta = 0.38$ )



	1	2	3	4	5	6	7	8	9
1		0,2	0,37	0	0,13	0,13	0,22	0,65	0,22
2			0,27	0	0,15	0,15	0,7	0,27	0,78
3				0,33	0,34	0,26	0	0	0
4					0	0,36	0	0	0
5						1,33	0	0	0
6							0	0	0
7								0	0
8									0
9									

Figure 9

Matrice de pseudo corrélation entre les classes  
(cas :  $\Delta_2$ ,  $\alpha = 0.5$ ,  $\beta = 0.38$ )

Fig. 10. Resultat de la classificació

(cas:  $\Delta_3$ ,  $\alpha = 0.67$ ,  $\beta = 0.37$ )

CLASSES	
1	1
2	2
	44
	30
3	3
4	5
5	10
6	59
7	51
8	53
9	63
10	65
	23
	52
	40
	38
	25
	9
	58
	62
	15
	67
	4
	34
	17
	8
	57
	61
	47
	24
	54
	46
	21
	7
	56
	60
	19
	16
	66
	42
	49
	6
	55
	27
	20
	64
	29
	14
	35
	48
	26
	18
	33
	12
	43
	36
	22
	11
	39
	32
	50
	41



	1	2	3	4	5	6	7	8	9	10
1		0,08	0,03	0,14	0,16	0	0,55	0	0,18	0
2			0,12	0,17	0,3	0,07	0,22	0,22	0,22	0,22
3				0,14	0,14	0	0	0,54	0	0,18
4					0,39	0	0	0	0	0
5						0,33	0,15	0,15	0,3	0,3
6							0	0	0,32	0,32
7								1	1	0
8									0	1
9										1
10										

Figure 12

Matrice de pseudo corrélation entre les classes  
(cas :  $\Delta_3$ ,  $\alpha = 0.67$ ,  $\beta = 0.37$ )

Fig. 13 Resultat de la classificacion  
(cas:  $\Delta_4$ ,  $\alpha = 4.8$ ,  $\beta = 0.34$ )

Classes

Class	1	2	3	4	5	6	7	8	9	10
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										
35										
36										
37										
38										
39										
40										
41										
42										
43										
44										
45										
46										
47										
48										
49										
50										
51										
52										
53										
54										
55										
56										
57										
58										
59										
60										
61										
62										
63										
64										
65										
66										
67										
68										
69										
70										
71										
72										
73										
74										
75										
76										
77										
78										
79										
80										

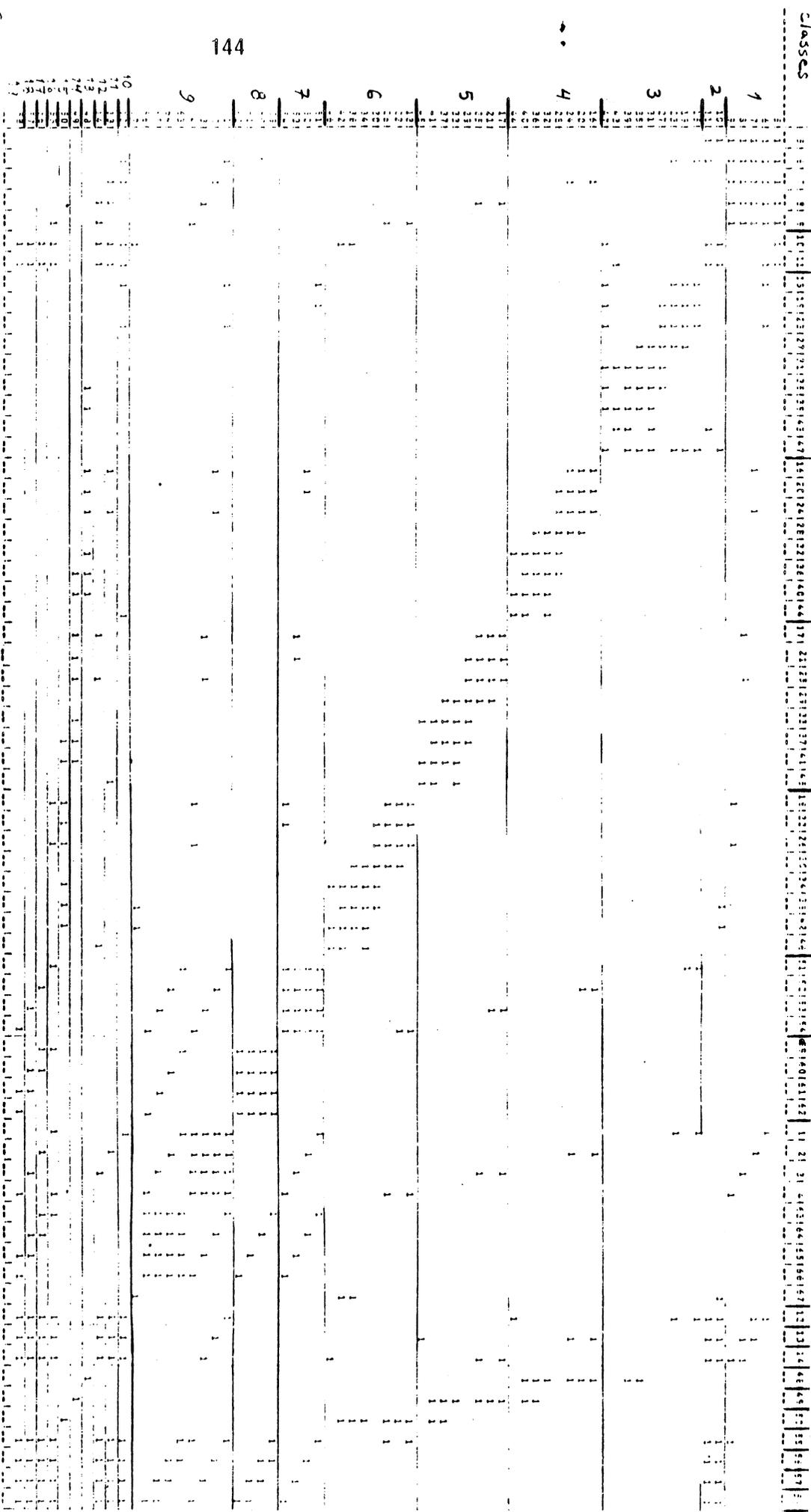


Fig. 14. Presentation matricielle des resultats de classification  
 (cas:  $\Delta_4$ ,  $\alpha = 4.8$ ,  $\beta = .0.34$ )

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
1		0,45	0,1	0,12	0,12	0,12	0	0	0,2	0,5	0,5	0,5	0	0	0	0,26	0	0	0	
2			0,22	0	0	0,26	0	0	0,12	1,12	1,12	1,15	0	0	0	1,15	1,15	1,15	1,15	
3				0	0	0	0,12	0	0,08	0,32	0	0	0,38	0	0	0	0	0	0	
4					0	0	0,14	0	0,09	0,22	0,45	0	1,34	0,45	0	0	0	0	0	
5						0	0,14	0	0,09	0	0,22	0,45	0	1,34	0,45	0	0	0	0	
6							0,14	0	0,1	0	0	0,22	0	0	1,34	0,45	0	0	0	
7								0	0,5	0	0	0	0	0	0	0,32	0,32	0,32	0,32	
8									0,25	0	0	0	0	0	0	0,32	0,63	0,63	0,63	
9										0,2	0,2	0,2	0	0	0	0,6	0,6	0,6	0,6	
10											1	1	0	0	0	1	1	1	1	
11												1	0	0	0	1	1	1	1	
12													0	0	0	1	1	1	1	
13														0	0	0	0	0	0	
14															0	0	0	0	0	
15																0	0	0	0	
16																	1	1	1	
17																		1	1	
18																			1	
19																				

Figure 15 : Matrice de pseudo corrélation entre les classes

(cas :  $\Delta_4$ ,  $\alpha = 4.8$ ,  $\beta = 0.34$ )

Nous allons dans ce qui suit, appliquer à titre d'illustration le procédé de résolution décrit dans III-2.5 au résultat trouvé par la méthode de classification présentée dans III-1 relativement à ce dernier cas par exemple ( $\Delta_4$  ,  $\alpha_4 = 4,8$ ,  $\beta = 0.34$ )

Nous avons choisi comme classes actives (figure 14) les classes : 1,3,4,5,6, 7,8,9.

Les classes 2,10,11,12,13,14,15,16,17,18, 19 jouent le rôle d'éléments supplémentaires.

L'application de l'ACP avec éléments supplémentaires à partir du tableau de la figure 15 a donné le résultat suivant (figure 16).

PLAN DE PROJECTION DES 19 POINTS SUR LES AXES 1 ET 2

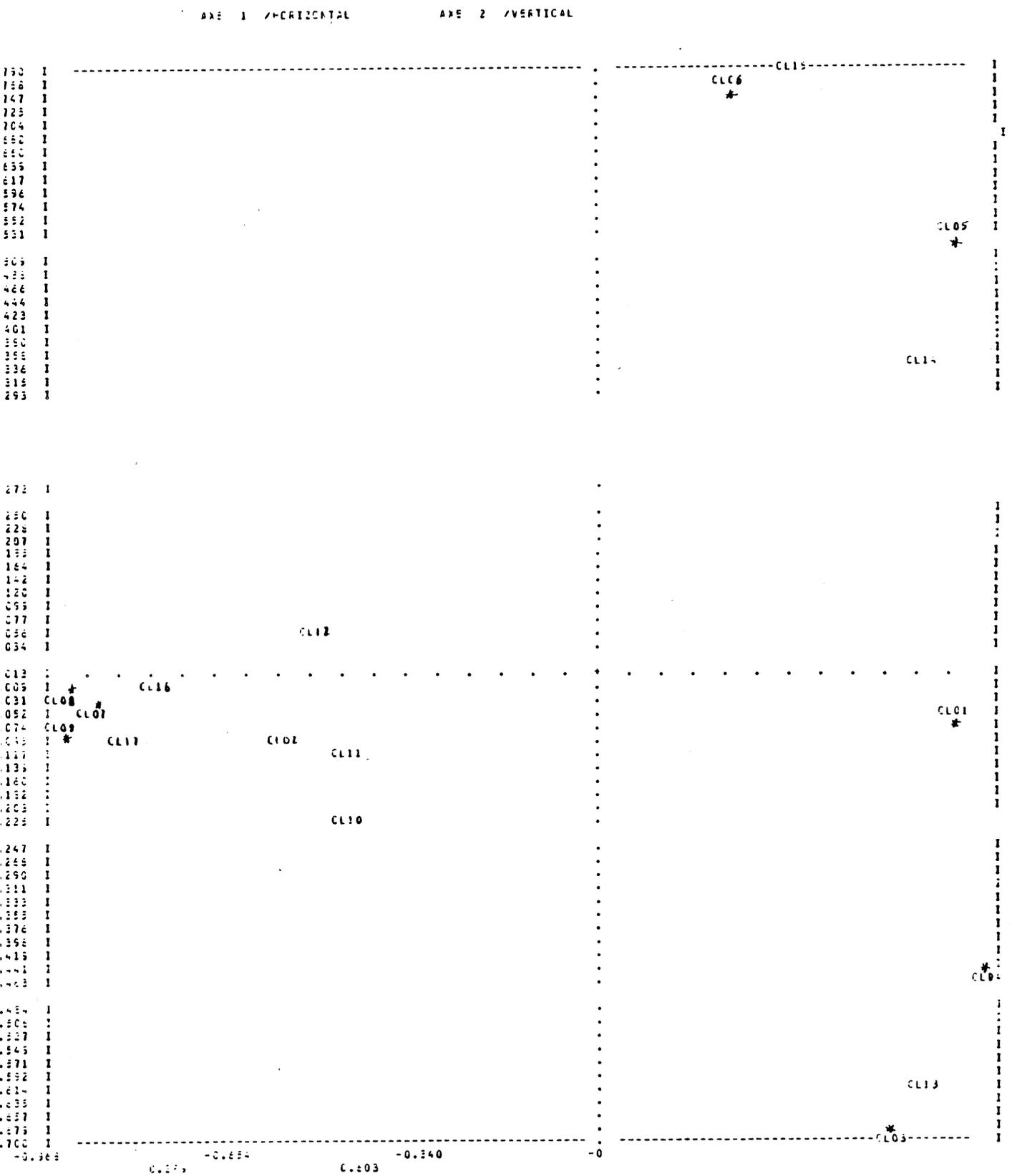


Figure 16

Il resterait à voir comment tirer profit au maximum d'une telle représentation graphique pour résoudre le problème posé au III-2.4. et choisir une bonne solution parmi l'ensemble des solutions qui se présentent.

Nous pouvons par exemple choisir la solution de la figure 17. Les figures 18 et 19 permettront de constater les améliorations obtenues après le choix de cette solution.

PLAN DE PROJECTION DES 15 POINTS SUR LES AXES 1 ET 2

AXE 1 / HORIZONTAL

AXE 2 / VERTICAL

0.790 I  
0.758 I  
0.747 I  
0.725 I  
0.704 I  
0.680 I  
0.660 I  
0.635 I  
0.617 I  
0.596 I  
0.574 I  
0.552 I  
0.531 I  
0.509 I  
0.488 I  
0.466 I  
0.444 I  
0.423 I  
0.401 I  
0.380 I  
0.358 I  
0.336 I  
0.315 I  
0.293 I

0.272 I  
0.250 I  
0.228 I  
0.207 I  
0.185 I  
0.164 I  
0.142 I  
0.120 I  
0.099 I  
0.077 I  
0.056 I  
0.034 I  
0.012 I  
0.009 I  
0.031 I  
0.052 I  
0.074 I  
0.095 I  
0.117 I  
0.139 I  
0.160 I  
0.182 I  
0.203 I  
0.225 I

0.247 I  
0.258 I  
0.290 I  
0.311 I  
0.333 I  
0.355 I  
0.376 I  
0.398 I  
0.419 I  
0.441 I  
0.463 I  
0.484 I  
0.506 I  
0.527 I  
0.549 I  
0.571 I  
0.592 I  
0.614 I  
0.635 I  
0.657 I  
0.679 I  
0.700 I

-0.363  
0.027  
-0.634  
0.093  
-0.340  
0

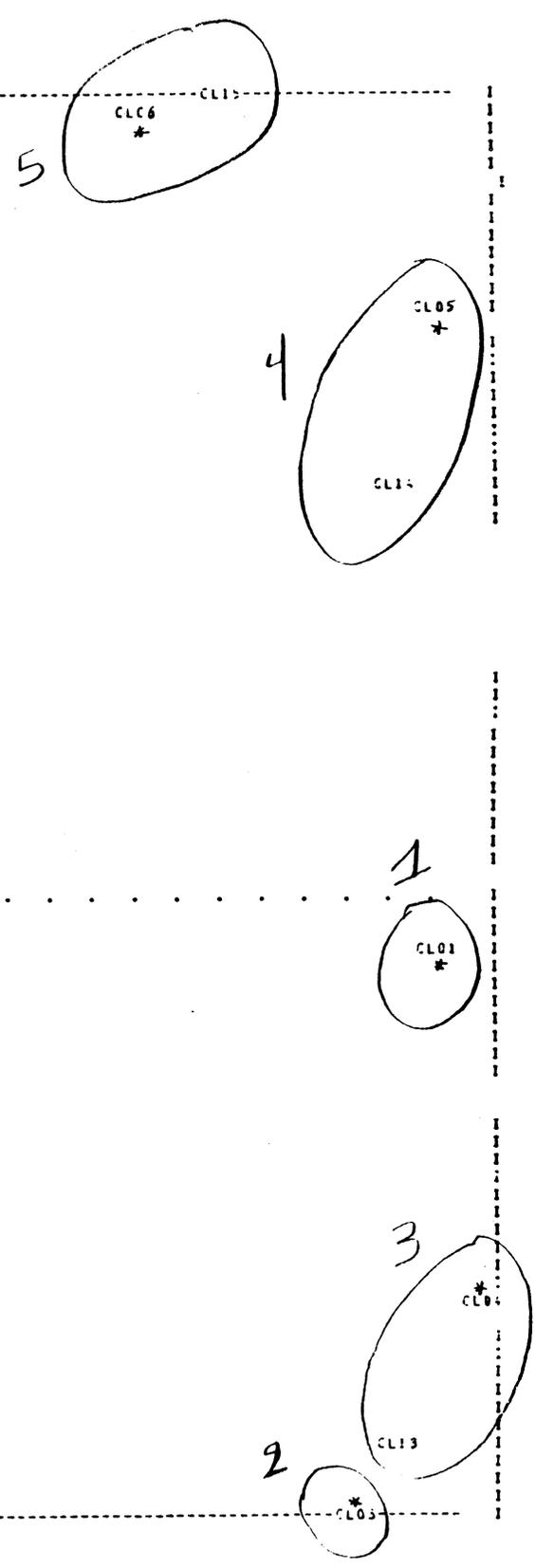


Figure 17

Proposition d'une solution

Figure 18 : Présentation matricielle des résultats de classification après ACP

(cas :  $\Delta_4$ ,  $\alpha = 4,8$ ,  $\beta = 0.34$ )

Numero classes	1	2	3	4	5	6
1	1	1	1	1	1	1
2	1	1	1	1	1	1
3	1	1	1	1	1	1
4	1	1	1	1	1	1
5	1	1	1	1	1	1
6	1	1	1	1	1	1

	1	2	3	4	5	6
1		0,1	0,1	0,1	0,1	0,32
2			0,07	0	0	0,13
3				0,07	0	0,12
4					0,07	0,12
5						0,18
6						

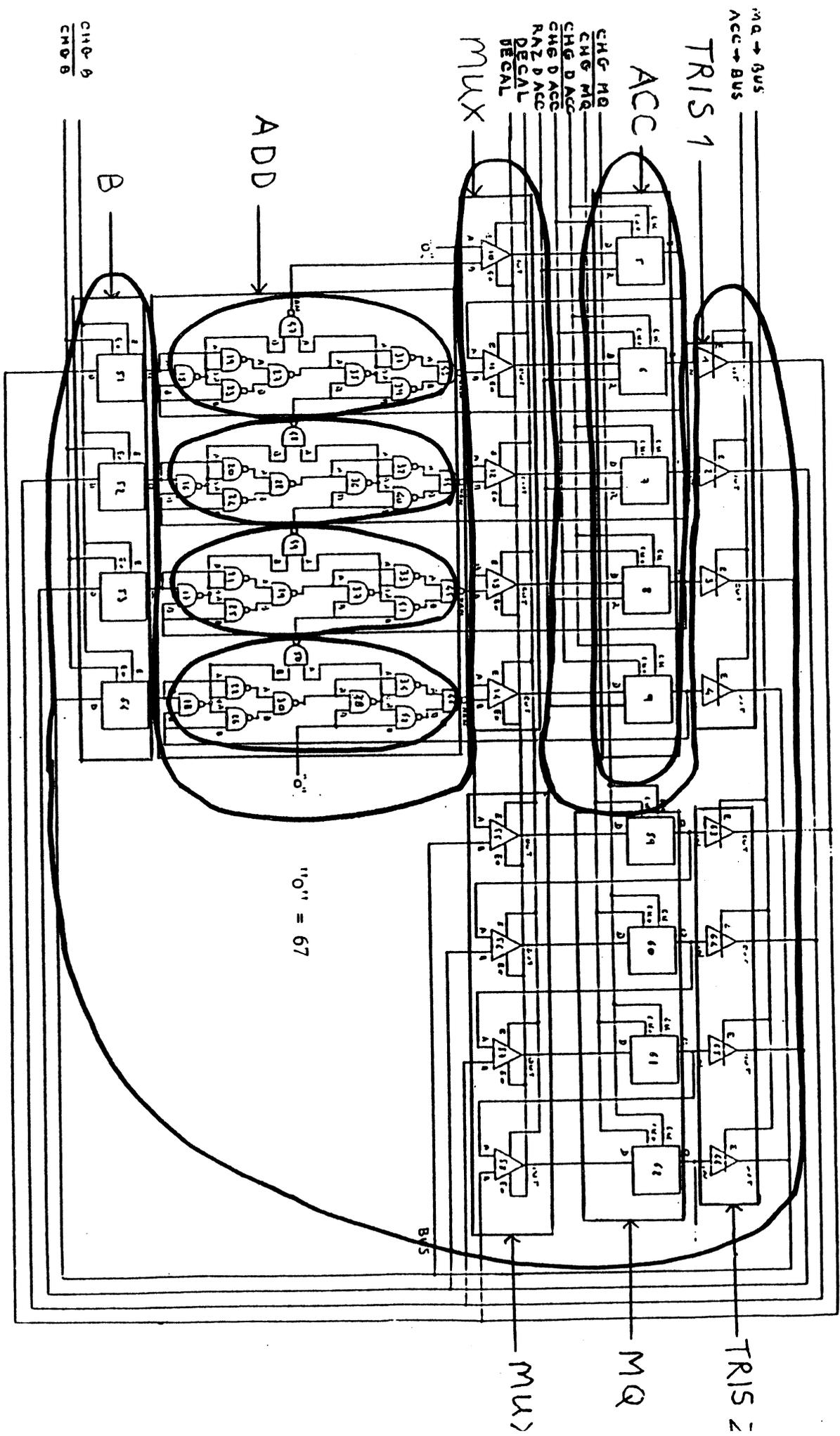
Figure 19

Matrice de pseudo corrélation entre  
les classes trouvées après ACP

(cas :  $\Delta_4$ ,  $\alpha = 4,8$ ,  $\beta = 0.34$ )

Représentation de la solution proposée après ACP sur

Fig. 20 Schéma logique de la partie opérative du circuit traité



CHD-9  
CHD-8

## CONCLUSION

Dans cette troisième partie nous avons proposé l'emploi des méthodes de classification en vue de subdiviser les problèmes de taille trop importante. Ces méthodes étant uniquement des méthodes exploratoires, on a dans un deuxième temps proposé un moyen de contrôle de qualité pour juger les résultats obtenus auprès de l'implanteur, essayer de résoudre les petits problèmes découlant de la classification (classes fortement liées, singletons...) et réajuster les classes.

L'illustration par un exemple de petite taille nous a permis d'entrevoir les difficultés que posent l'emploi de ces méthodes ; de plus l'automatisation complète du système pose encore des problèmes mathématiques (sensibilités aux paramètres) et algorithmiques (choix lors du réajustement des classes après ACP...).

On peut enfin remarquer que le découpage obtenu n'est pas un découpage fonctionnel, en particulier il ne retourne pas la hiérarchisation intuitive que l'on donnerait à partir du schéma architectural.

Par exemple, dans le cas traité (fig. 20), on ne retrouve pas le découpage en bit-slice, et on ne retrouverait pas non plus le découpage partie opérative/partie contrôle. La solution obtenue n'est pas une solution évidente au premier abord, mais qui s'explique quand on regarde le schéma de plus près.

Une hiérarchisation de type fonctionnel nous fournirait des blocs qui pourront être réutilisés pour d'autres circuits (précaractérisés).

Une hiérarchisation par classification se base sur des critères mathématiques. Elle permet de subdiviser un gros problème non traitable en sous-problèmes facilement traitables. Par exemple, elle cherche à minimiser le nombre de fils entre les blocs, donc à faciliter l'interlacement de ces blocs chose qui n'est pas forcément évidente avec une hiérarchisation fonctionnelle.

## REFERENCES

- [BAR 84a] BARRA J.R.,  
"Modèles virtuels",  
4ème Rencontre Franco-Belge de statisticiens, Louvain,  
Publications des facultés universitaires de St Louis, Bruxelles  
1984.
- [BAR 84b] BARRA J.R., BALLIF J.F., BECKER M., TRICOT M.,  
"Application d'un algorithme de classification à un problème  
d'agrégation pour la résolution d'un modèle de systèmes  
informatiques",  
Rapport de recherche n°451, LCS-IMAG, Grenoble, 1984.
- [BAR 85] BARRA J.R., BECKER M., KOUKA E.F., TRICOT M.,  
"Elaboration de plan de masse d'un placement automatisé de  
blocs",  
Colloque National "Conception de circuits à la demande sur  
réseaux prédifusés et précaractérisés", 23-24 mai 1985,  
LCS-INPG, Grenoble.
- [BEL 84] BELLON A.,  
"Optimisation des méthodes d'implantation automatisées sur les  
réseaux précaractérisés",  
Thèse de Docteur 3ème Cycle, LCS-USMG, Grenoble, Décembre  
1984.

- [BRE 77] BREUER M.,  
"Min-cut placement"  
J. Des. Automat. Fault Tolerant Comput., Vol.1, n°4, pp.343-362,  
Octobre 1977,  
et sur le même sujet :  
"A class of min cut of placement algorithms"  
Proc. of the 14th Design Automation Conference, New Orleans,  
Louisiane, Juin 1977, pp.284-290.
- [BUR 83] BURSTEIN M., HONG S.J., PELAVIN R.,  
"Hierarchical VLSI layout : simultaneous placement and wiring of  
gate arrays",  
IFIP Symposium "VLSI 83", Trondheim, Norvège, 16-19 Aout 1983,  
pp.45-60.
- [CHA 85] CHAISEMARTIN P.,  
"Implantation d'un multiplieur sur réseau prédifusé  
CIT-ALCATEL",  
Conception sur les réseaux programmables et prédifrusés,  
Fascicule 3, Formation professionnelle continue, INPG, 1985.
- [CRA 85] CRASTES DE PAULET M., RARIVOMANANA J.,  
"Simulation fonctionnelle fine : le langage CADOC",  
Colloque National "Conception de circuits à la demande sur  
réseaux prédifusés et précaractérisés", 23-24 mai 1985,  
LCS-INPG, Grenoble.

- [DON 80] DONATH W.E.,  
"Complexity theory and design automation",  
17th Design Automation Conference, Minneapolis, Minnesota, June  
1980, pp.412-419.
- [GLA 59] GLASER R.H.,  
"A quasi-simplex method for designing suboptimal packages for  
electronic building blocks"  
Proc. Comp. Appl. Symp. Armour Research Foundation, Illinois,  
Inst. of Tech., 1959, pp.100-111.
- [GOT 79] GOTO S.,  
"A two dimensional placement algorithm for the master slice LS  
layout problem",  
Proc. 16th Design Automation Conference, San Diego, June 1979,  
pp.11-17.
- [HAN 72] HANAN M., KURTZBERG J.M.,  
"Placement techniques"  
Edited by M.A. Breuer, Vol.1, Prentice Hall, 1972, pp.211-282.
- [HAN 73] HANAN M., WOLFF P.K., ANGULI B.J.,  
"Some experimental results on placement techniques"  
Proc. of the 13 th Design Automation Conference, 1973, San  
Francisco, pp.214-224.

- [KER 70] KERNIGHAN B.W., LIN S.,  
"An efficient procedure for partitioning graph",  
Bell System Tech. J. , vol.49, Février 1970, pp.291-308.
- [KER 72] KERNIGHAN B.W., SCHWEIKERT D.G.,  
"A proper model for the partitioning of electrical circuits",  
Proc. of the 9th Design Automation Workshop, Juin 1972,  
pp.57-62.
- [KIR 83] KIRPATRICK S., GELATT C.D., VECCHI Jr. M.P.,  
"Optimisation by simulated annealing",  
Science, vol.220, n°4598, 13 mai 1983, pp.671-680.
- [KRI 84] KRISHNAMURTHY B.  
"An improved Min-Cut algorithm for partitioning VLSI networks"  
IEEE Transactions on Computer, vol.C33, n°5, Mai 1984,  
pp.438-446.
- [KUN 72] KUNTZMANN J.,  
"Théories des réseaux (graphes)",  
Edité par Dunod, Paris, 1972.
- [KUR 65] KURTZBERG J.M.,  
"Algorithms for blackplane formation",  
Microelectronics in Large Systems, Spartan Books, 1965,  
pp.61-76.

- [LAU 79] LAUTCHER V.,  
"A min-cut placement algorithm for general cell assemblies based  
on a graph representation",  
16th Design Automation Conference, San Diego, California, Juin  
1979, pp.1-9.
- [LEB 79] LEBART L., FENELON,  
"Traitement des données statistiques"  
Dunod, 1979
- [LIT 63] LITTLE J.D.C. et al.,  
"An algorithm for the traveling salesman problem",  
J. Oper. Es., Vol.1, 1963, pp.972-989.
- [MAR 84] MARTINET B.  
"Tracé automatique de connexions sur réseaux prédéfinis CMOS",  
DEA d'Informatique, INPG. Grenoble, Juin 1984.
- [MAS 85] Conférence de Mr MASSON,  
Journée INFOPT, G.C.I.S., 24 janvier 1985, Grenoble.
- [PAL 84] PALEZENSKI M.,  
"Performance of Algorithms for initial placement",  
21st Design Automation Conference, IEEE, 1984, pp.399-404.

- [PRE 78] PREAS B.T., GWYN C.W.,  
"Methods for hierarchical automatic layout of custom LSI circuit masks",  
Proc. du 15th Design Automation Conference, pp.206-212, Juin 1978.
- [REI 77] REINGOLD E.M., NIEVERGELT J., DEO N.,  
"Combinatorial Algorithms"  
Prentice Hall, 1977, Ch.8, Graph Algorithms.
- [RIV 82] RIVEST R., FIDUCCIA C.,  
"A greedy channel router",  
Proc. 19th Design Automation Conference, June 1982, pp418-434.
- [SER 80] SERREO G., LEBLOND A., VERDILLON A.,  
"Automatic layout of symbolic MD-MOS circuits",  
1st ICC Conference, Port Chester, New York, Octobre 1980,  
pp.772-776.
- [SER 81] SERRERO G., MALLADI R., VERDILLON A.,  
"Automatic placement of rectangular blocks ith the interconnections channels"  
18th Design Automation Conference, Nashville, Tennessee, Juin 1981, pp.419-425.

- [SER 82] SERRERO G.,  
"Implantation symbolique automatisée de circuits intégrés",  
Thèse INPG, Grenoble, Mars 1982.
- [SHI 80] SHIRAISHI H. HIROSE F.,  
"Efficient placement and routing technique for master slice  
LSI",  
Proc. 17th Design Automation Conference, pp. 458-464, 1980.
- [SOU 81] SOUKUP J.,  
"Circuit Layout"  
Proceeding of the IEEE, vol.69, n°10, Octobre 1981, pp.1281-1304
- [TSI 84] TSITSIMIS J.,  
"Implantation automatisée de circuits précaractérisés et  
prédiffrusés",  
Thèse INPG, Grenoble, Octobre 1984.



**A U T O R I S A T I O N de S O U T E N A N C E**

VU les dispositions de l'article 3 de l'arrêté du 16 avril 1974

VU le rapport de présentation de Madame le Professeur G. SAUCIER

**Monsieur JANATI IDRISSE Mohammed Abdou**

est autorisé à présenter une thèse en soutenance en vue de l'obtention du titre de DOCTEUR de TROISIEME CYCLE, spécialité "Informatique".

Fait à Grenoble, le 18 juin 1985

Le Président de l'I.N.P.-G

**D. BLOCH**

Président

de l'Institut National Polytechnique  
de Grenoble

*P.O. le Vice-Président,*





## RESUME

Cette étude concerne l'implantation automatisée des circuits intégrés sur réseaux prédiffusés.

Après une revue critique de méthodes de placement existantes, ce travail développe plus précisément les méthodes ascendantes. Il insiste sur trois points : une préstructuration logique du réseau à implanter, une prise en compte des contraintes topologiques et une forte prédiction de la connectique afin de gérer les ressources critiques. Il présente aussi un travail pratique illustrant ces considérations. Il s'agit de la conception d'une méthode et d'un logiciel d'implantation automatisée sur réseau prédiffusés CMOS à une couche d'aluminium.

Enfin l'utilisation des méthodes de classification pour hiérarchiser les problèmes complexes est introduite pour former des agrégats d'éléments à placer.

## MOTS CLES

Réseau prédiffusé, Circuit intégré, CMOS, Implantation automatisée, Placement, Méthode ascendante, Contrainte topologique, Préstructuration logique, Tracé, Classification.

