Utilisation de l'algèbre de Boole en logique mathématique

Mireille Dupraz

n° d'Ordre

UNIVERSITE DE GRENOBLE

FACULTE DES SCIENCES

UTILISATION DE L'ALGEBRE DE BOOLE EN LOGIQUE MATHEMATIQUE.

-:-:-:-:-:-:-:-:-:-:-

THESE

POUR OBTENIR

Le titre de DOCTEUR DE TROISIEME CYCLE

MATHEMATIQUES APPLIQUEES

-:-:-:-:-:-:-:-:-:-:-

Présentée par

Mireille DUPRAZ

-:-:-

Thèse soutenue le 19 Octobre 1966

Devant la commission d'Examen :

M. KUNTZMANN — Président

M. VAUQUOIS, Mme BERTRANDIAS — Examinateurs

n° d'Ordre

UNIVERSITE DE GRENOBLE

FACULTE DES SCIENCES

UTILISATION DE L'ALGEBRE DE BOOLE EN LOGIQUE MATHEMATIQUE.

-:-:-:-:-:-:-:-:-:-

THESE

POUR OBTENIR

Le titre de DOCTEUR DE TROISIEME CYCLE

MATHEMATIQUES APPLIQUEES

-:-:-:-:-:-:-:-:-:-

Présentée par

Mireille DUPRAZ

-:-:-

Thèse soutenue le 19 Octobre 1966

Devant la commission d'Examen :

M. KUNTZMANN — Président

M. VAUQUOIS
Mme BERTRANDIAS — Examinateurs

F A C U L T E   D E S   S C I E N C E S

LISTE DES PROFESSEURS

DOYENS HONORAIRES :

M. FORTRAT P.
M. MORET L.

DOYEN :

M. WEIL L.

PROFESSEURS TITULAIRES :

MM.	NEEL L.	Magnetisme
	HEILMANN R.	Chimie Organique
	KRAVTCHENKO J.	Mecanique Rationnelle
	CHABAUTY C.	Mathematiques Pures
	PARDE M.	Potamologie
	BENOIT J.	Radioelectricité
	CHENE M.	Chimie Papetière
	BESSON J.	Electrochimie
	WEIL L.	Thermodynamique
	FELICI N.	Electrostatique
	KUNTZMANN J.	Mathematiques Appliquées
	BARBIER R.	Géologie Appliquée
	SANTON L.	Mécanique des Fluides
	OZENDA P.	Botanique
	FALLOT M.	Physique Industrielle
	GALVANI O.	Mathématiques
	MOUSSA A.	Chimie Nucléaire et Radioactivité

TRAYNARD P.	Chimie Générale
SOUTIF M.	Physique Générale
CRAYA A.	Hydrodynamique
REULOS R.	Théorie des Champs
AYANT Y.	Physique Approfondie
GALISSOT F.	Mathématiques Pures
Melle LUTZ E.	Mathématiques Générales
MM. BLAMBERT M.	Mathématiques
BOUCHEZ	Physique Nucléaire
LLIBOUTRY L.	Géophysique
MICHEL R.	Géologie et Minéralogie
BONNIER E.	Métallurgie
DESSAUX G.	Physiologie Animale
PILLET E.	Electrotechnique
DEBELMAS J.	Géologie Générale
GERBER R.	Mathématiques Pures
PAUTHENET R.	Electrotechnique
VAUQUOIS B.	Calcul Electronique
SILBER R.	Mécanique des Fluides
MOUSSIEGT J.	Electronique
BARBIER J.C.	Physique Expérimentale
BUYLE-BODIN M.	Electronique
KOSZUL J.L.	Mathématiques
DREYFUS B.	Thermodynamique
VAILLANT F.	Zoologie
KLEIN J.	Mathématiques Pures
SENGEL P.	Zoologie
ARNAUD P.	Chimie
BARJON R.	Physique Nucléaire
BARNOUD F.	Biosynthese de la Cellulose

PROFESSEURS ASSOCIES :

MM.	WAGNER H.	Botanique
	NAPP-ZINN K.	Botanique

PROFESSEURS SANS CHAIRE :

Mme	KOFLER L.	Botanique
	DEPASSEL R.	Mécanique
	PERRET R.	Servomecanisme
Mme	BARBIER M.J.	Electrochimie
MM.	COHEN J.	Physique
	GIDON P.	Géologie
Mme	SOUTIF J.	Physique Générale
MM.	GIRAUD P.	Géologie
	GASTINEL N.	Mathématiques Appliquées
	LACAZE A.	Thermodynamique
	GLENAT R.	Chimie Organique
	BRISSONNEAU P.	Physique Générale
	DUCROS P.	Minéralogie
	ANGLES D'AURIAC	Mécanique des Fluides
	ROBERT A.	Chimie Papetière
	COUMES A.	Electronique
	PEBAY-PEROULA	Physique
	DEGRANGE C.	Zoologie
	GAGNAIRE D.	Chimie Papetière
	RASSAT A.	Chimie
	PERRIAUX J.	Géologie
	BARRA J.	Mathématiques Appliquées

PROFESSEURS HONORAIRES

MM.	FORTIER A.	Mécanique des fluides
	BRELOT M.	Mathematiques
	WOLFERS F.	Physique
	DORIER A.	Zoologie

MAITRES DE CONFERENCES

MM.	BIAREZ J.	Mécanique des fluides
	DODU J.	Mécanique des fluides
	DOLIQUE J.M.	Electronique
	HACQUES G.	Mathématiques Appliquées
	LANCIA R.	Physique Automatique
	POULOUJADOFF M.	Electrotechnique
	KAHANE A.	Physique
Mme	BONNIER J.	Chimie
Mme	KAHANE J.	Physique
MM.	DEPORTES C.	Chimie Minerale
	DEPOMMIER P.	Physique Nucléaire
	CAUQUIS G.	Chimie Générale
	BONNET G.	Physique
Mme	BOUCHE L.	Mathématiques
MM.	COLOBERT L.	Physiologie Animale
	PAYANT J.J.	Mathématiques
	CAUBET J.P.	Mathématiques
	LAURENT P.J.	Mathématiques Appliquées
	BERTRANDIAS J.P.	Mathématiques Appliquées
	BRIERE G.	Physique
	LAJZEROWICZ J.	Physique
	VALENTIN J.	Physique
	DESPRE P.	Metallurgie
	BONNETAIN L.	Chimie Minérale

MAITRES DE CONFERENCE ASSOCIE :

M. RADELLI L. Géologie

MAITRE DE CONFERENCE HONORAIRE :

M. GASTEX A. Essais électriques

Je voudrais remercier tout particulièrement :

Monsieur le Professeur KUNTZMANN, directeur de l'Institut de Mathématiques Appliquées de l'Université de Grenoble, qui a dirigé mon travail et m'a aidé de ses nombreux conseils.

Monsieur le Professeur VAUQUOIS et Madame BERTRANDIAS, Maître de Conférences, qui ont bien voulu être membres du Jury.

- TABLE DES MATIERES -

- INTRODUCTION -

Nous nous intéresserons ici au domaine de la logique mathématique et en particulier à la démonstration automatique de théorèmes.

Un théorème, en logique, est une expression pouvant être obtenue à partir d'axiomes et de règles de déduction donnés. L'ensemble de ces axiomes, règles et des théorèmes, qui peuvent en être déduits, constitue un système formel.

Si nous considérons tout d'abord la logique classique, il en existe une interprétation qui établit une correspondance entre les formules du système et l'ensemble des valeurs 'vrai' et 'faux'. Cette correspondance est telle que tous les axiomes et tous les théorèmes, et eux seuls, soient vrais.

Notre but est de montrer ce que peut apporter, à la démonstration automatique, l'emploi de méthodes avancées de manipulation des fonctions booléennes. Nous arriverons à des résultats pour le cas des propositions et celui des prédicats à une variable, mais, comme l'on peut s'y attendre, la méthode employée sera difficilement utilisable pour des prédicats contenant plus d'une variable.

Nous considérerons ensuite la logique intuitionniste et essaierons de faire un travail analogue à celui effectué pour la logique classique. Il faudra tout d'abord essayer de valuer les systèmes formels de la logique intuitionniste. Nous pourrons alors montrer que tout axiome et tout théorème, de cette logique, peut être écrit comme identique à 1 dans une algèbre de Post à n valeurs mais la réciproque ne sera pas vraie. Nous pourrons alors donner une condition nécessaire (mais non suffisante) pour qu'une expression soit un théorème et un algorithme efficace pour vérifier cette condition.

- CHAPITRE I -

TRANSPOSITION EN LANGAGE BOOLEEN DU CALCUL CLASSIQUE DES PROPOSITIONS.

I - Généralités :

Un système formel est constitué d'axiomes, et de règles de déduction qui permettent à partir de formules valables d'obtenir d'autres formules valables.

Nous considérerons, dans ce chapitre, uniquement des formules de propositions, c'est-à-dire des formules obtenues à partir d'expressions logiques ne dépendant pas de variables et prenant directement une des deux valeurs 'vrai' ou 'faux'. Si les propositions de départ sont notées A, B, C, D,..., les formules considérées sont construites à l'aide des opérateurs $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$.

Dans le cas de la logique classique des propositions, il est possible d'attacher à toute proposition une des valeurs booléennes 0 et 1 (correspondant respectivement à 'faux' et 'vrai') et d'interpréter les connecteurs logiques comme des opérateurs booléens de manière que les formules valables et elles seules soient des identités à 1. Nous nous proposons d'analyser à l'aide de l'algèbre de Boole, les procédés de reconnaissance proposés dans différents systèmes.

II - Représentation des opérateurs logiques en algèbre de Boole :

Les expressions logiques obtenues à partir des différents opérateurs sont interprétées de la manière suivante :

1) $\neg$ A représente la négation de A, c'est-à-dire est vraie si A est fausse et inversement, donc en algèbre de Boole peut s'exprimer par A'.

2) $A \wedge B$ est une expression vraie si et seulement si A et B le sont donc correspond au produit booleen A.B.

3) $A \vee B$ est une expression vraie si et seulement si au moins une des formules A ou B est vraie donc peut se représenter par la somme booléenne A + B.

4) $A \rightarrow B$ est une expression vraie si et seulement si A est fausse ou B est vraie donc peut s'exprimer, en algèbre de Boole, par A' + B.

5) $A \leftrightarrow B$ est une expression vraie si et seulement si A et B sont simultanément vraies ou fausses donc, en algèbre de Boole, s'écrira
$AB + A'B' = (A'+B)(A + B')$.

Il existe aussi l'opérateur d'implication généralisée $\Rightarrow$ portant sur des chaînes de formules et dont l'interprétation est la suivante :

Si π et ρ sont des chaînes de formules, l'èxpression $\pi \Rightarrow \rho$ est vraie si l'une des formules de π est fausse ou si l'une des formules de ρ est vraie. D'où si $\pi_1, \ldots, \pi_n$ désignent les formules de π, $\rho_1, \ldots, \rho_m$ celles de ρ nous pouvons exprimer $\pi \Rightarrow \rho$ par $\pi_1 \wedge \pi_2 \wedge \ldots \wedge \pi_n \rightarrow \rho_1 \vee \rho_2 \vee \ldots \vee \rho_m$ c'est-à-dire en algèbre de Boole par :

$$(\pi_1 . \pi_2 \ldots \pi_n)' + \rho_1 + \rho_2 + \ldots + \rho_m = \pi'_1 + \pi'_2 + \ldots + \pi'_n + \rho_1 + \rho_2 + \ldots + \rho_m.$$

Le symbolisme logique emploie aussi le signe $\vdash$ pour indiquer une déduction $\Gamma \vdash B$, c'est-à-dire qu'à partir des formules de Γ supposées vraies nous pouvons déduire B vraie. Etant donné que toute formule valable prend la valeur 1, et réciproquement, nous pouvons remplacer dans notre travail $\dashv$ par $\Rightarrow$. En effet la seule possibilité qui contredirait $\Gamma \vdash B$ est que toutes les formules de Γ soient vraies et que B soit fausse. Donc $\Gamma \vdash B$ peut être considérée comme vraie si B est vraie ou si l'une au moins des formules de Γ est fausse, ce qui revient bien à l'interprétation de l'implication généralisée. Il faut cependant remarquer que $\vdash B$ signifie que nous devons montrer que B est identique à 1, il faut donc interpréter l'expression vide comme un 1 et non comme un zéro.

Système de Kleene :

Remarquons que les systèmes logiques ne prévoient pas toujours une méthode systématique de reconnaissance des formules valables, soit par exemple le système de Kleene :

Opérateurs employés : $\neg$, $\wedge$, $\vee$, $\rightarrow$.

L'opérateur $\leftrightarrow$ est défini comme une intersection d'implications.

Axiomes : Ils sont au nombre de 10.

1. $A \rightarrow (B \rightarrow A)$.
2. $(A \rightarrow B) \rightarrow ((A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C))$.
3. $A \rightarrow (B \rightarrow A \wedge B)$.
4. $A \wedge B \rightarrow A$.
5. $A \wedge B \rightarrow B$.
6. $A \rightarrow A \vee B$.
7. $B \rightarrow A \vee B$.
8. $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow (A \vee B \rightarrow C))$.
9. $(A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A)$.
10. $\neg \neg A \rightarrow A$.

Règle de déduction : $A, A \rightarrow B \vdash B$.

Pour déduire une formule il faut l'obtenir par la règle de déduction à partir d'une suite de formules qui sont soit des axiomes soit des formules déjà déduites.

L'établissement de cette suite de formules n'est en général pas simple et ceci ne nous donne pas un procédé systématique de reconnaissance de formules identiques à 1.

Soient encore les systèmes de Church :

a) Dans un premier système le seul opérateur employé est l'implication et il est introduit une constante f qui prend toujours la valeur 0.

Il existe deux règles de déduction et trois axiomes :

1. $A, A \to B \vdash B$. , 2. $A \vdash S_B^b A|$

(où $S_B^b A|$ représente une formule qui résulte de la substitution dans la formule A de chaque occurence de b par B.)

3. $p \to (q \to p)$.
4. $(S \to (p \to q)) \to ((S \to p) \to (S \to q))$.
5. $((p \to f) \to f) \to p$.

b) Dans un deuxième système les deux opérateurs implication et négation sont employés. Les règles de déduction et les axiomes sont les mêmes que pour le système précédent sauf le dernier qui est remplacé par :

$$(\neg p \to \neg q) \to (q \to p).$$

III - <u>Méthode proposée</u> :

<u>Importance de l'opérateur implication</u> :

Les opérateurs $\vee, \wedge, \leftrightarrow$ s'expriment tous en fonction de l'opérateur $\to$ et de la négation, en effet :

- $A \vee B$ s'écrit en algèbre de Boole $A + B$ donc devient : $\neg A \to B$.
- $A \wedge B$ s'écrit $AB = (A'+B')'$ donc devient : $\neg (A \to \neg B)$.
- $A \leftrightarrow B$ s'écrit $(A'+B)(B'+A)$ c'est-à-dire : $\neg((A \to B) \to \neg(B \to A))$.

D'autre part nous avons vu que la déduction pouvait être considérée comme une implication généralisée donc s'exprimer en fonction de l'implication, de l'intersection et de l'union par conséquent en fonction de l'implication et de la négation.

Considérons maintenant toutes les formules en écriture d'algèbre de Boole et en utilisant seulement l'implication et la négation avec la notation préfixée (dont le principal avantage est de nous permettre de supprimer les parenthèses) :

$$\rightarrow x\ y = x' + y$$

Avec cet opérateur implication nous pourrons obtenir toutes les expressions identiques à 1 en considérant la forme canonique. En effet, en algèbre de Boole, considérons une fonction f(x) écrite sous forme canonique :

$$f = x\ f_1 + x' f_0$$

avec la notation utilisée nous obtenons :

$$f = \rightarrow \rightarrow f_0 x \rightarrow \rightarrow f_1\ x'0$$

Or pour que f = 1 il faut et il suffit que :

$$f_0 = f_1 = 1$$

Donc $\boxed{1 \equiv \rightarrow \rightarrow 1x \rightarrow \rightarrow 1x'0.}$ (1)

Pour vérifier si une expression écrite avec l'opérateur implication est identique à 1 il suffira alors de la ramener systématiquement à la forme (1). Nous ferons cela en considérant que :

- La forme canonique a la propriété que toute opération sur f se produit de même sur f_0 et f_1 donc $\rightarrow fg = \rightarrow \rightarrow \rightarrow f_0\ g_0\ x \rightarrow \rightarrow \rightarrow f_1\ g_1\ x'0$ (2)
- $\boxed{\rightarrow 1A = A,\ \rightarrow A0 = A',\ \rightarrow A1 = 1,\ \rightarrow 0A = 1.}$ (3)

<u>Méthode pour identifier une expression à 1</u> :

Si une formule contient plusieurs lettres nous l'écrivons sous forme d'expression lexicographique par rapport à la première, A par exemple, en considérant que A = 0A'+ 1A, que A' = 1A'+ 0A, que toute lettre Y = YA'+ YA si Y est différente de A et A', et en utilisant (2). Nous faisons alors toutes les réductions possibles à l'aide de (3). Nous pouvons avoir comme coefficients, dans la nouvelle formule obtenue, d'autres formules ne contenant pas A, nous faisons alors de même par rapport à B et ainsi de suite jusqu'à ce que nous ayons épuisé toutes les lettres si nécessaire. Si l'expression est identique à 1 les formules intermédiaires puis la forme finale doivent être identiques à (1).

<u>Remarque</u> : cette méthode revient en fait à considérer un nouveau système axiomatique composé uniquement de règles de déduction qui sont les suivantes ($S^{f_1}_{f_2}F|$ représentant une formule qui résulte de la substitution dans la formule F de chaque occurence de f_1 par f_2) :

$$F \vdash S^{X}_{\to\to 0X\to\to 1X'0}F \,| \;;\quad F \vdash S^{X'}_{\to\to 1X\to\to 0X'0}F.$$

$$F \vdash S^{Y}_{\to\to YX\to\to YX'0}F \,| \quad (\text{avec } Y \neq X \text{ et } Y \neq X').$$

$$F \vdash S^{\to fg}_{\to\to\to f_0\, g_0\, X\to\to\to f_1\, g_1\, X'0}F \,|$$

(si $f = \to\to f_0\, X\to\to f_1\, X'0$; $g = \to\to g_0\, X\to\to g_1\, X'0$; f_0, f_1, g_0, g_1 étant soit des constantes, soit des variables de la forme Y, soit des implications entre de telles variables).

$$F \vdash S^{\to 1Y}_{Y}F\,| \;;\; F \vdash S^{\to Y0}_{Y'}F\,| \;;\; F \vdash S^{\to Y1}_{1}F\,| ;\; F - S^{\to 0Y}_{1}F\,|$$

$$F \vdash S^{\to\to 1X\to\to 1X'0}_{1}F\,|$$

Une formule F sera un théorème si elle peut se ramener par ces substitutions à 1.

Essayons cette méthode sur diverses formules. Nous les écrirons tout d'abord avec la notation employée en considérant que :

- $A \vee B$ s'écrit $\to A'B$
- $A \wedge B$ ou, en algèbre de Boole, $AB = (A'+B')'$ et par conséquent nous obtenons : $\to\to AB'0$.

- $A \longleftrightarrow B$ est un produit des implications $(A \longrightarrow B)$ et $(B \rightarrow A)$ donc s'écrit :
$\rightarrow \rightarrow AB \rightarrow \rightarrow BA\ 00$.

Nous vérifions ainsi systématiquement la validité des axiomes du système de Kleene qui sont des expressions identiques à 1.
Considérons par exemple le deuxième :
$(A \rightarrow B) \rightarrow (A \rightarrow (B \rightarrow C) \rightarrow (A \rightarrow C))$ qui s'écrit en notation préfixée
$\rightarrow \rightarrow AB \rightarrow \rightarrow A \rightarrow BC \rightarrow AC$:

$\rightarrow AC = \rightarrow \rightarrow \rightarrow 0CA \rightarrow \rightarrow \rightarrow 1CA'0 = \rightarrow \rightarrow 1A \rightarrow \rightarrow CA'0$.
$\rightarrow BC = \rightarrow \rightarrow \rightarrow BCA \rightarrow \rightarrow \rightarrow BCA'0$.

(Ce qui montre que si $f = \rightarrow BC$: $f_0 = \rightarrow BC$, $f_1 = \rightarrow BC$).

$\rightarrow A \rightarrow BC = \rightarrow \rightarrow \rightarrow 0 \rightarrow BCA \rightarrow \rightarrow \rightarrow 1 \rightarrow BCA'0 = \rightarrow \rightarrow 1A \rightarrow \rightarrow \rightarrow BCA'0$.

$\rightarrow \rightarrow A \rightarrow BC \rightarrow AC = \rightarrow \rightarrow \rightarrow 11A \rightarrow \rightarrow \rightarrow \rightarrow BCCA'0$.

$= \rightarrow \rightarrow 1A \rightarrow \rightarrow \rightarrow \rightarrow BCCA'0$.

$\rightarrow AB = \rightarrow \rightarrow 1A \rightarrow \rightarrow BA'0$.

$\rightarrow \rightarrow AB \rightarrow \rightarrow A \rightarrow BC \rightarrow AC = \rightarrow \rightarrow \rightarrow 11A \rightarrow \rightarrow \rightarrow B \rightarrow \rightarrow BCCA'0$

$= \rightarrow \rightarrow 1A \rightarrow \rightarrow \rightarrow B \rightarrow \rightarrow BCCA'0$.

Nous devons alors considérer la formule $\rightarrow B \rightarrow \rightarrow BCC$.

$\rightarrow BC = \rightarrow \rightarrow 1B \rightarrow \rightarrow CB'0$.

$\rightarrow \rightarrow BCC = \rightarrow \rightarrow 1CB \rightarrow \rightarrow \rightarrow CCB'0 = \rightarrow \rightarrow CB \rightarrow \rightarrow \rightarrow CCB'0$.

$\rightarrow B \rightarrow \rightarrow BCC = \rightarrow \rightarrow \rightarrow 0CB \rightarrow \rightarrow \rightarrow 1 \rightarrow CCB'0 = \rightarrow \rightarrow 1B \rightarrow \rightarrow \rightarrow CCB'0$.

Nous devons considérer maintenant la formule $\rightarrow CC$.

$\rightarrow CC = \rightarrow \rightarrow \rightarrow 00C \rightarrow \rightarrow \rightarrow 11C'0 = \rightarrow \rightarrow 1C \rightarrow \rightarrow 1C'0 \equiv 1$

Donc :

$\rightarrow B \rightarrow \rightarrow BCC = \rightarrow \rightarrow 1B \rightarrow \rightarrow 1B'0 \equiv 1$

Et : $\rightarrow \rightarrow AB \rightarrow \rightarrow A \rightarrow BC \rightarrow AC = \rightarrow \rightarrow 1A \rightarrow \rightarrow 1A'0 \equiv 1$

Un théorème du système de Kleene pourra être vérifié avec cette méthode sans établir de suite de formules mais systématiquement, par exemple :
$C \rightarrow C$ ou $\rightarrow CC$ qui a été démontré dans le cas de l'axiome précédent.

Les règles de déduction peuvent se vérifier aussi facilement, par exemple :

$A, A \to B \vdash B$ s'écrit sous la forme $(A \wedge (A \to B)) \to B$ et avec la notation utilisée $\to \to \to \to ABA'OB$.

Or :

$\to AB = \to \to 1A \to \to BA'0$.

$\to \to ABA' = \to \to \to 11A \to \to \to BOA'0 = \to \to 1A \to \to B'A'0$.

$\to \to \to ABA'0 = \to \to \to 10A \to \to \to B'0A'0 = \to \to 0A \to \to BA'0$.

$\to \to \to \to AB'0B = \to \to \to 0BA \to \to \to BBA'0 = \to \to 1A \to \to \to BBA'0$.

Et comme nous avons vu que : $\to BB \equiv 1$.

$\to \to \to \to ABA'0B = \to \to 1A \to \to 1A'0 \equiv 1$.

La règle de déduction s'identifie bien à 1.

De même écrivons le théorème de la déduction sous cette forme :

$$\begin{cases} \Gamma, A \vdash B \text{ s'écrit } (\Gamma \wedge A) \to B \text{ ou } \to \to \to \Gamma A'0B. \\ \Gamma \vdash A \to B \text{ s'écrit } \Gamma \to (A \to B) \text{ ou } \to \Gamma \to AB. \end{cases}$$

Il est alors très facile de le démontrer en prouvant que les deux expressions sont identiques.

En effet :

$\to \Gamma A' = \to \to \to \Gamma 1A \to \to \to \Gamma 0A'0 = \to \to 1A \to \to \Gamma A'0$.

$\to \to \Gamma A'0 = \to \to \to 10A \to \to \to \Gamma'0A'0 = \to \to 0A \to \to \Gamma A'0$.

$\to \to \to \Gamma A'0B = \to \to \to 0BA \to \to \to \Gamma BA'0 = \underline{\to \to 1A \to \to \to \Gamma BA'0}$

D'autre part :

$\to \Gamma \to AB = \to \to \to \Gamma 1A \to \to \to \Gamma BA'0 = \underline{\to \to 1A \to \to \to \Gamma BA'0}$.

<u>Remarque</u> : au lieu de la forme lexicographique pour l'opérateur implication, nous aurions aussi bien pu utiliser la forme lexicographique en ni ou en somme et produit (c'est d'ailleurs ce que nous ferons au chapitre II).

IV - Système de Wang :

Ce système mérite une mention spéciale car il propose un algorithme de reconnaissance systématique.

Opérateurs employés : $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$, $\Longrightarrow$.

Implication généralisée : l'expression fondamentale dans ce système est représentée par un séquent $\pi \Longrightarrow \rho$ où π et ρ sont des chaînes de formules pouvant être vides.

Pour montrer qu'une expression est identiquement vraie il faudra se ramener à $\pi \Longrightarrow \rho$, π et ρ contenant une formule commune. En effet, en algèbre de Boole ceci s'écrit : $\pi'_1 + \ldots + \pi'_n + \rho_1 + \ldots + \rho_m$ et sera identique à 1 quelque soit les valeurs de $\pi_1, \ldots, \pi_n, \rho_1, \ldots, \rho_m$ si et seulement si une variable est à la fois sous forme directe et complémentée :
$P = \pi_i = \rho_j$.

Les formules identiques à 1 devront donc toutes se ramener systématiquement à la forme :

$$\boxed{P, \pi \Longrightarrow P, \rho} \qquad (4)$$

Règles permettant de se ramener à une telle forme :
Si Φ et ψ sont des formules uniques, π, ρ, ξ, θ des chaînes de formules :

1) Règles sur l'opérateur $\neg$:
$\Phi, \xi \Longrightarrow \theta, \rho \vdash \xi \Longrightarrow \theta, \neg\Phi, \rho$.
$\theta, \rho \Longrightarrow \pi, \Phi \vdash \theta, \neg\Phi, \rho \Longrightarrow \pi$.

2) Règles sur l'opérateur $\wedge$:
$(\xi \Longrightarrow \theta, \Phi, \rho), (\xi \Longrightarrow \theta, \psi, \rho) \vdash \xi \Longrightarrow \theta, \Phi \wedge \psi, \rho$.
$\theta, \Phi, \psi, \rho \Longrightarrow \pi \vdash \theta, \Phi \wedge \psi, \rho \Longrightarrow \pi$.

3) Règles sur l'opérateur $\vee$:

$\xi \Rightarrow \theta,\ \Phi,\ \psi,\ \rho \vdash \xi \Rightarrow \theta,\ \Phi \vee \psi,\ \rho.$

$(\theta,\ \Phi,\ \rho \Rightarrow \pi),\ (\theta,\ \psi,\ \rho \Rightarrow \pi) \vdash \theta,\ \Phi \vee \psi,\ \rho \Rightarrow \pi.$

4) Règles sur l'opérateur $\rightarrow$:

$\xi,\ \Phi \Rightarrow \theta,\ \psi,\ \rho \vdash \xi \Rightarrow \theta,\ \Phi \rightarrow \psi,\ \rho.$

$(\theta,\ \psi,\ \rho \Rightarrow \pi),\ (\theta,\ \rho \Rightarrow \pi,\ \Phi) \vdash \theta,\ \Phi \rightarrow \psi,\ \rho \Rightarrow \pi.$

5) Règles sur l'opérateur $\leftrightarrow$:

$(\Phi,\ \xi \Rightarrow \theta,\ \psi,\ \rho),\ (\psi,\ \xi \Rightarrow \theta,\ \Phi,\ \rho) \vdash \xi \Rightarrow \theta,\ \Phi \leftrightarrow \psi,\ \rho.$

$(\Phi,\ \psi,\ \theta,\ \rho \Rightarrow \pi),\ (\theta,\ \rho \Rightarrow \pi,\ \Phi,\ \psi) \vdash \theta,\ \Phi \leftrightarrow \psi,\ \rho \Rightarrow \pi.$

Ces règles utilisées à rebours, nous permettent d'éliminer systématiquement chaque opérateur contenu initialement dans la formule étudiée. Car si nous considérons ces règles en algèbre de Boole, nous nous apercevons :

1) Qu'elles sont basées sur le fait que pour exprimer un séquent en algèbre de Boole nous écrivons une somme de variables sous forme complémentée, si elles sont à gauche de $\Rightarrow$, ou directe si elles sont à droite de $\Rightarrow$.

2) Qu'elles permettent d'écrire la formule étudiée sous forme <u>seulement de sommes</u> de variables complémentées ou non.

En effet :

- Les règles à un antécédent sont telles que la formule, contenant l'opérateur à éliminer, s'exprime en algèbre de Boole directement sous forme d'une somme de variables :

  1) $\Phi \wedge \psi$ placée à gauche de $\Rightarrow$ devient $(\Phi\psi)' = \Phi' + \psi'$ d'où la possibilité de remplacer $\Phi \wedge \psi$ par $\Phi,\ \psi$ toujours à la gauche de $\Rightarrow$.

  2) $\Phi \vee \psi$ placée à droite de $\Rightarrow$ devient $\Phi + \psi$ d'où la possibilité de remplacer $\Phi \vee \psi$ par $\Phi,\ \psi$ toujours à droite de $\Rightarrow$.

3) $\Phi \rightarrow \psi$ placée à droite de $\Longrightarrow$ devient $\Phi' + \psi$ d'où la possibilité de remplacer $\Phi \rightarrow \psi$ par Φ à gauche de $\Longrightarrow$ et de garder ψ à droite.

- Les règles à deux antécédents permettent d'éliminer les produits en transformant une somme contenant un produit de variables en deux sommes contenant d'une part une variable de ce produit et d'autre part toutes les variables de la somme initiale. Alors pour vérifier que le séquent initial est identique à 1, il faut et il suffit de vérifier que les deux nouveaux séquents le sont car pour que : $\Phi\psi = 1$ il faut et il suffit que $\Phi = \psi = 1$.

D'où l'élimination des opérateurs :

1) $\wedge$, $\Phi \wedge \psi$ étant placée à droite de $\Longrightarrow$ devient $\Phi\psi$ d'où remplacement du séquent initial contenant $\Phi \wedge \psi$ par deux séquents contenant l'un Φ, l'autre ψ à droite de $\Longrightarrow$.

2) $\vee$, $\Phi \vee \psi$ étant placée à gauche de $\Longrightarrow$ devient $(\Phi + \psi)' = \Phi'\psi'$ d'où remplacement du séquent initial contenant $\Phi \vee \psi$ par deux séquents contenant l'un Φ, l'autre ψ à gauche de $\Longrightarrow$.

3) $\rightarrow$, $\Phi \rightarrow \psi$ étant placée à gauche de $\Longrightarrow$ devient $(\Phi' + \psi)' = \Phi\psi'$ d'où remplacement du séquent contenant $\Phi \rightarrow \psi$ par deux séquents l'un contenant Φ à droite de $\Longrightarrow$, l'autre contenant ψ à gauche de $\Longrightarrow$.

4) $\longleftrightarrow$, $\Phi \leftrightarrow \psi$ à droite de $\Longrightarrow$ devient $\Phi\psi + \Phi'\psi' = (\Phi' + \psi)(\Phi + \psi')$.
$\Phi \leftrightarrow \psi$ à gauche de $\Longrightarrow$ devient $(\Phi' + \psi')(\Phi + \psi)$.

D'où pour chaque cas le remplacement du séquent unique par deux séquents permettant de faire disparaître le produit, ces deux séquents contiennent Φ et ψ placés convenablement par rapport à $\Longrightarrow$.

Finalement ces règles permettent d'écrire toute formule sous la forme de séquents $\pi \Longrightarrow \rho$. La formule initiale est d'ailleurs égale au produit de ces séquents car les règles permettant d'obtenir deux séquents à partir d'un seul ne sont qu'une application de la distributivité de l'addition booléenne par rapport à la multiplication, en effet :

$\xi' + \theta + \Phi\psi + \pi = (\xi' + \theta + \Phi + \pi)(\xi' + \theta + \psi + \pi)$, c'est-à-dire $(\xi \Longrightarrow \theta,\ \Phi \wedge \psi,\ \pi) = (\xi \Longrightarrow \theta,\ \Phi,\ \pi)(\xi \Longrightarrow \theta,\ \psi,\ \pi)$. De plus l'écriture employant l'opérateur $\Longrightarrow$ permet de considérer toutes les variables sous forme directe car il suffit de faire passer une variable complémentée de l'autre côté de cet opérateur pour l'avoir sous forme directe (règles sur l'opérateur $\neg$). En langage d'algèbre de Boole la méthode de Wang revient donc à écrire toute expression sous forme d'un produit de sommes de variables.

<u>Méthode d'identification à 1 de Wang</u> :

Pour obtenir la forme (4) nous cherchons dans la formule initiale l'opérateur de plus grande portée, nous l'éliminons par la règle correspondante. Nous faisons de même pour l'opérateur de plus grande portée dans la ou les nouvelles expressions obtenues et ceci jusqu'à l'élimination de tous les opérateurs si nécessaire. L'expression primitive sera identique à 1 si toutes les nouvelles expressions obtenues contiennent une même lettre de part et d'autre de $\Longrightarrow$.

Nous remarquons que pour qu'une expression initiale ou intermédiaire puisse être identique à 1 il faut, évidemment, qu'elle contienne au moins deux fois une même lettre car aucune règle ne permet de mettre une lettre de part et d'autre de $\Longrightarrow$. Donc dès que nous rencontrerons un séquent ne remplissant pas cette condition, il sera inutile d'éliminer les opérateurs restants et nous pourrons en déduire immédiatement que la formule initiale n'est pas identique à 1.

V - <u>Reconnaissance systématique d'expressions identiques à 0</u> :

Nous n'avons, dans ce qui précède, considéré que des expressions identiquement vraies. Or il peut être aussi intéressant de vérifier de même qu'une formule est soit identique à zéro soit ni identique à 0 ni identique à 1. La méthode de Wang transforme la formule initiale en un produit de sommes et sous cette forme il n'est pas possible de voir systématiquement que nous avons une identité à 0. Il faudrait donc, si la formule f considérée n'est pas identique à 1, étudier $\neg f$. La transformation qu'aurait déjà subit f serait inutile si nous arrivions à la conclusion que $\neg f$ est identiquement vraie. La méthode de l'opérateur implication nous permet de vérifier qu'une expression est identique à zéro sans pour cela prendre son complément. En effet f est identiquement nulle si et seulement si $f_0 \equiv f_1 \equiv 0$.

Donc dans le cas d'une expression identiquement nulle, en employant les mêmes règles que celles qui nous permettent d'identifier une expression à 1 à l'aide de l'opérateur implication, nous devons obtenir des expressions intermédiaires et finales de la forme :

$$\boxed{0 \equiv \rightarrow \rightarrow 0x \rightarrow \rightarrow 0x'0.}$$

Si l'expression de départ n'est identique ni à 1 ni à 0, les coefficients f_0 et f_1 seront soit des constantes (mais nous aurons $f_0 \neq f_1$) soit des variables. Dans ce cas ces variables se rencontrent dans chaque expression intermédiaire, en ordre lexicographique ; en lisant de gauche à droite nous obtenons l'ordre inverse de celui dans lequel les variables ont été utilisées.

Considérons des exemples illustrant le cas de formules non identiques à 1 :

a $\rightarrow \rightarrow \rightarrow \rightarrow AB'0A0$.

$\rightarrow AB' = \rightarrow \rightarrow \rightarrow 0B'A \rightarrow \rightarrow \rightarrow 1B'A'0 = \rightarrow \rightarrow 1A \rightarrow \rightarrow B'A'0$.

$\rightarrow \rightarrow AB'0 = \rightarrow \rightarrow 0A \rightarrow \rightarrow BA'0$.

$\rightarrow \rightarrow \rightarrow AB'0A = \rightarrow \rightarrow 1A \rightarrow \rightarrow 1A'0$.

$\rightarrow \rightarrow \rightarrow \rightarrow AB'0A0 = \rightarrow \rightarrow 0A \rightarrow \rightarrow 0A'0 \equiv 0$.

b $\rightarrow \rightarrow BA' \rightarrow \rightarrow \rightarrow \rightarrow CB'0A0$.

$\rightarrow \rightarrow CB' = \rightarrow \rightarrow \rightarrow CB'A \rightarrow \rightarrow \rightarrow CB'A'0$.

$\rightarrow \rightarrow CB'0 = \rightarrow \rightarrow \rightarrow \rightarrow CB'0A \rightarrow \rightarrow \rightarrow \rightarrow CB'0A'0$.

$\rightarrow \rightarrow \rightarrow CB'0A = \rightarrow \rightarrow \rightarrow CB'A \rightarrow \rightarrow 1A'0$.

$\rightarrow \rightarrow \rightarrow \rightarrow CB'0A0 = \rightarrow \rightarrow \rightarrow \rightarrow CB'0A \rightarrow \rightarrow 0A'0$.

$\to BA' = \to\to 1A \to\to B'A'O.$

$\to\to BA' = \to\to\to\to CB'OAO = \to\to\to\to CB'OA \to\to BA'O.$

Si nous considérons $\to\to CB'O$ par rapport à B :

$\to CB' = \to\to 1B \to\to CB'O.$

$\to\to CB'O = \to\to OB \to\to CB'O.$

Et $\underline{\to\to BA' \to\to\to\to CB'OAO = \to\to\to\to OB \to\to CB'OA \to\to BA'O}.$

VI - <u>Comparaison des méthodes de Wang et de l'opérateur implication</u> :

a) La méthode de Wang pourrait paraître plus systématique que celle de l'opérateur implication car elle traite l'expression à identifier à 1 immédiatement sous sa forme primitive, alors, que pour utiliser la deuxième méthode, il faut tout d'abord écrire la formule donnée avec l'opérateur unique considéré. Mais ceci peut se faire également d'une manière systématique et être programmé.

b) Si la formule étudiée n'est ni identique à 0 ni identique à 1, la méthode de Wang nous permettra de l'écrire sous forme d'un produit de sommes de variables complémentées ou non. La méthode de l'opérateur implication nous permettra d'obtenir la formule sous forme lexicographique par rapport à un opérateur donné. D'autre part cette méthode nous indique si la formule étudiée ou sa négation est un théorème ou, si ni la formule considérée ni sa négation ne sont des théorèmes. La méthode de Wang nous permet seulement de conclure si la formule donnée est un théorème ou non.

c) Une formule logique étant donnée, pour démontrer si elle est un théorème ou non, la méthode de Wang consiste à éliminer systématiquement, l'un après l'autre, tous les opérateurs contenus dans cette formule. Si celle-ci n'est pas un théorème dès que nous rencontrons un séquent ne contenant pas deux fois une même lettre, il est inutile d'éliminer les opérateurs restants car nous pouvons en déduire immédiatement que la formule initiale n'est pas valide.

Mais Wang n'utilise pas cette possibilité dans son programme et ne s'arrête que lorsque les séquents obtenus ne contiennent plus aucun opérateur. Nous appliquerons donc, les règles de déduction de Wang, au moins autant de fois que la formule contient d'opérateurs (un opérateur étant compté autant de fois qu'il apparaît dans la formule).

La méthode de l'opérateur implication revient à considérer la formule donnée comme une fonction booléenne des différentes propositions élémentaires qu'elle contient. Nous cherchons les coefficients de Lagrange par rapport à la première lettre dans l'ordre lexicographique. Si ces coefficients sont des constantes ou si l'un deux est égal à une variable, nous nous arrêtons immédiatement, sinon nous continuons par rapport à la deuxième lettre, etc... Nous appliquons donc les règles, nous permettant d'obtenir les coefficients par rapport à une lettre quelconque, au plus autant de fois que la formule étudiée contient de lettres différentes.

D'autre part, avec la méthode de l'opérateur implication, nous cherchons à chaque pas deux coefficients. Donc quand nous considérons ces coefficients par rapport à la lettre suivante nous devons étudier deux nouvelles formules si nécessaire (c'est-à-dire si aucune n'est égale à une constante ou une variable) au lieu d'une. Avec la méthode de Wang nous étudions, au pas suivant, deux séquents au lieu d'un si l'opérateur éliminé, au pas considéré est celui d'équivalence ou suivant la position des opérateurs $\wedge$, $\vee$, $\rightarrow$ par rapport à $\Longrightarrow$.

Si nous considérons uniquement le fait qu'une formule est un théorème ou non et le nombre de pas nécessaires pour arriver au résultat, dans le cas général, la méthode de l'opérateur implication sera la plus intéressante si la formule étudiée contient moins de lettres que d'opérateurs, dans le cas contraire celle de Wang sera meilleure. Dans le cas où le nombre de lettres est égal au nombre d'opérateurs, la méthode de l'opérateur implication pourra encore être la plus intéressante car nous n'utiliserons pas obligatoirement toutes les lettres alors que la méthode de Wang nous oblige à éliminer tous les opérateurs.

Remarquons, de plus, que la seule possibilité pour obtenir une formule contenant plus de lettres que d'opérateurs est d'avoir n lettres différentes et (n - 1) opérateurs, ce qui sera assez rare, la formule considérée ne sera d'ailleurs pas un théorème.

d) Etudions quelques exemples avec les deux méthodes.

1) $\neg (P \vee Q) \Rightarrow \neg P$ (3 opérateurs, 2 lettres)

Méthode de Wang :

$\Rightarrow \neg P, P \vee Q$ (1)
$P \Rightarrow P \vee Q$ (2)
$P \Rightarrow P, Q$ (3)

Méthode de l'opérateur implication :

$\neg (P \vee Q) \Rightarrow \neg P$ s'écrit $(((P \to 0) \to Q) \to 0) \to (P \to 0)$

$$\left.\begin{array}{l} f_1 = (((1 \to 0) \to Q) \to 0) \to (1 \to 0) = 1 \\ f_0 = (((0 \to 0) \to Q) \to 0) \to (0 \to 0) = 1 \end{array}\right\} (1)$$

Il suffit donc de considérer la première lettre.

2) $\Rightarrow (\neg P \wedge \neg Q) \to (P \leftrightarrow Q)$ (5 opérateurs, 2 lettres)

Méthode de Wang :

$\neg P \wedge \neg Q \Rightarrow P \leftrightarrow Q$ (1)
$\neg P, \neg Q \Rightarrow P \leftrightarrow Q$ (2)
$\neg Q \Rightarrow P \leftrightarrow Q, P$ (3)
$\Rightarrow P \leftrightarrow Q, P, Q$ (4)
$P \Rightarrow Q, P, P$ $\qquad$ $Q \Rightarrow P, P, Q$ (5)

Méthode de l'opérateur implication :

$(\neg P \wedge \neg Q) \to (P \leftrightarrow Q)$ s'écrit avec l'opérateur implication seul :

$$(((P \to 0) \to Q) \to 0) \to (((P \to Q) \to ((Q \to P) \to 0)) \to 0)$$

$$\left.\begin{aligned} f_1 &= (((1 \to 0) \to Q) \to 0) \to (((1 \to Q) \to ((Q \to 1) \to 0)) \to 0) = 1 \\ f_0 &= (((0 \to 0) \to Q) \to 0) \to (((0 \to Q) \to ((Q \to 0) \to 0)) \to 0) = (Q \to 0) \to (((Q \to 0) \to 0) \to 0) \end{aligned}\right\} (1)$$

$$\left.\begin{aligned} f_{01} &= (1 \to 0) \to (((1 \to 0) \to 0) \to 0) = 1 \\ f_{00} &= (0 \to 0) \to (((0 \to 0) \to 0) \to 0) = 1. \end{aligned}\right\} (2).$$

3) $P \vee (Q \vee R) \Rightarrow P \vee Q, R$ (3 opérateurs, 3 lettres).

Méthode de Wang :

(1) $P \Rightarrow P \vee Q, R$

(2) $P \Rightarrow P, Q, R$

$$\begin{array}{ccl} & Q \vee R \Rightarrow P \vee Q, R & (1)' \\ Q \Rightarrow P \vee Q, R & R \Rightarrow P \vee Q, R & (2)' \\ Q \Rightarrow P, Q, R & R \Rightarrow P, Q, R & (3)' \end{array}$$

Méthode de l'opérateur implication :

La formule étudiée s'écrit avec l'opérateur implication :

$$((P \to 0) \to ((Q \to 0) \to R)) \to ((((P \to 0) \to Q) \to 0) \to R)$$

$$\left.\begin{aligned} f_1 &= ((1 \to 0) \to ((Q \to 0) \to R)) \to ((((1 \to 0) \to Q) \to 0) \to R) = 1 \\ f_0 &= ((0 \to 0) \to ((Q \to 0) \to R)) \to ((((0 \to 0) \to Q) \to 0) \to R) = ((Q \to 0) \to R) \to ((Q \to 0) \to R) \end{aligned}\right\} (1)$$

$$\left.\begin{aligned} f_{01} &= ((1 \to 0) \to R) \to ((1 \to 0) \to R) = 1 \\ f_{00} &= ((0 \to 0) \to R) \to ((0 \to 0) \to R) = R \to R \end{aligned}\right\} (2)$$

$$\left.\begin{aligned} f_{001} &= 1 \to 1 = 1 \\ f_{000} &= 0 \to 0 = 1 \end{aligned}\right\} (3)$$

Remarque : il serait aussi intéressant de comparer ces deux méthodes à l'aide du temps nécessaire, sur machine, pour démontrer si une formule donnée est vraie ou non. Mais ceci est difficile, la méthode de Wang ayant été programmée pour IBM 704 et le programme, correspondant à la méthode de l'opérateur implication, ayant été utilisé sur IBM 7044.

VII - Programme correspondant à la méthode de l'opérateur implication :

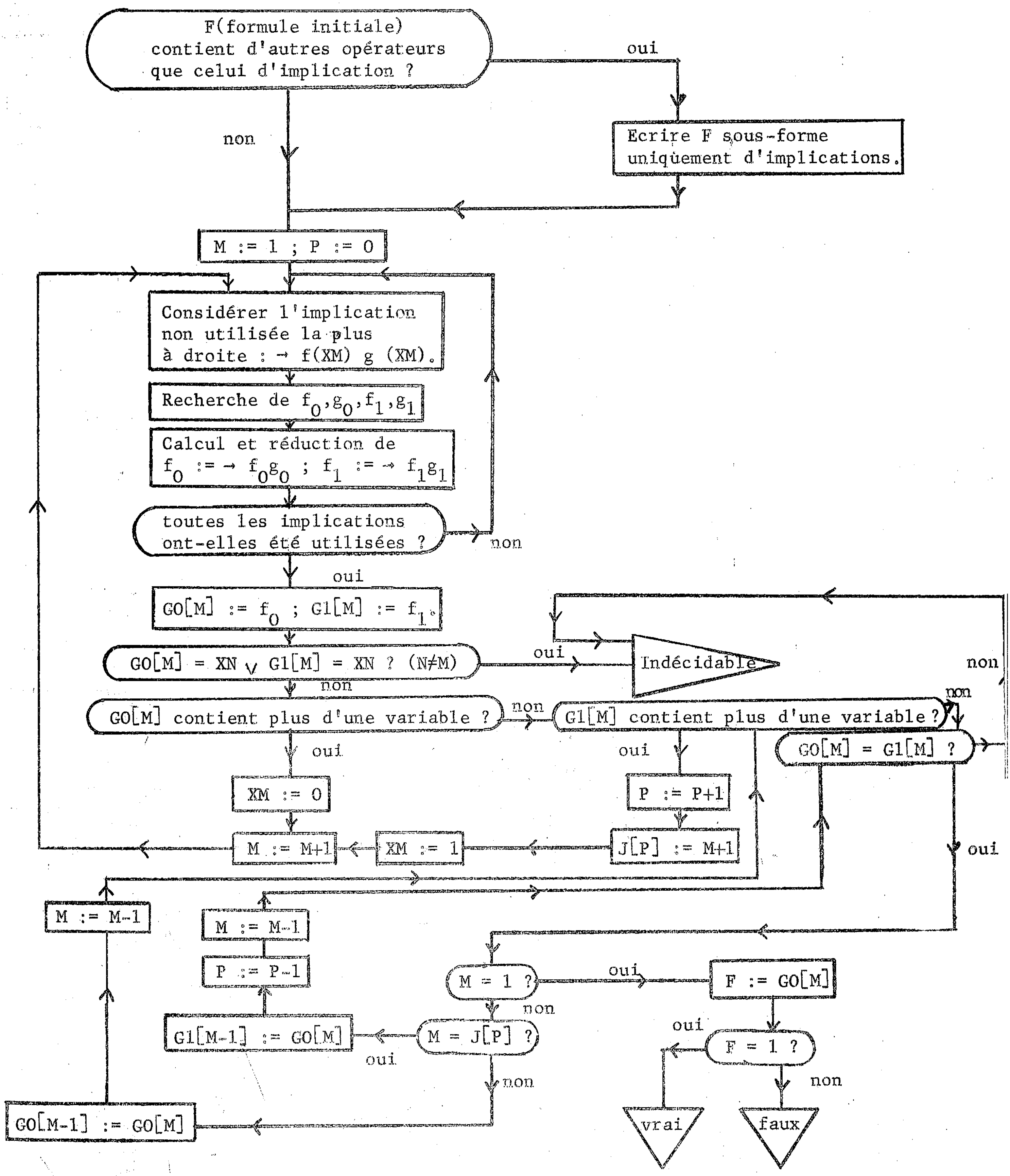
Nous pouvons à partir de la méthode utilisant l'opérateur implication et à l'aide de l'organigramme ci-après, écrire une procédure ALGOL qui nous indiquera, si, une formule f de propositions donnée sous sa forme primitive est vraie, fausse ou indécidable (c'est-à-dire telle que ni f ni $\neg$ f ne soient vraies).

Nous écrirons cette procédure pour des formules contenant que des variables directes, une variable complémentée A' sera remplacée par $\rightarrow$ A0.

Cette procédure utilise elle-même deux autres procédures, la première nous permettant d'obtenir les coefficients d'une lettre quelconque par rapport à celle considérée pour écrire la forme lexicographique, la deuxième nous donnant les coefficients d'une implication de la forme $\rightarrow$ fg.

L'ensemble de ces procédures comprend 1 115 unités syntaxiques auxquelles viennent s'ajouter 1 023 unités syntaxiques, correspondant à une procédure TRANSFORM nous permettant d'obtenir une formule donnée sous une forme contenant uniquement l'opérateur implication, si nécessaire.

Organigramme :



Ce programme, dont le listage suit, a été utilisé sur IBM 7044.

```
Début
  procédure  X(M, K, FO, F1) ; valeur K ; entier K, M, FO, F1 ;
  début  si K=M alors  début  FO := 0 ; F1 := 1  fin
                 sinon  FO := F1 := K
  fin ;
  procédure  I(X, Y, RO, R1, M, K1, K2, Q, T, C);
                 entier M, K1, K2, Q ; entier  tableau RO, R1, T ;
                 procédure  X, Y ; booleen C ;
  début  entier  procédure F(A, B) ; valeur A, B ; entier A, B ;
         début  si  A = 0 ∨ B = 1  alors  F := 1
                  sinon  si  A = 1  alors  F := B  sinon  F := A+B
         fin ;
         aiguillage  AIG := E1, E2, E3, E4 ; entier  J ;
         booleen  tableau  B[1:4] ;
         B[1] := K1 ≥ 0 ∧ K2 ≥ 0 ; B[2] := K1 < 0 ∧ K2 ≥ 0 ;
         B[3] := K1 ≥ 0 ∧ K2 < 0 ; B[4] := vrai ;
         pour  J := 1, 2, 3, 4  faire  si  B[J] alors  allera AIG[J] ;
      E1 : X(M, K1, RO[1], R1[1]) ; Y(M, K2, RO[2], R1[2]) ;
         allera  TERM ;
      E2 : RO[1] := T[0,1] ; R1[1] := T[0,2] ; Y(M, K2, RO[2], R1[2]) ;
         allera  TERM ;
      E3 : RO[2] := T[0,1] ; R1[2] := T[0,2] ; X(M, K1, RO[1], R1[1]) ;
         allera  TERM ;
      E4 : RO[1] := T[0,1] ; RO[2] := T[Q,1] ; R1[1] := T[0,2] ; R1[2] := T[Q,2] ;
         Q := Q-1 ;
    TERM : T[0,1] := F(RO[1], RO[2]) ; T[0,2] := F(R1[1], R1[2]) ;
         si  C  alors  début  T[Q+1,1] := T[0,1] ; T[Q+1,2] := T[0,2] fin
  fin ;
```

```
procédure  TRANSFORM(KI, OP, CI, K, KK, C, TO, T) ;
           entier T, TO ; entier tableau KI, OP, K, KK ;
           booleen tableau CI, C ;
début  entier  R, M, L, Q, N1, N2 ; entier tableau S[O,TO] ;
       pour R := 1 pas 2 jusqua 2 X TO faire
       pour M := 1,2 faire KI[R,M] := EDONNEE ;
       pour R := 1 pas 2 jusqua 2 X TO faire
       début  OP[R] := EDONNEE ;
              si KI[R,2] = -1 ∧ KI[R,1] = -1 alors début
              N2 := S[Q] - S[Q-1] ; N1 := T-N2 ; Q := Q-1 fin
              sinon si KI[R,2] = -1 alors N2 := T-S[Q]
              sinon si KI[R,1] = -1 alors N1 := T-S[Q] ;
              si OP[R] = 1 alors            début
              K[L+1] := KI[R,1] ; K[L+2] := KI[R,2] ;
              C[L+1] := CI[R] ; L := L+2 ; T := T+1 fin ;
              si OP[R] = 2        alors        début
              K[L+1] := KI[R,1] ; K[L+2] := 0 ; K[L+3] := -1 ; K[L+4]:=KI[R,2];
              C[L+3] := CI[R] ; L := L+4 ; T := T+2 ;
              si KI[R,2] = -1 alors C[L-5-2 X N1] := vrai fin ;
              si OP[R] = 3 alors début
              K[2 X (S[Q]+N2)+1] := KI[R,2] ; K[2 X (S[Q]+N2)+2] := 0 ;
              pour M := 1 pas 1 jusqua 2 X N1 faire
              K[2 X (S[Q]+N2)+2+M] := KK[2 X (S[Q]+N2)+M] ;
              K[L+3] := KI[R,1] ; K[L+4] := K[L+5] := -1 ; K[L+6] := 0 ;
              C[L+5] := CI[R] ; L := L+6 ; T := T+3 ;
              si KI[R,1] = -1 alors début C[L-5-2 X N1] := vrai ;
              si KI[R,2] = -1 alors C[L-7-2 X N1] := faux fin
                                           fin ;
              si OP[R] = 4     alors        début
              pour M := 1 pas 1 jusqua 2 X N1 faire
              K[2 X S[Q]+M] := KK[2 X (S[Q]+N2)+M] ;
```

```
                pour M := 1 pas 1 jusqua 2 × N2 faire
                K[2 × (S[Q]+N1)+M] := KK[2 × S[Q]+M] ;
                K[L+1]:=KI[R,2];K[L+2]:=KI[R,1];K[L+3]:=-1;K[L+4]:=0;C[L+3]:=vrai ;
                pour M := 1 pas 1 jusqua 2 × N1 faire
                K[L+4+2 × N2+M] := KK[2 × (S[Q]+N2)+M] ;
                M := 2 × (N2+N1) ; K[L+5+M] := KI[R,1] ; K[L+6+M] := KI[R,2] ;
                K[L+7+M] := K[L+8+M] := K[L+9+M] := -1 ; K[L+10+M] := 0 ;
                C[L+9+M] := CI[R] ; L := L+10+M ; T := 2 × T+5 ;
                si KI[R,1] =-1 ∧ KI[R,2] =-1 alors début
                C[2×(S[Q]+N2)-1]:= faux; C[2×(S[Q]+N1+2×N2)+3]:=C[2×(S[Q]+N1)-1]
                := vrai fin
                                                  fin ;
                pour M := 1 pas 1 jusqua L faire KK[M] := K[M] ;
                si CI[R] alors début Q := Q+1 ; S[Q] := T fin ;
                N1 := N2 := 0 ;
        fin
    fin ;
    procédure IMPLICATION (N, T, I, X, K, C) ; valeur N, T ;
              entier N, T ; procédure I, X ; entier tableau K ;
              booleen tableau C ;
              commentaire Une formule étant donnée sous forme d'implications,
              N représente le nombre de variables, T le nombre d'implications I.
              Les valeurs du tableau K seront données de telle sorte que les
              implications soient faites de droite à gauche ;
    début entier M, P, L, R, S, Q ;
          entier tableau G0,G1[1:N],J[0:N],R0E,R1E[1:2],TE[0:T,1:2],KK[1:2×T,1:N];
          M := 1 ; P := 0 ;
    RETOUR : R := 1 ; S := 2 ; Q := 0 ;
   RETOUR1 : I(X, X, R0E, R1E, M, K[R], K[S], Q, TE, C[R]) ;
          commentaire M représente la variable par rapport à laquelle est
          écrite la forme lexicographique.K[R] et K[S] représentent les numéros
          des variables sur lesquelles portent l'implication I. Ils seront né-
          gatifs si X est une implication. C[R] et Q nous permettent de garder
          le résultat de I s'il doit être utile ultérieurement.
```

```
        si C[R] alors Q := Q+1 ;
        si S < 2 × T alors début R := R+2 ; S := S+2 ; allera RETOUR1 fin ;
        GO[M] := TE[0,1] ; G1[M] := TE[0,2] ;
        ECRIRE (GO[M], G1[M]) ;
        si (M+1 ≤ GO[M]∧GO[M] ≤ 2×M+1)∨(M+1 ≤ G1[M]∧G1[M] ≤ 2×M+1)
        alors allera TERM ;
        si GO[M] = 0 ∨ GO[M] = 1 alors allera SUITE1 ;
        pour L := 1 pas 1 jusqua 2 × T faire KK[L,M] := K[L] ;
        pour L := 1 pas 1 jusqua 2 × T faire si K[L] = M alors K[L]:=0 ;
        M := M+1 ; allera RETOUR ;
  SUITE1 : si G1[M] = 0 ∨ G1[M] = 1 alors allera SUITE2 ;
        P := P+1 ; J[P] := M+1 ;
        pour L := 1 pas 1 jusqua 2 × T faire KK[L,M] := K[L] ;
        pour L := 1 pas 1 jusqua 2 × T faire si K[L] = M alors K[L]:=1 ;
        M := M+1 ; allera RETOUR ;
  SUITE2 : si GO[M] ≠ G1[M] alors allera TERM ;
           si M = 1 alors début si GO[M] = 1 alors ECRIRE (vrai)
                                sinon ECRIRE (faux) ; allera TERM1
                          fin ;
           si M = J[P] alors début G1[M-1] := GO[M] ; P := P-1 ;
                                   M := M-1 ; allera SUITE2
                             fin
           sinon début GO[M-1] := GO[M] ; M := M-1 ;
                       pour L := 1 pas 1 jusqua 2×T faire K[L]:=KK[L,M] ;
                       allera SUITE1 ;
                 fin ;
   TERM : ECRIRE ('('INDECIDABLE')') ;
  TERM1 :
  fin ;
  entier tableau KI[1 : 10, 1 : 2], K, KK, OP[1 : 10] ; entier T ;
  booleen talbeau CI, C [1 : 10]
  TRANSFORM (KI, OP, CI, K, KK, C, 4, T) ;
  IMPLICATION (2, T, I, X, K, C)
fin ;
```

Ce programme nous a permis de montrer la validité, en particulier, des formules suivantes :

$- \neg (X1 \vee X2) \rightarrow \neg X2$ (formule considérée pour activer, ci-dessus, les procédures).

Nous obtenons comme résultat :

1 1 (coefficients par rapport à X1)

VRAI

$-(X1 \rightarrow X2) \rightarrow ((X1 \rightarrow (X2 \rightarrow X3)) \rightarrow (X2 \rightarrow X3))$

(deuxième axiome du système de Kleene)

résultat :	1	10	(coefficients par rapport à X1)
	1	6	(coefficients par rapport à X2)
	1	1	(coefficients par rapport à X3)
	VRAI		

(Les coefficients différents de 1 sont représentés par la somme des numéros des variables apparaîssant dans la nouvelle formule obtenue ; l'emploi du langage ALGOL, uniquement, entraînant à considérer les variables sous forme numérique).

$-(X1 \wedge (X1 \rightarrow X2)) \rightarrow X2$ (règle de déduction du système de Kleene)

résultat :	1	4
	1	1
	VRAI	

$-((X3 \wedge X1) \rightarrow X2) \longleftrightarrow (X3 \rightarrow (X1 \rightarrow X2))$ (théorème de la déduction)

résultat :	1	20
	12	1
	1	1
	VRAI	

Dans les cas d'utilisation de la procédure TRANSFORM le travail total a duré 1 minute 13 secondes, la recherche de la validité ou non d'une formule durant de une à deux secondes. Dans le cas contraire le travail total a duré 49 secondes et la recherche de la validité a duré en moyenne 1 s.

- CHAPITRE II -

CALCUL DES PREDICATS UNAIRES EN LOGIQUE CLASSIQUE.

I - Ecriture des formules de prédicats en algèbre de Boole :

1) Symboles de prédicats :

En logique un prédicat $A(x_1,\ldots,x_n)$ est tel que les variables $x_1,\ldots,x_n$ prennent respectivement leurs valeurs dans un ensemble $(0,\ldots,k)$, lui-même ne pouvant être que vrai ou faux. En algèbre de Boole ce sera donc une fonction de n variables non booléennes en général (n pouvant être nul : dans ce cas nous avons une proposition c'est-à-dire une fonction constante), qui, pour les différentes valeurs données aux variables, prendra une des valeurs booléennes simples $A_0, A_1,\ldots,A_{(k+1)^n-1}$.

2) Formules de prédicats :

Si nous donnons des valeurs aux variables, la valeur de la fonction correspondante donnera une formule de prédicats.

En logique si A et B sont des formules, nous en obtenons de nouvelles à l'aide des opérateurs $\rightarrow, \wedge, \vee, \neg, \leftrightarrow$ et des deux nouveaux opérateurs $\forall x_i$ et $\exists x_i$. De même que pour le calcul des propositions, en algèbre de Boole, elles s'écriront :

$A \rightarrow B : A'+ B$; $A \wedge B : AB$; $A \vee B : A + B$; $\neg A = A'$; $A \leftrightarrow B = A'B'+ AB$.

Nous pouvons aussi exprimer, en algèbre de Boole, les formules $\forall x\ A(x)$ et $\exists x\ A(x)$ obtenues à partir des quantificateurs $\forall x$ et $\exists x$ en considérant leur interprétation :

- $\forall x\ A(x)$ est vraie, c'est-à-dire en algèbre de Boole prend la valeur 1, si pour tout x $A(x) = 1$ donc si $A_0 = A_1 = \ldots = A_k = 1$ ce qui peut s'exprimer par $A_0\ A_1 \ldots A_k = 1$ et $\forall x\ A(x)$ est fausse, c'est-à-dire en algèbre de Boole égale à zéro, si $A(x)$ a au moins une valeur nulle donc si $A_0\ A_1 \ldots A_k = 0$ et nous pouvons donc écrire :

$$\boxed{\forall x\ A(x) = A_0\ A_1 \ldots A_k.}$$

- De même $\exists x\ A(x) = 1$ si $A(x)$ prend au moins une fois la valeur 1 donc si $A_0 + A_1 + \ldots + A_k = 1$ et $\exists x\ A(x) = 0$ si $A(x)$ est nulle pour tout x c'est-à-dire si $A_0 + A_1 + \ldots + A_k = 0$ donc nous pouvons écrire :

$$\boxed{\exists x\ A(x) = A_0 + A_1 + \ldots + A_k.}$$

Nous noterons dans ce qui suit : $\forall x\ A(x) = \prod\limits_i A_i$; $\exists x\ A(x) = \sum\limits_i A_i$.

Ceci peut se généraliser évidemment à des prédicats à plusieurs variables. Par exemple :

$\forall x_1\ \exists x_2\ \forall x_3\ A(x_1, x_2, x_3)$ s'écrira $\prod\limits_{i_1=0}^{k}\ \sum\limits_{i_2=0}^{k}\ \prod\limits_{i_3=0}^{k}\ A_{i_1 i_2 i_3}$ mais de telles expressions sont peu maniables. Dans toute la suite nous considérerons uniquement des prédicats à une variable.

Les quantificateurs nous conduisent donc à introduire des sommes et produits. D'autre part les prédicats eux-mêmes peuvent s'écrire sous forme de sommes et de produits. Pour cela considérons des fonctions $\delta_i(x)$ telles que $\delta_i(x) = 0$ si $i \neq x$ et $\delta_i(x) = 1$ si $x = i$. Nous pouvons alors écrire :

$A(x) = A_0\, \delta_0(x) + \ldots + A_i\, \delta_i(x) + \ldots + A_k\, \delta_k(x) = \sum_i A_i\, \delta_i(x)$.

II - Reconnaissance systématique d'expressions identiquement vraies :

En raison de la présence de sommes et produits dans les expressions à manier, l'opérateur implication est peu commode et nous utiliserons comme opérateurs la somme, le produit et le complément.

1) Introduction de variables généralisées :

Si nous écrivons les formules de prédicats, comportant des quantificateurs, sous forme de sommes ou produits des différentes valeurs de l'expression $E(x)$ sur laquelle porte le quantificateur ($\exists x\, E(x) = \sum_i E_i$; $\forall x\, A(x) = \prod_i E_i$ où E_i peut être une expression booléenne quelconque de variables indicées A_i, B_i, $C_i \ldots$) et si nous écrivons tout prédicat $A(x) = \sum_i A_i\, \delta_i(x)$, nous pouvons considérer toute formule F de prédicats comme une fonction booléenne de sommes et produits de l'une des formes ci-dessus. Nous pouvons, de plus, toujours nous ramener à des expressions E_i qui soient sous forme de somme de produits et écrire $E_i = A_i\, B_i + A'_i\, C_i$ où B_i et C_i pourront être des constantes ou des expressions booléennes quelconques de variables indicées sous forme de somme de produits, et où A_i pourra être une constante ou une variable simple. Nous avons donc $\sum_i E_i = \sum_i A_i\, B_i + \sum_i A'_i\, C_i$. D'autre part nous pouvons aussi écrire $E_i = (A_i + C_i)(A'_i + B_i)$ et par conséquent $\prod_i E_i = \prod_i (A_i + C_i) \prod_i (A'_i + B_i)$. F peut alors être considérée comme une fonction booléenne d'expressions G de la forme : $\prod_i (A_i + B_i)$, $\prod_i (A'_i + B_i)$, $\sum_i A_i B_i$, $\sum_i A'_i B_i$.

Toutes ces expressions G peuvent s'exprimer en fonction de $\prod_i A_i$ ou $\sum_i A_i$ et d'expressions ne dépendant plus des A_i. En effet nous pouvons constater que :

- si $\prod_i A_i = 1$ c'est-à-dire si tous les A_i sont égaux à 1 : $\prod_i (A_i + B_i) = 1$, $\sum_i A'_i B_i = 0$.
- si $\prod_i A_i = 0$ c'est-à-dire si au moins un des A_i égale zéro : $\prod_i (A_i + B_i) = \prod_j B_j$, $\sum_i A'_i B_i = \sum_j B_j$ (les éléments repérés par l'indice j vérifiant $A_j = 0$).

Ce qui permet d'écrire :

$$\begin{cases} \prod_i (A_i + B_i) = \prod_i A_i + \prod_j B_j. \\ \sum_i A'_i B_i = \sum_j B_j \sum_i A'_i. \end{cases}$$

De même si nous considérons $\sum_i A_i B_i$ et $\prod_i (A'_i + B_i)$:

- si $\sum_i A_i = 0$, donc si tous les A_i sont nuls, alors $\sum_i A_i B_i = 0$, $\prod_i (A'_i + B_i) = 1$.
- si $\sum_i A_i = 1$, donc si au moins un des A_i est égal à 1, $\sum_i A_i B_i = \sum_k B_k$, $\prod_i (A'_i + B_i) = \prod_k B_k$ (les éléments repérés par l'indice k vérifiant $A_k = 1$).

Ce qui entraîne :

$$\begin{cases} \sum_i A_i B_i = \sum_i A_i \sum_k B_k. \\ \prod_i (A'_i + B_i) = \prod_k B_k + \prod_i A'_i. \end{cases}$$

Si nous considérons maintenant les expressions booléennes $\prod_i A_i$ et $\sum_i A_i$ comme de nouvelles variables X et Y, nous pourrons écrire toute formule F de prédicats sous-forme d'expression canonique par rapport à ces deux variables :

$$F\,(\prod_i A_i,\ \sum_i A_i,\ G) = F(X, Y, G) = F_0(Y, G)\ X' + F_1(Y, G)\ X.$$

$$F\,(X, Y, G) = (F_{00}(G)\ Y' + F_{01}(G)\ Y)\ X' + (F_{10}(G)\ Y' + F_{11}(G)\ Y)\ X.$$

En tenant compte que $X = \prod_i A_i$, $Y = \sum_i A_i$, $X' = \sum_i A'_i$ et $Y' = \prod_i A'_i$:

$$F\,(X, Y, G) = F_{00}(G) \prod_i A'_i + F_{01}(G) \sum_i A_i \sum_i A'_i + F_{11}(G) \prod_i A_i.$$

Donc pour que F soit identique à 1 (ou à 0) il faut et il suffit que :

$$F_{00} \equiv F_{01} \equiv F_{11} \equiv 1 \text{ (ou } F_{00} \equiv F_{01} \equiv F_{11} \equiv 0).$$

Remarque : la valeur de F_{10} est indifférente, ceci provient du fait qu'elle correspond aux valeurs des variables $\prod_i A_i = 1$ et $\sum_i A_i = 0$, valeurs incompatibles. Nous pourrions écrire F d'abord par rapport à $\sum_i A_i$ puis par rapport à $\prod_i A_i$. Nous aurions alors $X = \sum_i A_i$, $Y = \prod_i A_i$, $X' = \prod_i A'_i$, $Y' = \sum_i A'_i$ et ce serait le produit YX' qui deviendrait nul et par conséquent la valeur F_{01} qui serait indifférente.

Nous allons essayer d'étudier systématiquement toute formule de prédicats d'une manière analogue à celle utilisée pour les propositions. Nous avons à la place de fonctions de variables simples, des fonctions de variables généralisées de la forme G. Nous remarquons que ces variables généralisées peuvent être toutes considérées comme non complémentées car

$$(\prod_i E_i)' = \sum_i E'_i,\quad (\sum_i E_i)' = \prod_i E'_i.$$

2) Description de la méthode permettant de reconnaître qu'une formule de prédicats est identique à 1 :

a) Nous devons, tout d'abord, considérer les règles supplémentaires :

$$\sum_i \delta_i(x) = \sum_i \delta'_i(x) \equiv 1 \quad , \quad \prod_i \delta_i(x) = \prod_i \delta'_i(x) \equiv 0 \qquad (3)'$$

b) D'autre part il nous faut trouver systématiquement les coefficients G_{00}, G_{01}, G_{11} correspondant à une expression G par rapport à X et Y considérées. Si nous prenons $X = \prod_i A_i$, $Y = \sum_i A_i$, deux cas se présentent :

- G ne contient pas de variables indicées A_i

$$G_{00} = G_{01} = G_{11} = G.$$

- G contient des variables indicées A_i. G peut alors s'écrire $Z * H$ où Z représente X, Y, X' ou Y', $*$ une somme ou un produit et où H ne contient pas de variables indicées A_i. Donc :

$H_{00} = H_{01} = H_{11} = H$ et $G_{00} = Z_{00} * H$, $G_{01} = Z_{01} * H$, $G_{11} = Z_{11} * H$.

Z_{00}, Z_{01}, Z_{11} sont obtenues facilement en considérant que :

$\prod_i A_i = 0 \sum_i A'_i + 1 \prod_i A_i$. $\qquad (Z_{00} = Z_{01} = 0,\ Z_{11} = 1)$.

$\sum_i A'_i = 1 \sum_i A'_i + 0 \prod_i A_i$. $\qquad (Z_{00} = Z_{01} = 1,\ Z_{11} = 0)$.

$\sum_i A_i = 1 \prod_i A_i + \sum_i A_i \sum_i A'_i = 1 \sum_i A_i + 0 \prod_i A'_i$. $\qquad (Z_{00} = 0,\ Z_{01} = Z_{11} = 1)$.

$\prod_i A'_i = 0 \prod_i A_i + \prod_i A'_i \sum_i A'_i = 0 \sum_i A_i + 1 \prod_i A'_i$. $\qquad (Z_{00} = 1,\ Z_{01} = Z_{11} = 0)$.

c) Connaissant G_{00}, G_{01}, G_{11} de chaque variable généralisée G d'une formule F de prédicats nous obtiendrons les coefficients F_{00}, F_{01}, F_{11} par rapport à X et Y en employant deux expressions analogues à (2) qui nous donnait f_0 et f_1 d'une formule f de propositions. C'est-à-dire :

$$\left.\begin{aligned} G + K &= (G_{00} + K_{00})\, Y' + (G_{01} + K_{01})\, YX' + (G_{11} + K_{11})\, X \\ G\, K &= (G_{00}\, K_{00})\, Y' + (G_{01}\, K_{01})\, YX' + (G_{11}\, K_{11})\, X \end{aligned}\right\} \quad (2)'$$

et :

$$F_{00} = G_{00} * K_{00} \;;\; F_{01} = G_{01} * K_{01} \;;\; F_{11} = G_{11} * K_{11}.$$

Evidemment aucun des trois coefficients ne contiendra la variable indicée correspondant à X et Y mais ils dépendront d'expressions de la forme $\prod_i E_i$, $\sum_i E_i$, $\prod_j E_j$, $\sum_j E_j$, $\sum_k E_k$, $\prod_k E_k$; les ensembles I, J, K respectivement des indices i, j, k étant tels que $J \wedge K = \Phi$ et $J \vee K = I$.

Supposons, de nouveau, que $X = \prod_i A_i$ et $Y = \sum_i A_i$. Le coefficient F_{11} correspondant au cas où tous les $A_i = 1$ nous remplacerons $\prod_k E_k$ et $\sum_k E_k$ par $\prod_i E_i$ et $\sum_i E_i$, nous aurons alors une expression contenant uniquement l'indice i, l'indice j n'apparaîssant jamais dans ce coefficient. De même l'indice k n'apparaît jamais dans le coefficient F_{00} et comme nous sommes dans le cas où tous les $A_i = 0$ nous remplacerons $\prod_j E_j$ et $\sum_j E_j$ par $\prod_i E_i$ et $\sum_i E_i$. De plus nous pouvons écrire $\prod_i E_i = \prod_j E_j \prod_k E_k$ et $\sum_i E_i = \sum_j B_j + \sum_k B_k$ ce qui entraîne que le coefficient F_{01} dépend alors de deux couples de variables, $\prod_j E_j$ et $\sum_j E_j$ d'une part et $\prod_k B_k$, $\sum_k B_k$ d'autre part, ayant des valeurs indépendantes.

Les coefficients F_{00} et F_{11} sont des expressions analogues à F mais contenant la variable indicée A_i en moins. Nous les étudierons de même que F par rapport à B_i si nous considérons que les expressions G sont de la forme $\pi_i(B_i + E_i)$, $\pi_i(B'_i + E_i)$, $\Sigma_i B_i E_i$, $\Sigma_i B'_i E_i$ où B_i est maintenant une variable simple ou une constante et E_i une expression booléenne quelconque ne contenant pas les variables indicées A_i et B_i. Le coefficient F_{01} dépend d'expressions de la forme $\pi_j(B_j + E_j)$, $\pi_j(B'_j + E_j)$, $\Sigma_j B_j C_j$, $\Sigma_j B'_j C_j$, $\pi_k(B_k + C_k)$, $\pi_k(B'_k + C_k)$, $\Sigma_k B_k C_k$, $\Sigma_k B'_k C_k$, il faudra donc l'étudier par rapport aux variables $\pi_j B_j$ et $\Sigma_j B_j$ et ensuite les coefficients obtenus par rapport aux variables $\pi_k B_k$ et $\Sigma_k B_k$. Alors toutes les expressions comportant des variables indicées en B auront à leur tour disparu et nous étudierons les coefficients obtenus à partir de F_{00} et F_{01} soit par rapport à $\pi_i C_i$, $\Sigma_i C_i$ soit par rapport à $\pi_\ell C_\ell$, $\Sigma_\ell C_\ell$ et $\Sigma_m C_m$, $\pi_m C_m$ (si $B_\ell = 0$, $B_m = 1$), et les coefficients obtenus à partir de F_{01} soit par rapport à $\Sigma_\ell C_\ell$, $\pi_\ell C_\ell$ ou $\Sigma_m C_m$, $\pi_m C_m$ soit par rapport à $\pi_\ell C_{j\ell}$, $\Sigma_\ell C_{j\ell}$ ($B_\ell = 0$ et $A_j = 0$), $\pi_m C_{jm}$, $\Sigma_m C_{jm}$ ($B_m = 1$, $A_j = 0$), $\pi_m C_{km}$, $\Sigma_m C_{km}$($B_m = 1$, $A_k = 1$), $\pi_\ell C_{k\ell}$, $\Sigma_\ell C_{k\ell}$($B_\ell = 0$, $A_k = 1$) et nous continuerons ainsi jusqu'à ce que nous ayons épuisé toutes les lettres indicées si nécessaire.

Nous remarquons qu'après chaque nouvelle élimination d'une lettre indicée nous doublons le nombre d'indices des variables restantes quand nous nous occupons de coefficients analogues à F_{01}. Ceci nous permet cependant d'arriver au résultat en un nombre fini de pas, puisque à chaque pas nous éliminons toutes les apparitions d'une lettre quelque soit l'indice qui la concerne.

3) <u>Formule de départ non identique à 1</u> :

Si la formule de départ est identique à 0 la méthode décrite nous conduira évidemment au résultat de même que dans le cas des propositions.

Si elle n'est identique ni à 1 ni à 0 nous obtiendrons, évidemment, comme pour le calcul des propositions, la formule sous une forme lexicographique par rapport aux variables non indicées, l'ordre lexicographique obtenu étant encore l'inverse de l'ordre dans lequel les variables indicées ont été utilisées mais de plus dans certaines sous-expressions nous rencontrerons à nouveau un ordre lexicographique portant sur les différents indices de la même variable indicée.

4) Exemples d'utilisation de cette méthode :

a) Considérons tout d'abord un exemple où nous n'avons que des expressions G contenant soit A_i seule soit ne contenant pas A_i, il est possible dans ce cas d'étudier directement les coefficients F_{00}, F_{01}, F_{11} par rapport à $\prod_i B_i$ et $\sum_i B_i$ sans faire intervenir deux indices différents pour B.

$$\exists x(A(x) \rightarrow B(x)) \rightarrow (\forall x\ A(x) \rightarrow \exists x\ B(x)) \text{ c'est-à-dire :}$$

$$(\sum_i(A'_i + B_i))' + (\prod_i A_i)' + \sum_i B_i = \prod_i A_i \prod_i B'_i + \sum_i A'_i + \sum_i B_i$$

$$\prod_i A_i \prod_i B'_i = (0 \times \prod_i B'_i) \sum_i A'_i + (1. \prod_i B'_i) \prod_i A_i = 0 \sum_i A'_i + \prod_i B'_i \prod_i A_i.$$

$$\sum_i A'_i + \sum_i B_i = (1 + \sum_i B_i) \sum_i A'_i + (0 + \sum_i B_i) \prod_i A_i = 1 \sum_i A'_i + \sum_i B_i \prod_i A_i.$$

$$\prod_i A_i \prod_i B'_i + \sum_i A'_i + \sum_i B_i = 1 \sum_i A'_i + (\prod_i B'_i + \sum_i B_i) \prod_i A_i.$$

Dans cet exemple il est aussi inutile de considérer les coefficients obtenus par rapport à $\sum_i A_i$, ceux-ci ne dépendant plus de A_i. Nous avons donc $F_0 = F_{00} = F_{01} = 1$ et $F_{11} = \prod_i B'_i + \sum_i B_i$.

Nous allons étudier F_{11} par rapport à $\prod_i B_i$ et $\sum_i B_i$:

$$\prod_i B'_i + \sum_i B_i = (1 + 0) \prod_i B'_i + (0 + 1) \sum_i B_i \sum_i B'_i + (0 + 1) \prod_i B_i.$$

$$= 1 \prod_i B'_i + 1 \sum_i B_i \sum_i B'_i + 1 \prod_i B_i \equiv 1.$$

Et l'expression initiale est donc bien une formule identiquement vraie.

b) Soit un exemple contenant un prédicat non précédé de quantificateur :

$A(x) \to \exists x\ A(x)$ (Axiome du système de Kleene).

Nous pouvons l'étudier par rapport à l'opérateur implication ce qui donnera une écriture assez simple puisqu'il ne dépend que d'une variable.

$$\to \sum_i A_i \delta_i \sum_i A_i = \to\to\to\to\to 00 \sum_i A_i \to\to\to \sum_k \delta_k 1 \pi A'_i 0 \pi A_i \to\to\to \sum_k \delta_k 1 \sum_i A'_i 0.$$

$$= \to\to\to\to 1 \sum_i A_i \to\to 1 \pi_i A'_i\ 0 \pi_i A_i \to\to 1 \sum_i A'_i\ 0 \equiv 1.$$

c) $\pi_i(A'_i + B'_i) + \pi_i(A'_i + C'_i) + \sum_i A_i B_i$:

$$\pi_i(A'_i + B'_i) = 1 \pi_i A'_i + \pi_k B'_k \sum_i A_i \sum_i A'_i + \pi_k B'_k \pi_i A_i.$$

$$\pi_i(A'_i + B'_i) + \pi_i(A'_i + C'_i) = 1 \pi A'_i + (\pi_k B'_k + \pi_k C'_k) \sum_i A_i \sum_i A'_i + (\pi_k B'_k + \pi_k C'_k) \pi_i A_i.$$

$$\sum_i A_i B_i = 0 \pi_i A'_i + \sum_k B_k \sum_i A_i \sum_i A'_i + \sum_k B_k \pi_i A'_i.$$

$$\pi_i(A'_i + B'_i) + \pi_i(A'_i + C'_i) + \sum_i A_i B_i = 1 \pi_i A'_i + (\pi_k B'_k + \pi_k C'_k + \sum_k B_k) \sum_i A_i \sum_i A'_i + (\pi_k B'_k + \pi_k C'_k + \sum_k B_k) \pi_i A_i$$

Pour le coefficient F_{11} nous pourrions remplacer les indices k par des indices i mais ici ce n'est pas utile puisque seul le coefficient k apprait. Etudions $\pi_k B'_k + \pi_k C'_k + \sum_k B_k$ par rapport à $\pi_k B_k$ et $\sum_k B_k$:

$$\pi_k B'_k + \pi_k C'_k + \sum_k B_k = (1 + \pi_k C'_k + 0) \pi_k B'_k + (0 + \pi_k C'_k + 1) \sum_k B_k \sum_k B'_k + (0 + \pi_k C'_k + 1) \pi_k B_k.$$

$$= 1 \pi_k B'_k + 1 \sum_k B_k \sum_k B'_k + 1 \sum_k B_k \equiv 1$$

Et : $\pi_i(A'_i + B'_i) + \pi_i(A'_i + C'_i) + \sum_i A_i B_i \equiv 1.$

d) Considérons maintenant une expression identique ni à 1 ni à 0 :

$$\prod_i (A'_i + B'_i) + \prod_i (A'_i + C'_i) + \sum_i A_i B_i C_i .$$

$$\sum_i A_i B_i C_i = 0 \prod_i A'_i + \sum_k B_k C_k \sum_i A_i \sum_i A'_i + \sum_k B_k C_k \prod_i A'_i .$$

$$\prod_i (A'_i + B'_i) + \prod_i (A'_i + C'_i) + \sum_i A_i B_i C_i$$

$$= 1 \prod_i A'_i + (\prod_k B'_k + \prod_k C'_k + \sum_k B_k C_k) \sum_i A_i \sum_i A'_i + (\prod_k B'_k + \prod_k C'_k + \sum_k B_k C_k) \prod_i A_i .$$

$$= 1 \prod_i A'_i + (\prod_k B'_k + \prod_k C'_k + \sum_k B_k C_k) \sum_i A_i \sum_i A'_i + (\prod_i B'_i + \prod_i C'_i + \sum_i B_i C_i) \prod_i A_i .$$

Etudions $\prod_k B'_k + \prod_k C'_k + \sum_k B_k C_k$ par rapport à $\prod_k B_k$ et $\sum_k B_k$:

$$\prod_k B'_k + \prod_k C'_k + \sum_k B_k C_k = (1 + \prod_k C'_k + 0) \prod_k B'_k + (0 + \prod_k C'_k + \sum_m C_{km}) \sum_k B_k \sum_k B'_k + (0 + \prod_k C'_k + \sum_m C_{km}) \prod_k B_k .$$

$$= 1 \prod_k B'_k + (\prod_k C'_k + \sum_m C_{km}) \sum_k B_k \sum_k B'_k + (\prod_k C'_k + \sum_m C_{km}) \prod_k B_k .$$

$F_{11} = \prod_k C'_k + \sum_m C_{km}$ peut s'écrire $F_{11} = \prod_k C'_k + \sum_k C_k$ que nous trouverions évidemment identique à 1 en l'écrivant par rapport à $\prod_k C_k$ et $\sum_k C_k$.

Etudions $F_{01} = \prod_k C'_k + \sum_m C_{km} = \prod_m C'_{km} \prod_\ell C'_{k\ell} + \sum_m C_{km}$ par rapport à $\prod_\ell C_{k\ell}$ et $\sum_\ell C_{k\ell}$:

$$\prod_m C'_{km} \prod_\ell C'_{k\ell} + \sum_m C_{km} = (\prod_m C'_{km} + \sum_m C_{km}) \prod_\ell C'_{k\ell} + \sum_m C_{km} \sum_\ell C'_{k\ell} \sum_\ell C_{k\ell} + \sum_m C_{km} \prod_\ell C_{k\ell}$$

Etudions $\prod_m C'_{km} + \sum_m C_{km}$ par rapport à $\prod_m C_{km}$ et $\sum_m C_{km}$:

$$\prod_m C'_{km} + \sum_m C_{km} = 1 \prod_m C'_{km} + 1 \sum_m C_{km} + 1 \prod_m C_{km} \equiv 1 .$$

D'où la formule finale :

$$\prod_i (A'_i + B'_i) + \prod_i (A'_i + C'_i) + \sum_i A_i B_i C_i =$$

$$1 \prod_i A'_i + (1 \prod_k B'_k + (1 \prod_\ell C'_{k\ell} + \sum_m C_{km} \sum_\ell C'_{k\ell} \sum_\ell C_{k\ell} + \sum_m C_{km} \prod_\ell C_{k\ell}) \sum_k B_k \sum_k B'_k$$

$$+ 1 \prod_k B_k) \sum_i A_i \sum_i A'_i + (1 \prod_i B'_i + (1 \prod_\ell C'_\ell + \sum_m C_m \sum_\ell C'_\ell \sum_\ell C_\ell$$

$$+ \sum_m C_m \prod_\ell C_\ell) \sum_i B_i \sum_i B'_i + 1 \prod_i B_i) \prod_i A_i .$$

e) Nous pourrions aussi utiliser ceci pour démontrer que le résultant de la somme de deux relations $R_1(A,B)$ et $R_2(A,B)$ est égal à la somme des résultants, en vérifiant que :

$$\exists_A(R_1(A,B) + R_2(A,B)) \longleftrightarrow (\exists_A R_1(A,B) + \exists_A R_2(A,B))$$

Les quantificateurs portent tous sur la même variable, nous pouvons considérer que nous avons un problème de prédicats à une variable et si pour simplifier nous n'écrivons plus les variables nous avons à montrer que :

$\sum_i(R_{1i} + R_{2i}) \longleftrightarrow (\sum_i R_{1i} + \sum_i R_{2i})$ est une tautologie.

De même nous pouvons démontrer que le produit des résultants de deux relations est supérieur au résultant du produit de ces relations en montrant que :

$$\exists_A(R_1(A,B)_\wedge R_2(A,B)) \rightarrow (\exists_A R_1(A,B) \wedge \exists_A R_2(A,B))$$

c'est-à-dire :

$$\sum_i(R_{1i}\ R_{2i}) \rightarrow \sum_i R_{1i} \sum_i R_{2i}$$

est une tautologie.

En opérant de même que pour les exemples précédents nous obtenons facilement les résultats cherchés.

III - Déduction :

Nous voudrions aussi, avec la même méthode vérifier les règles de déduction faisant intervenir des prédicats. Considérons une déduction $\Gamma(x) \vdash B$, elle signifie que le membre de droite B peut être construit à partir des expressions situées à gauche de Γ et ceci quelque soient les variables dont ces dernières expressions dépendent. C'est-à-dire que si pour x donné $\Gamma(x) \vdash B$ nous devrons avoir $\forall x\ \Gamma(x) \vdash B$ et par conséquent, de même que pour le calcul des propositions, nous pouvons remplacer $\Gamma(x) \vdash B$ par $\forall x\ \Gamma(x) \Longrightarrow B$.

Vérifions par exemple une règle de déduction du système de Kleene :

$$C \rightarrow A(x) \vdash C \rightarrow \forall x\, A(x)$$

Ce qui se ramène à la déduction : $\forall x\ (C \rightarrow A(x)) \vdash C \rightarrow \forall x\, A(x)$.

Nous aurions donc à démontrer l'identité à 1 de :

$\prod_i (C' + A_i) \rightarrow (C \rightarrow \prod_i A_i)$ or C étant indépendant de x

$\prod_i (C' + A_i) = \prod_i C' + \prod_i A_i = C \rightarrow \prod_i A_i$ et $(C \rightarrow \prod_i A_i) \rightarrow (C \rightarrow \prod_i A_i)$

peut être considéré comme une formule du calcul des propositions, évidemment identique à 1.

Démontrons maintenant que si deux relations sont telles que $R_1(A,B,C) \leq R_2(A,B,C)$ alors les résultants par rapport à C sont tels que :

$S_1(A,B) \leq S_2(A,B)$. Ce qui revient à prouver que :

$R_1(A,B,C) \rightarrow R_2(A,B,C) \vdash \exists_C R_1(A,B,C) \rightarrow \exists_C R_2(A,B,C)$

Si nous n'écrivons plus les variables nous avons à montrer que :

$\rightarrow \prod_i (R'_{1i} + R_{2i}) \rightarrow \sum_i R_{1i} \sum_i R_{2i} = \sum_i R_{1i}\, R'_{2i} + \prod_i R'_{1i} + \sum_i R_{2i}$ est une identité à 1.

Considérons cette formule par rapport à $\sum_i R_{1i}$ et $\prod_i R_{1i}$:

$\sum_i R_{1i}\, R'_{2i} = 0 \prod_i R'_{1i} + \sum_k R'_{2k} \sum_i R_{1i} \sum_i R'_{1i} + \sum_k R'_{2k} \prod_i R_{1i}$.

$\sum_i R_{1i}\, R'_{2i} + \prod_i R'_{1i} + \sum_i R_{2i} =$

$$= 1 \prod_i R'_{1i} + (\sum_k R'_{2k} + \sum_i R_{2i}) \sum_i R_{1i} \sum_i R'_{1i} + (\sum_k R'_{2k} + \sum_i R_{2i}) \prod_i R_{1i}.$$

$$= 1 \prod_i R'_{1i} + (\sum_k R'_{2k} + \sum_k R_{2k} + \sum_j R_{2j}) \sum_i R_{1i} \sum_i R'_{1i} + (\sum_i R'_{2i} + \sum_i R_{2i}) \prod_i R_{1i}.$$

Etudions $F_{01} = \sum_k R'_{2k} + \sum_k R_{2k} + \sum_j R_{2j}$ par rapport à $\prod_k R_{2k}$ et $\sum_k R_{2k}$.

$$\sum_k R'_{2k} + \sum_k R_{2k} + \sum_j R_{2j} = (1+0+\sum_j R_{2j}) \prod_k R'_{2k} + (1+1+\sum_j R_{2j}) \sum_k R_{2k} \sum_k R'_{2k} + (0+1+\sum_j R_{2j}) \prod_k R_{2k}.$$

$$= 1 \prod_k R'_{2k} + 1 \sum_k R_{2k} \sum_k R'_{2k} + 1 \prod_k R_{2k} \equiv 1.$$

De même :

$$\sum_i R'_{2i} + \sum_i R_{2i} = 1 \sum_i R'_{2i} + 1 \sum_i R_{2i} \sum_i R'_{2i} + 1 \prod_i R_{2i} \equiv 1.$$

Donc

$$\sum_i R_{1i} R'_{2i} + \prod_i R'_{1i} + \sum_i R_{2i} = 1 \prod_i R'_{1i} + 1 \sum_i R_{1i} \sum_i R'_{1i} + 1 \prod_i R_{1i} \equiv 1.$$

IV - Programme pour les formules de prédicats à une variable :

De même que pour le calcul des propositions nous pouvons écrire une procédure ALGOL, utilisant la méthode décrite dans ce chapitre, qui nous indiquera si une formule F de prédicats à une variable est vraie, fausse ou indécidable. La formule F de prédicats sera auparavant écrite sous forme d'implications de variables généralisées de la forme $\sum_i E_i$ et $\prod_i E_i$, E_i étant sous forme de somme de produits de variables indicées. Nous utilisons ici l'opérateur implication à la place des deux opérateurs somme et produit car ceci nous permet de réduire le volume du programme.

Par rapport au calcul des propositions la procédure considérée ici sera plus longue, au point de vue écriture, car elle devra rechercher les valeurs de 3 coefficients au lieu de deux, à chaque pas, et surtout modifier la formule initiale en dédoublant les variables quand elle devra étudier les coefficients de la forme F_{01}. Nous devrons aussi écrire une procédure supplémentaire nous permettant d'obtenir les coefficients G_{00}, G_{01}, G_{11} de chaque variable généralisée G. L'ensemble de ces nouvelles procédures comprend 4542 unités syntaxiques.

Ce programme dont le listage suit, a été utilisé sur IBM 7044 :

```
Début
 procédure X(M,K,P,F00,F01,F11) ; valeur K ; entier M,K,P,F00,F01,F11 ;
 début si K=M alors début si P=1 alors début F00 := 0 ; F01 := -1 ; F11 := 1 fin
                               sinon début F00 := 1 ; F01 := -2 ; F11 := 0 fin
                          fin sinon F00 := F01 := F11 := K
 fin ;
 entier procédure PROD(A,B,N) ; valeur A,B ; entier A,B,N ;
 début si A=0 ∨ B=0 ∨ (A = -1 ∧ B = -2) ∨ (A = -2 ∧ B = -1) alors PROD := 0
      sinon si A=1 ∨ B=1 alors début si B≠1 alors PROD := B sinon PROD := A
                                 fin
      sinon si A < 0 ∨ B < 0 alors début si B > 0 alors PROD := A sinon PROD := B
                                    fin
      sinon si A=B alors PROD := A sinon PROD := N+2
 fin ;
 entier procédure SOM(A,B,N) ; valeur A,B ; entier A,B,N ;
 début si A=1 ∨ B=1 ∨ (A= -1 ∧ B= -2) ∨ (A= -2 ∧ B= -1) alors SOM := 1
      sinon si A=0 ∨ B=0 alors début si B≠0 alors SOM := B sinon SOM := A
                                 fin
      sinon si A < 0 ∨ B < 0 alors début si B > 0 alors SOM := A sinon SOM := B
                                    fin
      sinon si A=B alors SOM := A sinon SOM := N+2
 fin ;
 procédure PS(X,PROD,SOM,M,N,NE,B,C,P,K,H00,H01,H11,R) ; valeur B,C ;
          entier M,N,NE,B,C,H00,H01,H11,R ; entier tableau P,K ;
          procédure X ; entier procédure PROD, SOM ;
 début entier S ; entier tableau F00,F01,F11[1:N] ;
      X(M,K[R,1], P[R,1], H00,H01,H11) ;
      si N > 1 alors début S := 2 ;
                     RETOUR : X(M,K[R,S], P[R,S], F00[S], F01[S], F11[S]) ;
                              si C=1 alors début H00 := PROD(H00,F00[S], NE) ;
                                                 H01 := PROD(H01,F01[S], NE) ;
                                                 H11 := PROD(H11,F11[S], NE)
                                           fin
```

```
                              sinon début H00 := SOM(H00,F00[S], NE) ;
                                          H01 := SOM(H01,F00[S], NE) ;
                                          H11 := SOM(H11,F11[S], NE)
                                    fin
                              si S<N alors début S := S+1 ; allera RETOUR fin
                         fin ;
     si (B=C ∨ M=NE)∧H01<0 alors début si B=1 alors H01 := 0 sinon H01 := 1 fin ;
     si B=2 alors début si H00 = NE+1 alors H00 := 1 ;
                        si H01 = NE+1 alors H01 := 1 ;si H11=NE+1 alors H11:=1
                  fin
           sinon début si H00 = NE+1 alors H00 := 0 ;
                        si H01 = NE+1 alors H01 := 0 ; si H11=NE+1 alors H11:=0
                  fin
fin ;
procédure IP(PS,X,PROD,SOM,R00,R01,R11,M,N1,N2,B1,B2,C1,C2,P1,P2,K1,K2,T,Q,C,NE,S1,S2);
          entier M,N1,N2,B1,B2,C1,C2,Q,NE,S1,S2; booleen C ; procédure PS,X ;
          entier procédure PROD,SOM ; entier tableau R00,R01,R11,P1,P2,K1,K2,T ;
début entier procédure F(A,B,N) ; valeur A,B ; entier A,B,N ;
     début si A=0 ∨ B=1 alors F := 1 sinon si A=1 alors F := B sinon F := N+2
     fin ;
     aiguillage AIG := E1,E2,E3,E4 ; entier J ; booleen tableau D[1:4] ;
     D[1] := B1 > 0 ∧ B2 > 0  ;  D[2] := B1 < 0 ∧ B2 > 0 ;
     D[3] := B1 > 0 ∧ B2 < 0  ;  D[4] := vrai ;
     pour J := 1,2,3,4 faire si D[J] alors allera AIG[J]
  E1 : PS(X,PROD,SOM,M,N1,NE,B1,C1,P1,K1,R00[1], R01[1], R11[1], S1) ;
       PS(X,PROD,SOM,M,N2,NE,B2,C2,P2,K2,R00[2], R01[2], R11[2], S2) ;
     allera TERM ;
  E2 : R00[1] := T[0,1] ; R01[1] := T[0,2] ; R11[1] := T[0,3] ;
       PS(X,PROD,SOM,M,N2,NE,B2,C2,P2,K2,R00[2], R01[2], R11[2], S2) ;
     allera TERM ;
  E3 : R00[2] := T[0,1] ; R01[2] := T[0,2] ; R11[2] := T[0,3] ;
       PS(X,PROD,SOM,M,N1,NE,B1,C1,P1,K1,R00[1], R01[1], R11[1], S1) ;
     allera TERM ;
```

```
    E4 : ROO[1] := T[0,1] ; RO1[1] := T[0,2] ; R11[1] := T[0,3] ;
         ROO[2] := T[Q,1] ; RO1[2] := T[Q,2] ; R11[2] := T[Q,3] ;
         Q := Q-1 ;
  TERM : T[0,1] := F(ROO[1], ROO[2], NE) ;
         T[0,2] := F(RO1[1], RO1[2], NE) ;
         T[0,3] := F(R11[1], R11[2], NE) ;
       si C alors début T[Q+1,1]:= T[0,1];T[Q+1,2]:=T[0,2];T[Q+1,3]:=T[0,3] fin
fin ;
procédure IMPLICATION(T,NE,PS,IP,X,PROD,SOM,N,B,P,K,C,CB) ; valeur T,NE ;
         entier T,NE ; procédure PS,IP,X ; entier procédure PROD,SOM ;
         entier tableau N,B,P,K,C ; booleen tableau CB ;
début entier A,M,L,R,S,Q,D1,D2,AA,LL ; booleen V ;
      entier tableau G11,GOO,GO1[1:2×NE×(NE-1)], H,J[0:2×NE×(NE-1)],
                     ROO,RO1,R11[1:2],TE[0:2×T×NE↑2], T1[1:2×T×NE↑2],
                     PP,KK [1:2×T×NE↑2, 1:2×NE×(NE-1), 1:2×NE×(NE-1)],
                     BB,CC,NN[1:2×T×NE↑2, 1:2×NE×(NE-1)],
                     PG,KG [1:2×T×NE↑2 , 1:2×NE×(NE-1), 1:2],
                     BG,CG,NG[1:2×T×NE↑2, 1:2] ;
      booleen tableau CBG[1:2×T×NE↑2, 1:2] ;
      pour L := 1 pas 1 jusqua 2×T faire début
      N[L] := EDONNEE ; B[L] := EDONNEE ; C[L] := EDONNEE fin
      M := 1 ; D1 := 0 ; D2 := 0 ;
RETOUR2 : si M > NE alors allera TERM ;
      R := 1 ; S := 2 ; Q := 0 ;
RETOUR1 : IP(PS,X,PROD,SOM,ROO,RO1,R11,M,N[R], N[S], B[R], B[S], C[R], C[S],
          P,P,K,K,TE,Q,CB[R], NE,R,S) ;
      si CB[R] alors Q := Q+1 ;
      si S < 2×T alors début R := R+2 ; S := S+2 ; allera RETOUR1 fin ;
      G11[M] := TE[0,3] ; GO1[M] := TE[0,2] ; GOO[M] := TE[0,1] ;
      ECRIRE (GOO[M], GO1[M], G11[M]) ;
      si (M+1 < GOO[M] ∧ GOO[M] < NE) ∨ (M+1 < GO1[M] ∧ GO1[M] < NE) ∨
         (M+1 < G11[M] ∧ G11[M] < NE) alors allera TERM ;
      si GOO[M] = 0 ∨ GOO[M] = 1 alors allera SUITE1 ;
```

```
      pour R := 1 pas 1 jusqua 2xT faire
      début BB[R,M] := B[R] ; CC[R,M] := C[R] ; NN[R,M] := N[R] ;
            pour L := 1 pas 1 jusqua NE faire
            début PP[R,L,M] := P[R,L] ; KK[R,L,M] := K[R,L] ;
                  si K[R,L] = M alors
                  début si P[R,L] = 1 alors K[R,L] := 0 sinon K[R,L] := 1 fin
            fin
      fin ;
      M := M+1 ; allera RETOUR2 ;
SUITE1 : si G11[M] = 0 ∨ G11[M] = 1 alors allera SUITE2 ;
      D1 := D1+1 ; H[D1] := M+1 ;
      pour R := 1 pas 1 jusqua 2xT faire
      début BB[R,M] := B[R] ; CC[R,M] := C[R] ; NN[R,M] := N[R] ;
            pour L := 1 pas 1 jusqua NE faire
            début PP[R,L,M] := P[R,L] ; KK[R,L,M] := K[R,L] ;
            si K[R,L] = M alors
            début si P[R,L] = 1 alors K[R,L] := 1 sinon K[R,L] := 0 fin
            fin
      fin ;
      M := M+1 ; allera RETOUR2 ;
SUITE2 : si G00[M] ≠ G11[M] alors allera TERM ;
        si G01[M] = 0 ∨ G01[M] = 1 alors allera SUITE3 ;
        D2 := D2+1 ; J[D2] := M+1 ; T1[M] := T ;
      pour R := 1 pas 1 jusqua 2xT faire
      début BB[R,M] := B[R] ; CC[R,M] := C[R] ; NN[R,M] := N[R] ;
            pour L := 1 pas 1 jusqua NE faire début
            KK[R,L,M] := K[R,L] ; PP[R,L,M] := P[R,L] fin
      fin ;
      pour R := 1 pas 1 jusqua 2xT faire si B[R] ≠ C[R] alors
      début pour L := 1 pas 1 jusqua N[R] faire
            si (K[R,L] = M) ∧ (¬ V) alors allera CHGT
      fin ;
```

```
        pour R := 1 pas 1 jusqua 2×T faire pour L := 1 pas 1 jusqua NE faire
        si K[R,L] = M alors début si B[R]=1 alors K[R,L]:=0 sinon K[R,L]:=1 fin ;
        Si V alors V := faux ; M := M+1 ; allera RETOUR2 ;
CHGT :  pour R := 1 pas 2 jusqua 2×T faire
        début pour L := 1, 2 faire
              début BG[R,L] := B[R+L-1] ; CG[R,L] := C[R+L-1] ;
                    NG[R,L] := N[R+L-1] ; CBG[R,L] := CB[R+L-1]
              fin ;
              pour L := 1 pas 1 jusqua NE faire     début
              KG[R,L,1] := K[R,L] ; KG[R,L,2] := K[R+1,L] ;
              PG[R,L,1] := P[R,L] ; PG[R,L,2] := P[R+1,L]      fin
        fin ;
        S := 1
        pour R := 1 pas 2 jusqua 2×T1[M] faire
        début AA := 1 ; si BG[R,2] ≠ -1 alors
              début pour L := 1 pas 1 jusqua NG[R,2] faire
                    si KG[R,L,2] = M alors début LL := 1, allera RET fin
                    sinon si L = NG[R,2] alors LL := 2
              fin   sinon LL := 1 ;
RET :         pour L := 1 pas 1 jusqua NG[R,LL] faire
              si KG[R,L,LL] = M alors allera ETI ;
              si BG[R,LL] = 1 alors
              début T := T+2 ; B[S] := BG[R,LL] ; C[S] := CG[R,LL] ;
                    N[S] := N[S+1] := NG[R,LL] ; B[S+1] := 2 ; B[S+2] := -1 ;
                    B[S+3] := C[S+3] := N[S+3] := 1 ;
                    si CG[R,LL] = 1 alors C[S+1] := 2 sinon C[S+1] := 1 ;
                    pour L := 1 pas 1 jusqua NG[R,LL] faire
                    début si KG[R,L,LL] = NE+1 alors K[S,L] := K[S+1,L]:=(2×NE-M)+1
                          sinon K[S,L] := KG[R,L,LL] ; P[S,L] := PG[R,L,LL] ;
                          A := KG[R,L,LL] -M ;
                          si KG[R,L,LL]≠0∧KG[R,L,LL]≠1∧KG[R,L,LL] ≠ NE+1
                          alors K[S+1,L] := NE+A sinon si KG[R,L,LL] ≠ NE+1
                          alors début si KG[R,L,LL]= 1 alors K[S+1,L]:=0 sinon
                                K[S+1,L] := 1 fin ;
```

```
                si PG[R,L,LL]=0 alors P[S+1,L]:=1 sinon P[S+1,L] := 0
            fin ; K[S+3,1] := 0 ;
            si LL=1 alors début B[S+4] := -1, si AA=2 alors
            début B[S+5] := -1 ; CB[S-2] := vrai ; S := S+6 fin
            sinon S := S+5 ; CB[S+4] := CBG[R,LL] fin sinon S := S+4
        fin sinon si BG[R,LL] = 2 alors
        début T := T+1 ; B[S] := 1 ; B[S+1] := BG[R,LL] ; C[S+1] := CG[R,LL];
            N[S] := N[S+1] := NG[R,LL] ;
            si CG[R,LL] = 1 alors C[S] := 2 sinon C[S] := 1 ;
            pour L := 1 pas 1 jusqua NG[R,LL] faire
            début si KG[R,L,LL]=NE+1 alors K[S,L] := K[S+1,L]:=(2xNE-M)+1
                sinon si KG[R,L,LL] = 1 alors K[S,L] := 0 sinon
                si KG[R,L,LL]=0 alors K[S,L]:=1 sinon K[S,L]:=KG[R,L,LL];
                P[S+1,L] := PG[R,L,LL] ; A := KG[R,L,LL] - M ;
                si KG[R,L,LL]≠0 ∧ KG[R,L,LL]≠1 ∧ KG[R,L,LL]≠NE+1
                alors K[S+1,L] := KG[R,L,LL] ;
                si PG[R,L,LL] = 0 alors P[S,L] := 1 sinon P[S,L] := 0
            fin
            si LL=1 alors début B[S+2] := -1 ; si AA=2 alors
            début B[S+3] := -1 ; CB[S-2] := vrai ; S := S+4 fin
            sinon S := S+3 ; CB[S+2] := CBG[R,LL] fin sinon S := S+2
        fin sinon début B[S] := -1 ; si LL=1 alors
                début si AA=2 alors début
                    B[S+1] := -1 ; S := S+2 fin
                    sinon S := S+1 ; CB[S] := CBG[R,LL]
                fin sinon S := S+1
                fin ;
        allera TERMG ;
    ETI : pour L := 1 pas 1 jusqua NG[R,LL] faire début
        si KG[R,LL]=M alors début si BG[R,LL] = 1 alors KG[R,L,LL] := 0
                                  sinon KG[R,L,LL] := 1
                              fin ; P[S,L] := PG[R,L,LL] fin ;
```

```
        B[S] := BG[R,LL] ; C[S] := CG[R,LL] ; N[S] := NG[R,LL] ;
        pour L := 1 pas 1 jusqua NG[R,LL] faire
        si (CG[R,LL]=2 ∧ PG[R,LL]=1) ∨ (CG[R,LL]=1 ∧ PG[R,L,LL] =0)
        alors K[S,L] := KG[R,L,LL] sinon début A := KG[R,L,LL] - M ;
        si KG[R,L,LL]≠0 ∧ KG[R,L,LL]≠ 1 alors K[S,L] := NE+A
        sinon K[S,L] := KG[R,L,LL] fin ;
        si LL=1 alors début si AA=2 alors début B[S+1] := -1 ; S := S+2
                                                fin
                            sinon S := S+1 ; CB[S] := CBG[R,LL]
                      fin   sinon S := S+1 ;
  TERMG : AA := AA+1 ; si AA=2 alors début si LL=1 alors LL := 2
                                           sinon LL := 1 ; allera RET
                                      fin
        fin ; V := vrai ; NE := 2XNE-M ; M := M+1 ; allera RETOUR2 ;
 SUITE3 : si G01[M] ≠ G11[M] alors allera TERM ;
        si M=1 alors début si G11[M] = 1 alors ECRIRE (vrai)
        sinon ECRIRE (faux) ; allera TERM1 fin ;
        si M=J[D2] alors début G01[M-1] := G01[M] ; M := M-1 ;
        si V alors V:=faux ; D2:=D2-1 ; NE:=(NE+M)/2;T:=T1[M];allera SUITE3 fin ;
        si M=H[D1] alors début G11[M-1] := G11[M] ; D1 := D1-1 ; M := M-1 ;
        pour R := 1 pas 1 jusqua 2XT faire début
        pour L := 1 pas 1 jusqua NE faire début
        K[R,L] := KK[R L,M] ; P[R L] := PP[R,L,M] fin ;
        B[R] := BB[R,M] ; C[R] := CC[R,M] ; N[R] := NN[R,M] fin ;
        allera SUITE2 fin ;
        G00[M-1] := G00[M] ; M := M-1 ;
        pour R := 1 pas 1 jusqua 2XT faire début
        pour L := 1 pas 1 jusqua NE faire début
        K[R,L] := KK[R,L,M] ; P[R L] := PP[R,L,M] fin ;
        B[R] := BB[R,M] ; C[R] := CC[R,M] ; N[R] := NN[R,M] fin ;
        allera SUITE1 ;
   TERM : ECRIRE ('('INDECIDABLE')') ;
  TERM1 :
  fin ;
```

```
entier tableau P,K[1:32, 1:7], N,B,C[1:32] ; booleen tableau CB[1:32] ;
K[1,1] := 1 ; K[2,1] := 2 ; K[3,1] := 1 ; K[3,2] := 2 ;
P[1,1] := P[2,1] := P[3,2] := 1 ;
IMPLICATION (2,2,PS,IP,X,PROD,SOM,N,B,P,K,C,CB)
fin ;
```

Ce programme nous a permis d'étudier, en particulier, les formules suivantes :

- $\exists x (A(x) \to B(x)) \to (\forall x\ A(x) \to \exists x\ B(x))$ (formule considérée pour activer ci-dessus les procédures)

c'est-à-dire : $\sum_i (A'_i + B_i) \to (\prod_i A_i \to \sum_i B_i)$

Nous obtenons comme résultat :

F00	F01	F11	
1	1	4	(coefficients par rapport à $\prod_i A_i$ et $\sum_i A_i$)
1	1	1	(coefficients par rapport à $\prod_i B_i$ et $\sum_i B_i$)

VRAI

- $\sum A_i B_i \to \sum A_i B_i$

résultat :	1	4	4	(coefficients par rapport à $\prod_i A_i$, $\sum_i A_i$)
	1	1	1	(coefficients par rapport à $\prod_i B_i$, $\sum_i B_i$)
	5	5	5	(coefficients par rapport à $\prod_j B_j$, $\sum_j B_j$)
	1	1	1	(coefficients par rapport à $\prod_k B_k$, $\sum_k B_k$)
	1	1	1	(coefficients par rapport à $\prod_k B_k$, $\sum_k B_k$)
	1	1	1	(coefficients par rapport à $\prod_k B_k$, $\sum_k B_k$)

VRAI

- $A(x) \rightarrow \exists x\ A(x)$ ou $\sum_i A_i \delta_i \rightarrow \sum_i A_i$

résultat : 1 1 1 (coefficients par rapport à $\prod_i A_i$, $\sum_i A_i$)

VRAI

- $\sum_i B_i \rightarrow \sum_i A_i B_i$

résultat : 4 4 4 (coefficients par rapport à $\prod_i A_i$, $\sum_i A_i$)

1 0 0 (coefficients par rapport à $\prod_i B_i$, $\sum_i B_i$)

INDECIDABLE

Pour l'étude d'une formule, le travail total a duré 2 minutes 12 secondes, la recherche de la validité étant effectuée en 1 seconde. Ce programme a été aussi utilisé pour étudier six formules simultanément, le travail total a duré 2 minutes 20 secondes, et la recherche de la validité a été effectuée en deux secondes.

V - Syllogisme :

1) Définition :

Considérons un certain ensemble E et des propriétés P(x), R(x), S(x) des éléments x de cet ensemble. Le syllogisme consiste, une relation étant donnée d'une part entre P(x) et R(x), d'autre part entre R(x) et S(x), à déterminer la relation existant entre P(x) et S(x). Le syllogisme est basé sur les quatre relations, de départ, suivantes :

A : tous P sont R, ce qui peut s'exprimer sous forme de formules de prédicats : $\forall x\ (P(x) \rightarrow R(x))$.

O : quelques P ne sont pas R, c'està-dire : $\exists x\ (P(x) \wedge \neg R(x))$.

I : quelques P sont R ou : $\exists x\ (P(x) \wedge R(x))$.

E : aucun P n'est R, c'est-à-dire : $\forall x\ (P(x) \rightarrow \neg R(x))$

Si P_i, R_i, S_i repèrent la vérité ou la fausseté de P, R, S pour chaque valeur de x, les relations de départ peuvent s'écrire, en algèbre de Boole :

$$A = \prod_j (P'_j + R_j) \ ; \ O = \sum_i P_i R'_i \ ; \ I = \sum_i P_i R_i \ ; \ E = \prod_i (P'_i + R'_i).$$

Nous considérons dans la suite le syllogisme renforcé qui consiste à assurer l'existence d'une des propriétés dans le cas A qui devient : il existe P et tous P sont R ce qui peut s'écrire :

$$A = \sum_i P_i \prod_j (P'_j + R_j).$$

Nous utiliserons aussi les relations déduites des précédentes en échangeant P et R, ce qui nous donne :

$$\tilde{A} = \sum_i R_i \prod_j (R'_j + P_j) \quad ; \quad \tilde{O} = \sum_i R_i P'_i.$$

2) <u>Produits d'indicateurs</u> :

A partir de relations de la forme précédente, appelées indicateurs, σ_{PR} d'une part et ρ_{RS} d'autre part, nous allons chercher la relation existant entre P et S en considérant le résultant par rapport à R de ces deux indicateurs.

Nous pouvons établir la table suivante :

	A	$\tilde{A}$	0	$\tilde{0}$	I	E
A	A	$\sum_i P_i \sum_j S_j$	$\sum_i P_i \sum_j S'_j$	$\sum_i P_i \sum_j P'_j S_j$	$\sum_i P_i \sum_j P_j$	$\sum_i P_i \pi_j (P'_j + S'_j)$
$\tilde{A}$	I	$\tilde{A}$	0	$\sum_i P_i \sum_{j \neq i} S_j$	I	0
0	$\sum_i P_i \sum_{j \neq i} S_j$	$\sum_i P_i \sum_j P_j S'_j$	$\sum_i P_i \sum_{j \neq i} S'_j$	$\sum_i P_i \sum_j S_j$	$\sum_i P_i \sum_{j \neq i} S_j$	$\sum_i P_i$
$\tilde{0}$	$\tilde{0}$	$\sum_i P'_i \sum_j S_j$	$\sum_i P_i \sum_{j \neq i} S'_j$	$\sum_i P'_i \sum_{j \neq i} S_j$	$\sum_i P'_i \sum_j S_j$	$\sum_i P'_i S'_i$
I	I	$\sum_i P_i \sum_j S_j$	$\sum_i P_i \sum_j S'_j$	$\sum_i P_i \sum_{j \neq i} S_j$	$\sum_i P_i \sum_j S_j$	0
E	$\tilde{0}$	$\sum_i R_i \pi_j (P'_j + S'_j)$	$\sum_i P'_i S'_i$	$\sum_i S_i$	$\tilde{0}$	1

Certains résultants obtenus sont égaux à un indicateur multiplié soit par $\sum_i P_i$ soit par $\sum_i S_i$. Nous retrouvons donc cet indicateur à condition que P et S ne soient pas des propriétés identiquement fausses. Nous avons ainsi :

$A.\tilde{0} = (\sum_i P_i)\,\tilde{0}$; $A.E = (\sum_i P_i)\,E$; $0.\tilde{A} = (\sum_i R_i)\,0$; $E.\tilde{A} = (\sum_i R_i)\,E$;

De plus le but du syllogisme étant tel qu'une relation valable entre P et R et une relation valable entre R et S nous permettent de déduire immédiatement une relation valable entre P et S, si le résultant, obtenu à partir de deux indicateurs, est inférieur à une autre relation intéressante entre P et S, nous pourrons aussi considérer cette autre relation comme résultant. Car si $\sigma_{PR} = 1$, $\rho_{RS} = 1$ entraînent que $\tau_{PS} = 1$ et si $\tau_{PS} \leq F(P,S)$ nous avons bien $F(P,S) = 1$ si $\sigma_{PR} = \rho_{RS} = 1$.

Ainsi nous pouvons considérer que :

$A. \widetilde{O} \rightarrow \widetilde{O}$; $A. E \rightarrow E$; $O. \widetilde{A} \rightarrow O$; $E. \widetilde{A} \rightarrow E.$

De même A et $\widetilde{A}$ étant inférieurs ou égaux à I, chaque fois que nous rencontrons A et $\widetilde{A}$ dans le tableau nous pourrions les remplacer par I.

D'où $A. A \rightarrow I$; $\widetilde{A}. \widetilde{A} \rightarrow I$. Et comme $(\sum_i P_i)\ E \leq O$, $(\sum_i R_i)\ E \leq \widetilde{O}$ nous avons encore :

$A. E \rightarrow O$; $E. A \rightarrow \widetilde{O}.$

Les autres produits étant soit supérieurs soit non reliés par une relation aux indicateurs, ils ne donnent pas de propriétés intéressantes entre P et S sauf $\widetilde{O}. E$ et $E. O$ qui indiquent qu'il existe au moins une valeur de x telle que les propriétés P(x) et S(x) sont fausses ensemble.

Mais il peut être aussi intéressant, ayant une relation entre P et S, de connaître les relations entre P et R et entre R et S à partir desquelles elle peut être déduite. Remarquons que nous saurons seulement qu'il existe au moins une propriété R telle que ces relations soient valables.

Or le résultant $\tau_{PS} = 1$ de deux indicateurs σ_{PR} et ρ_{RS} est, par définition, une condition nécessaire et suffisante pour qu'il existe au moins un R tel que $\sigma_{PR} = 1$ et $\rho_{RS} = 1$. Donc si nous avons une relation $F(P,S) \leq \tau_{PS}$ et si $F(P,S) = 1$ nous pourrons en déduire que $\tau_{PS} = 1$ donc qu'il existe au moins un S tel que σ_{PR} et $\rho_{RS} = 1$. Par exemple $\sum_i P_i \sum_j S_j \geq I$ donc $\sum_i P_i \sum_j S_j \geq A$ et $\sum_i P_i \sum_j S_j \geq \widetilde{A}$, ce qui entraîne qu'il existe R tel que si :

$I_{PS} = 1$ alors $I_{PR} = 1$ et $I_{RS} = 1.$

$A_{PS} = 1$ alors $I_{PR} = 1$ et $I_{RS} = 1.$

$\widetilde{A}_{PS} = 1$ alors $I_{PR} = 1$ et $I_{RS} = 1.$

De même $\sum_i P_i \sum_j R'_j \geq O$; $\sum_i P_i \sum_j R'_j \geq (\sum_i P_i)\ E.$

Et nous pouvons faire ceci pour la plupart des résultants sauf pour ceux de la forme $\sum_i P_i \, (\sum_{j \neq i} R_j)$ non comparables avec les indicateurs. Remarquons que $E. E = 1$ entraîne qu'il est toujours possible pour P et S données de trouver R telle que $E_{PR} = E_{RS} = 1$.

3) Syllogisme et prédicats unaires :

Supposons que nous ayons une formule de prédicats notée k.

a) si $k \geq \tau_{PS}$ et s'il existe R telle que $\rho_{PR} = 1$ et $\sigma_{RS} = 1$ nous savons que $\tau_{PS} = 1$ donc $k = 1$. Pour connaître la validité de k il suffirait donc de montrer la validité de ρ et de σ.

b) si $K \leq \tau_{PS}$ et si k est une formule vraie donc si k=1 alors $\tau_{PS} = 1$ donc il doit exister R telle que $\rho_{PR} = 1$ et $\sigma_{RS} = 1$. D'où si nous possédons deux propriétés P et S telles qu'il soit impossible de trouver R telle que ρ_{PR} et σ_{RS} soient égaux à 1 simultanément, nous pourrons en déduire que k n'est pas vraie.

En quelque sorte si une formule de prédicats est comparable à un produit d'indicateurs, il sera possible d'établir sa validité ou sa non-validité par syllogisme. Mais ceci ne sera pas généralement, simple, en particulier pour établir la non-validité, et ne semble pas présenter d'intérêt par rapport à la méthode, décrite au début de ce chapitre, beaucoup plus systématique.

- CHAPITRE III -

VALUATION DE LA LOGIQUE DES PROPOSITIONS.

I - Valuation de la logique classique des propositions par l'algèbre de Boole :

Rappelons qu'il est possible de démontrer que tout axiome ou tout théorème, de la logique classique, prend la valeur 1 en algèbre de Boole et que, réciproquement, une expression booléenne quelconque identique à 1 correspond bien à un axiome ou à un théorème en logique classique.

En effet nous vérifions facilement que les axiomes du système de Kleene, par exemple, en écriture booléenne, deviennent égaux à 1. De même, tout théorème prendra la valeur 1 puisqu'il se déduira d'une suite de formules qui seront soit des axiomes soit obtenues à partir des formules précédentes et de la règle de déduction, or celle-ci est telle que si $A=1$ et $A'+B=1$ alors $B=1$.

D'autre part, toute expression booléenne f peut être considérée comme une fonction des lettres qu'elle contient et s'écrire par rapport à une de ces lettres, A par exemple, sous la forme :

$$f(A,X) = f_0(X)\ A' + f_1(X)\ A.$$

Supposons tout d'abord que f ne dépend que de A, elle sera identique à 1 si et seulement si $f_0 = f_1 = 1$ donc si $f = A' + A$ c'est-à-dire $\neg A \vee A$ en notation habituelle de la logique classique. Or il est possible de démontrer, à l'aide d'un système d'axiomes quelconque, de la logique classique, que $\neg A \vee A$ est un théorème. Ce qui entraîne que toute formule identique à 1 contenant une seule lettre est un théorème.

Supposons maintenant que f dépend de plusieurs variables et que $f_0(X)$ et $f_1(X)$ sont des théorèmes. En logique classique f s'écrira $(f_0(X) \wedge \neg A) \vee (f_1(X) \wedge A)$. Cette formule est un théorème, ce qui se montre très facilement en considérant le système de Wang. Le séquent de départ $\Rightarrow f_0(X) \wedge \neg A,\ f_1(X) \wedge A$ se décompose en quatre séquents :

$$\Rightarrow f_0(X),\ f_1(X).$$
$$\Rightarrow f_0(X),\ A.$$
$$\Rightarrow \neg A,\ f_1(X).$$
$$\Rightarrow \neg A,\ A \quad \text{qui devient } A \Rightarrow A.$$

De même que le quatrième séquent, les trois premiers se ramèneront à des expressions de la forme $\pi,\ P \Rightarrow \rho,\ P$ puisque $f_0(X)$ et $f_1(X)$ sont des théorèmes par hypothèse. Donc si $f_0(X)$ et $f_1(X)$ sont des théorèmes f(A,X) est aussi un théorème.

Or f(A,X) est identique à 1 si et seulement si $f_0(X)$ et $f_1(X)$ le sont. Si $f_0(X)$ et $f_1(X)$ ne dépendent que d'une variable et sont identiques à 1, elles sont des théorèmes et par conséquent f(A,X) aussi. Si $f_0(X)$ et $f_1(X)$ dépendent de plus d'une variable, nous pourrons les écrire sous forme de Lagrange par rapport à leurs variables jusqu'à ce que nous obtenions des expressions, ne dépendant plus que d'une variable, qui seront alors des théorèmes ; ce qui entraînera que $f_0(X)$ et $f_1(X)$ sont aussi des théorèmes et que, finalement, f(A,X) est un théorème.

Donc toute expression booléenne identique à 1 est un théorème en logique classique.

II - Semi-valuation de la logique intuitionniste :

1) Impossibilité d'une valuation finie :

Les systèmes d'axiomes utilisés en logique intuitionniste se déduisent de ceux de la logique classique en supprimant celui correspondant au principe du tiers exclus (principe selon lequel pour toute proposition A, on a soit A soit $\neg A$). Nous ne pouvons pas établir une correspondance entre l'algèbre de Boole et la logique intuitionniste car si nous considérons que chaque proposition peut prendre seulement deux valeurs, nous retombons évidemment sur la logique classique. Essayons donc d'associer à chaque proposition intuitionniste une des n valeurs (n fini) $0, \Phi_1, \ldots, \Phi_{n-2}, 1$, p valeurs choisies correspondant à 'vrai'. Il faut alors, que nous puissions écrire que tout axiome ou tout théorème est identiquement vrai et que toute règle de déduction est vérifiée. Le théorème $A \rightarrow A$, valable en logique intuitionniste, nous entraîne donc à écrire : $\underline{x \rightarrow x = \text{'vrai'}}$ pour $x = 0, 1$ ou $\Phi_i (1 \leq i \leq n-2)$. De plus la règle de déduction, $A, A \rightarrow B \vdash B$, nous donne $\underline{\text{'vrai'} \rightarrow y \neq \text{'vrai'} \text{ si } y \neq \text{'vrai'}}$. Ce qui nous permet de montrer que $\underline{\text{si } A = \text{'vrai' ou } B = \text{'vrai' alors } A \vee B = \text{'vrai'}}$. En effet les deux axiomes, du système de Kleene, $A \rightarrow A \vee B$ et $B \rightarrow A \vee B$ ne seront pas identiquement vrais si dans le cas $A =$ 'vrai' (ou $B =$ 'vrai') $A \vee B \neq$ 'vrai'.

Supposons, alors, que nous puissions trouver effectivement une valuation telle qu'à tout axiome et tout théorème de la logique intuitionniste nous puissions associer une valeur correspondant à 'vrai'. Réciproquement nous voudrions pouvoir dire que toute expression identique à une valeur 'vrai' est un axiome ou un théorème. Or un théorème de Gödel s'énonce :

Dans le calcul intuitionniste des propositions, on peut prouver $A \vee B$ si et seulement si on peut prouver soit A soit B, pour toutes formules A et B.

Considérons (n+1) propositions élémentaires différentes $A_1, A_2, \ldots, A_n, A_{n+1}$ et la formule logique : $\bigcup_{1 \leq \ell < k \leq n+1} (A_\ell \rightarrow A_k) = F_{n+1}$. Si nous associons toutes les combinaisons possibles des n valeurs 0, $\Phi_1, \ldots, \Phi_{n-2}$, 1 aux (n+1) propositions, nous avons pour chaque combinaison considérée deux propositions, au moins, correspondant à la même valeur. Donc d'après ce qui précède nous aurons, au moins, une implication $A_\ell \rightarrow A_k$ = 'vrai' et par conséquent F_{n+1} = 'vrai'. Or F_{n+1} ne peut être un théorème, de la logique intuitionniste, puisqu'aucune implication de la forme $A_\ell \rightarrow A_k$ n'en est un. Ce qui prouve qu'il est impossible d'avoir une valuation finie de la logique intuitionniste pour le calcul des propositions, même si plusieurs valeurs correspondent à 'vrai'. Aussi le fait de supposer, dans ce qui suit, que seule la valeur 1 correspond à 'vrai' semble n'enlever rien d'essentiel à la généralité du problème et nous permettra, en particulier, de simplifier l'écriture.

2) <u>Recherche d'une semi-valuation</u> :

Nous allons donc essayer de considérer une semi-valuation qui nous permettra de n'identifier à 1 qu'un minimum de formules qui ne sont pas des théorèmes en logique intuitionniste.

Remarquons tout d'abord que toute semi-valuation doit être telle que :

$\underline{x \rightarrow x = 1}$, $\underline{x \rightarrow 1 = 1}$ pour x = 1, 0 ou Φ_i $1 \leq i \leq n-2$.

Ceci provenant du théorème $A \rightarrow A$ et de $A \rightarrow (A \rightarrow A)$, cas particulier du premier axiome du système de Kleene.

Nous allons, maintenant, supposer que <u>$1 \rightarrow y = 1$ si et seulement si $y = 1$</u>, de manière, à pouvoir affirmer que tout théorème prendra la valeur 1 dès que nous aurons vérifié que tous les axiomes, d'un système quelconque, sont identiquement vrais. Sinon pour déterminer si la semi-valuation est valable il faudrait vérifier, effectivement, que chaque théorème correspond bien à la valeur 1 ce qui ne semble guère possible.

Avec l'hypothèse choisie nous avons vu que :
$x \vee y = 1$ si $x = 1$ ou $y = 1$. De même les axiomes $A \wedge B \rightarrow A$, $A \wedge B \rightarrow B$ et $A \rightarrow (B \rightarrow A \wedge B)$ entraînent :
$x \wedge y = 1$ si et seulement si $x = y = 1$.
Considérons, aussi, le théorème $(\neg A \vee B) \rightarrow (A \rightarrow B)$ et supposons que $\neg 1 = 1$; pour $A = 1$ et $B \neq 1$ nous obtiendrons $(1 \vee B) \rightarrow (1 \rightarrow B) = 1 \rightarrow (1 \rightarrow B)$, ce qui entraînerait $1 \rightarrow B = 1$, or ceci est impossible d'après ce qui précède. Donc $\neg 1 \neq 1$, et, nous poserons $\neg 1 = 0$. Comme le théorème $A \rightarrow \neg\neg A$ nous permet d'écrire que si $A = 1, \neg\neg A = 1$, nous en déduisons que $\neg 0 = 1$. Et pour $A = 0$, $B = \Phi_i$ $(1 \leq i \leq n-2)$, le théorème $(\neg A \vee B) \rightarrow (A \rightarrow B)$ nous donne, alors, $(1 \vee \Phi_i) \rightarrow (0 \rightarrow \Phi_i) = 1$, c'est-à-dire $1 \rightarrow (0 \rightarrow \Phi_i) = 1$ donc $0 \rightarrow \Phi_i = 1 (1 \leq i \leq n-2)$.

Nous savons de plus que si E est un théorème en logique classique alors $\neg\neg E$ est un théorème en logique intuitionniste. Soit donc E une formule, théorème en logique classique mais non en logique intuitionniste, si nous associons aux propositions élémentaires la composant tous les groupes de valeurs possibles, nous ne devrons jamais obtenir la valeur 0 $(\neg\neg 0 = 0)$, nous pourrons obtenir la valeur 1 $(\neg\neg 1 = 1)$ et nous voudrions, pour que la semi-valuation présente effectivement un intérêt, pouvoir obtenir des valeurs Φ_i afin que tous les théorèmes de la logique classique ne soient pas identiques à 1. De manière qu'un maximum d'expressions, de la forme E ne soient pas identifiées à 1 nous choisirons $\underline{\neg\neg\Phi_i = 1}$ pour $1 \leq i \leq n-2$, car nous aurons alors un maximum de valeurs possibles à associer à E. Si $\neg\neg \Phi_i = 1$ pour tout i, alors $\neg \Phi_i \neq 1$ et $\neg \Phi_i \neq \Phi_j$. En effet supposons que pour $i = i_0$ quelconque, $\neg \Phi_{i_0} = \Phi_j$ alors $\neg\neg\Phi_{i_0} = \neg \Phi_j = 1$, or ceci èst impossible car, alors, $\neg\neg\Phi_j = 0$. Donc si $\neg\neg\Phi_i = 1$, pour tout i, $\neg \Phi_i = 0$ $(1 \leq i \leq n-2)$. Si pour tout i $\neg \Phi_i = 0$ alors $\Phi_i \rightarrow 0 \neq 1$, en effet l'axiome $(A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A)$ nous donne pour $A = \Phi_i$, $B = 0$: $(\Phi_i \rightarrow 0) \rightarrow (1 \rightarrow 0) = 1$ et $\Phi_i \rightarrow 0 = 1$ entraînerait $1 \rightarrow 0 = 1$ ce qui est impossible d'après ce qui précède.

Pour A = 1, B = 1, ce même axiome entraîne $1 \to ((1 \to 0) \to 0) = 1$ donc $(1 \to 0) \to 0 = 1$, et, comme pour tout i $\Phi_i \to 0 \neq 1$, la seule valeur possible pour $1 \to 0$ est 0. Nous déduisons, alors, de $(\Phi_i \to 0) \to (1 \to 0) = 1$, que $\Phi_i \to 0 = 0$ $(1 \leq i \leq n-2)$. Soit maintenant l'axiome $A \to (B \to A)$ pour $A = \Phi_i$, $B = 1$, ce qui nous donne $\Phi_i \to (1 \to \Phi_i) = 1$ et ce qui entraîne $1 \to \Phi_i \neq 0$. De même les axiomes $A \wedge B \to A$ et $A \wedge B \to B$ entraînent qui si A = 0 ou B = 0 alors $A \wedge B = 0$ car $x \to 0 = 1$ si et seulement si $x = 0$. L'axiome $A \to (B \to A \wedge B)$ nous permet d'écrire que si $A \neq 0$ et $B \neq 0$ alors $A \wedge B \neq 0$; en effet si $B \neq 0$ $B \to 0 = 0$ et si $A \neq 0$ $A \to 0 = 0$. De même $(A \to C) \to ((B \to C) \to ((A \vee B) \to C))$ entraîne si $A = B = 0$ $A \vee B = 0$, et ce même axiome nous permet d'écrire que $0 \vee \Phi_i$, $\Phi_i \vee 0$, $\Phi_i \vee \Phi_i$ ont des valeurs différentes de 1 ; ces mêmes quantités ainsi que $\Phi_i \vee \Phi_j$ sont différentes de 0 car $B \to A \vee B$ et $A \to A \vee B$ ne pourraient pas alors être identiques à 1. De même $\Phi_i \to \Phi_j \neq 0$, sinon $\Phi_j \to (\Phi_i \to \Phi_j) \neq 1$ et l'axiome $A \to (B \to A)$ ne pourrait pas être vérifié comme étant identiquement vrai.

En résumé, <u>si nous choisissons une semi-valuation telle que $1 \to y \neq 1$ si $y \neq 1$ et $\neg\neg\Phi_i = 1$ pour tout i</u>, nous pouvons dresser le tableau suivant :

A	B	$A \to B$	$A \wedge B$	$A \vee B$	$\neg A$	$\neg B$
0	0	1	0	0	1	1
0	Φ_i	1	0	$z_2^i \neq 0$ et $\neq 1$	1	0
0	1	1	0	1	1	0
Φ_i	0	0	0	$z_4^i \neq 0$ et $\neq 1$	0	1
Φ_i	Φ_j	$x_5^{ij} \neq 0$	$y_5^{ij} \neq 0$ et $\neq 1$	$z_5^{ij} \neq 0$	0	0
Φ_i	Φ_i	1	$y_5^{ii} \neq 0$ et $\neq 1$	$z_5^{ii} \neq 0$ et $\neq 1$	0	0
Φ_j	Φ_i	$x_5^{ji} \neq 0$	$y_5^{ji} \neq 0$ et $\neq 1$	$z_5^{ji} \neq 0$	0	0
Φ_i	1	1	$y_6^i \neq 0$ et $\neq 1$	1	0	0
1	0	0	0	1	0	1
1	Φ_i	$x_8^i \neq 0$ et $x_8^i \neq 1$	$y_8^i \neq 0$ et $\neq 1$	1	0	0
1	1	1	1	1	0	0

Supposons maintenant que la semi-valuation que nous voulons obtenir, soit telle que pour i et j fixés nous ayons $\Phi_i \to \Phi_j = 1$ et $\Phi_j \to \Phi_i = 1$; et considérons le symbole d'équivalence défini, en logique, de telle manière que la formule $A \leftrightarrow B$ soit l'abréviation de $(A \to B) \wedge (B \to A)$. D'après le tableau précédent nous aurons $x \leftrightarrow y = 1$ si et seulement si $x \to y = 1$ et $y \to x = 1$. Soient les règles de déduction suivantes, valables en logique intuitionniste :

$$A \leftrightarrow B \vdash A * C \leftrightarrow B * C \;;\; A \leftrightarrow B \vdash C * A \leftrightarrow C * B$$

($*$ représentant l'un des trois symboles $\to$, $\wedge$, $\vee$). Le théorème de la déduction restant valable en logique intuitionniste, les formules :

$$(A \leftrightarrow B) \to (A * C \leftrightarrow B * C) \;;\; (A \leftrightarrow B) \to (C * A \leftrightarrow C * B)$$

sont des théorèmes en logique intuitionniste et par conséquent doivent pouvoir être vérifiées avec la semi-valuation. En particulier pour $A = \Phi_i$, $B = \Phi_j$ nous devons donc avoir : $\Phi_i * C \leftrightarrow \Phi_j * C = 1$ et $C * \Phi_i \leftrightarrow C * \Phi_j = 1$. Les expressions de la forme $\Phi_i * C$ et $\Phi_j * C$ (de même $C * \Phi_i$ et $C * \Phi_j$) pourront donc prendre soit la même valeur 0, 1 ou Φ_k $(1 \leq k \leq n-2)$ soit deux valeurs différentes Φ_k et Φ_ℓ $(1 \leq k \neq \ell \leq n-2)$ mais telles que $\Phi_k \leftrightarrow \Phi_\ell = 1$, c'est-à-dire telles que $\Phi_k \to \Phi_\ell = 1$ et $\Phi_\ell \to \Phi_k = 1$, ces deux valeurs jouant donc le même rôle que Φ_i et Φ_j. Donc si une formule de logique intuitionniste est donnée et si nous voulons déterminer si elle est identique à 1 ou non, que nous remplacions certaines propositions élémentaires par Φ_i ou par Φ_j nous obtiendrons soit le même résultat soit deux résultats de la forme Φ_k et Φ_ℓ ; si cette formule n'est pas un théorème et si en remplaçant certaines propositions élémentaires par Φ_i nous obtenons, cependant, pour valeur de la formule 1, nous obtiendrons aussi 1 en remplaçant les mêmes propositions élémentaires par Φ_j, il serait donc inutile de considérer les deux valeurs différentes Φ_i et Φ_j et une semi-valuation à n valeurs telle que $\Phi_i \to \Phi_j = 1$ et $\Phi_j \to \Phi_i = 1$ (i et j étant fixés) reviendrait donc à une semi-valuation à (n-1) valeurs où $\Phi_i = \Phi_j$.

Une telle semi-valuation semble donc sans intérêt et par conséquent <u>nous en choisirons une telle que</u>, pour i et j donnés, <u>si $\Phi_i \to \Phi_j = 1$ alors $\Phi_j \to \Phi_i \neq 1$</u>. Ceci entraîne immédiatement, d'après le tableau précédent, que si $x \leftrightarrow y = 1$ alors $x = y$ et réciproquement ; donc, comme $A \wedge A \leftrightarrow A$ et $A \vee A \leftrightarrow A$ sont des théorèmes en logique intuitionniste, $\Phi_i \wedge \Phi_i = \Phi_i = \Phi_i \vee \Phi_i$. De même, les théorèmes $A \vee B \leftrightarrow B \vee A$ et $A \wedge B \leftrightarrow B \wedge A$ nous permettent d'écrire $z_2^i = z_4^i$, $z_5^{ij} = z_5^{ji}$ et $y_5^{ij} = y_5^{ji}$, $y_6^i = y_8^i$. Le théorème $A \vee (B \wedge \neg B) \leftrightarrow A$, nous donne de plus, si $A = \Phi_i$, $B = 1$, $\Phi_i \vee 0 \leftrightarrow \Phi_i = 1$ donc $\Phi_i \vee 0 = \Phi_i$ et par conséquent $0 \vee \Phi_i = \Phi_i$; et de même pour $A = \Phi_i$, $B = 1$, grâce au théorème $A \wedge (A \vee B) \leftrightarrow A$, nous obtenons $\Phi_i \wedge 1 \leftrightarrow 1$ donc $\Phi_i \wedge 1 = \Phi_i$ d'où $1 \wedge \Phi_i = \Phi_i$. La règle de déduction $A \vdash (A \to B) \leftrightarrow B$, entraînant que $A \to ((A \to B) \leftrightarrow B)$ est un théorème, nous permet de montrer que pour $A = 1$, $B = \Phi_i$ $1 \to \Phi_i \leftrightarrow \Phi_i = 1$ donc que $1 \to \Phi_i = \Phi_i$ pour $1 \leq i \leq n-2$.

Résumons, à nouveau, dans un tableau les résultats obtenus à l'aide des <u>hypothèses</u> suivantes : <u>$1 \to y \neq 1$ si $y \neq 1$</u>, <u>$\neg\neg\Phi_i = 1$ pour tout i</u>, <u>si $\Phi_i \to \Phi_j = 1$ alors $\Phi_j \to \Phi_i \neq 1$ pour $1 \leq i \neq j \leq n-2$</u>.

A	B	A → B	A ∧ B	A ∨ B	¬A	¬B
0	0	1	0	0	1	1
0	Φ_i	1	0	Φ_i	1	0
0	1	1	0	1	1	0
Φ_i	0	0	0	Φ_i	0	1
Φ_i	Φ_j	$x_5^{ij} \neq 0$	$y_5^{ij} \neq 0$ et $\neq 1$	$z_5^{ij} \neq 0$	0	0
Φ_i	Φ_i	1	Φ_i	Φ_i	0	0
Φ_j	Φ_i	$x_5^{ji} \neq 0$	$y_5^{ji} = y_5^{ij}$	$z_5^{ji} = z_5^{ij}$	0	0
Φ_i	1	1	Φ_i	1	0	0
1	0	0	0	1	0	1
1	Φ_i	Φ_i	Φ_i	1	0	0
1	1	1	1	1	0	0

Nous remarquons alors que <u>$u \rightarrow t = 1$ est une relation d'ordre sur l'ensemble des valeurs 0, 1 et Φ_i</u> $(1 \leq i \leq n-2)$. En effet pour tous éléments u, t et v de cet ensemble.

- $u \rightarrow u = 1$.
- si $u \rightarrow t = 1$ et $t \rightarrow u = 1$ alors $u = t$.
- si $u \rightarrow t = 1$ et $t \rightarrow v = 1$ alors $u \rightarrow v = 1$

(ceci d'après l'axiome $(A \rightarrow B) \rightarrow ((A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C))$).

De plus <u>cet ensemble est un treillis distributif par rapport aux opérateurs $\vee$ et $\wedge$</u> car ($*$ pouvant être remplacé soit par $\vee$ soit par $\wedge$).

- $u * u = u$.
- $u * t = t * u$.
- $(u * t) * v = u * (t * v)$ (Ceci étant entraîné par les axiomes $(A \wedge B) \wedge C \leftrightarrow A \wedge (B \wedge C)$ et $A \vee (B \vee C) \leftrightarrow A \vee (B \vee C)$).
- $(u \vee t) \wedge u = u \vee (t \wedge u) = u$ (nous obtenons cela en considérant les axiomes $A \vee (A \wedge B) \leftrightarrow A$ et $A \wedge (A \vee B) \leftrightarrow A$)
- $u \wedge (t \vee v) = (u \wedge t) \vee (u \wedge v)$ (Axiome $A \wedge (B \vee C) \leftrightarrow (A \wedge B) \vee (B \wedge C)$).

Il serait intéressant qu'une semi-valuation ainsi choisie soit de plus en plus puissante si n augmente. Or nous avons vu qu'aucune valuation n'est possible, car, en particulier, l'expression $\bigcup_{1 \leq \ell < k \leq n} (A_\ell \rightarrow A_k)$ qui n'est plus un théorème en logique intuitionniste prend cependant la valeur 1 si nous considérons que $1 \rightarrow y \neq 1$ si $y \neq 1$, et, si nous nous plaçons dans le cas d'une semi-valuation à (n-1) valeurs. Nous voudrions donc, pour n valeurs, ne plus pouvoir identifier cette formule à 1. Pour cela il faut qu'aucune des implications $A_\ell \rightarrow A_k$ $(\ell < k)$ soit égale à 1 pour au moins une des combinaisons possibles des n valeurs associées aux n propositions élémentaires A_m $(m = \ell$ ou $m = k)$. Une combinaison possible associée aux propositions A_m devra donc être telle que $A_k \neq 1$ (pour $k = 2, 3, \ldots, n-1, n$) $A_\ell \neq 0$ (pour $\ell = 1, 2, \ldots, n-2, n-1$) et

$A_\ell \neq A_k$ pour tous ℓ et k. Donc la seule possibilité est d'associer 1 à A_1 et 0 à A_n, la valeur Φ_i associée à une proposition A_ℓ $(2 \leq \ell \leq n-1)$ devra être telle que $\Phi_i \to \Phi_j \neq 1$ si Φ_j est associée à A_k avec $\ell < k$. Nous pouvons, par exemple, associer Φ_1 à A_{n-1}, Φ_2 à $A_{n-2}, \ldots, \Phi_{n-3}$ à A_3, Φ_{n-2} à A_2 en choisissant la semi-valuation telle que $\Phi_i \to \Phi_j \neq 1$ pour $i > j$ (toutes les permutations possibles de $\Phi_1, \ldots, \Phi_{n-2}$ sont aussi convenables si nous choisissons la semi-valuation correspondante, mais elle ne présentera pas un plus grand intérêt, jusqu'ici aucune des valeurs Φ_j ne se différentiant par rapport aux autres). Ceci et l'axiome $A \wedge B \to A$ entraîne que $\Phi_1 \wedge \Phi_j \to \Phi_1 = 1$, et, par conséquent que $\Phi_1 \wedge \Phi_j = \Phi_j \wedge \Phi_1 = \Phi_1$ $(1 \leq j \leq n-2)$; et de même, l'axiome $A \wedge B \to B$, nous permet alors d'écrire $\Phi_1 \to \Phi_j = 1$ $(1 \leq j \leq n-2)$. Nous en déduisons à l'aide du théorème $A \vee (A \wedge B) \leftrightarrow A$, que $\Phi_j \vee (\Phi_j \wedge \Phi_1) \leftrightarrow \Phi_j = 1$ donc que $\Phi_j \vee \Phi_1 = \Phi_j$ $(1 \leq j \leq n-2)$.

Si nous voulons, de plus, qu'une semi-valuation de n valeurs nous permette de reconnaître comme non théorèmes toutes les formules reconnues comme non théorèmes par une semi-valuation de moins de n valeurs, nous sommes conduits à choisir la semi-valuation telle que pour (n-1) valeurs quelconques nous obtenions les mêmes résultats que ceux obtenus dans le cas de la semi-valuation à (n-1) valeurs. Ceci nous entraînerait à choirir $\Phi_2 \to \Phi_j = 1$ $(3 \leq j \leq n-2)$; $\Phi_3 \to \Phi_j = 1$ $(4 \leq j \leq n-2)$ et de manière générale $\underline{\Phi_i \to \Phi_j = 1}$ $\underline{\text{pour } 1 \leq i < j \leq n-2}$, c'est-à-dire <u>à considérer $u \to t = 1$ comme une relation d'ordre total</u>. Nous avons alors $\Phi_i \vee \Phi_j = \Phi_j$ et $\Phi_i \wedge \Phi_j = \Phi_i$ pour $1 \leq i < j \leq n-2$. En effet l'axiome $(A \to C) \to ((B \to C) \to (A \vee B \to C))$, pour $A = C = \Phi_j$, $B = \Phi_i$, interdit que $\Phi_i \vee \Phi_j = 1$ et entraîne $\Phi_i \vee \Phi_j = \Phi_k$ avec $k \leq j$, et, d'autre part l'axiome $A \to A \vee B$ donne $\Phi_j \to \Phi_k = 1$ c'est-à-dire $j \leq k$, donc nous obtenons bien $\Phi_i \vee \Phi_j = \Phi_j$. De plus le théorème $A \wedge (A \vee B) \leftrightarrow A$ nous permet d'écrire, si $\Phi_i \vee \Phi_j = \Phi_j$, que $\Phi_i \wedge \Phi_j = \Phi_j \wedge \Phi_i = \Phi_i$ (Ces deux égalités se vérifient aussi très facilement en utilisant le fait que l'ensemble des valeurs 0, 1, Φ_k ($1 \leq k \leq n-2$) est un treillis par rapport aux opérateurs $\vee$ et $\wedge$, donc que $\Phi_i \vee \Phi_j$ et $\Phi_i \wedge \Phi_j$ sont respectivement le plus petit majorant et le plus grand minorant de Φ_i et Φ_j. De plus comme $\Phi_i \vee \Phi_j \neq 1$ pour tous i et j, l'ensemble des valeurs Φ_k seules est un sous-treillis de l'ensemble précédent).

Remarquons, dès maintenant, que l'hypothèse $\Phi_i \to \Phi_j = 1$, pour $1 \leq i < j \leq n-2$, nous a permis d'écrire $\Phi_i \vee \Phi_j \neq 1$. Donc la condition nécessaire $\Phi_j \to \Phi_i \neq 1$ (qui est d'ailleurs entraînée par l'hypothèse précédente), pour que l'expression $\bigcup_{1 \leq \ell < k \leq n} (A_\ell \to A_k)$ ne soit pas identifiée à 1, devient suffisante et la semi-valuation choisie est bien alors de plus en plus puissante si n augmente.

Il nous reste à déterminer les valeurs de $\Phi_j \to \Phi_i$ pour $1 \leq i < j \leq n-2$. Pour cela considérons l'axiome $A \to (B \to A)$ et écrivons $\Phi_i \to (\Phi_j \to \Phi_i) = 1$, $\Phi_j \to \Phi_i$ peut donc prendre une valeur Φ_k telle que $i \leq k$; mais d'après l'axiome $(A \to B) \to ((A \to (B \to C)) \to (A \to C))$ nous avons

$(\Phi_j \to \Phi_j) \to ((\Phi_j \to (\Phi_j \to \Phi_i)) \to (\Phi_j \to \Phi_i)) = 1$ donc $(\Phi_j \to (\Phi_j \to \Phi_i)) \to (\Phi_j \to \Phi_i) = 1$

et comme $(\Phi_j \to \Phi_i) \to (\Phi_j \to (\Phi_j \to \Phi_i))' = 1$ (d'après l'axiome $A \to (B \to A)$) nous devons donc avoir $\Phi_j \to \Phi_i = \Phi_j \to (\Phi_j \to \Phi_i)$; si $\Phi_j \to \Phi_i = \Phi_k$ ($i \leq k$) k devra être tel que $j > k \geq i$ sinon $\Phi_j \to \Phi_k = 1$ ce qui n'est pas possible puisque $\Phi_j \to \Phi_i = \Phi_k \neq 1$. De plus l'axiome précédent nous permet encore d'écrire $(1 \to \Phi_j) \to ((1 \to (\Phi_j \to \Phi_i)) \to (1 \to \Phi_i)) = 1$, c'est-à-dire $\Phi_j \to ((1 \to (\Phi_j \to \Phi_i)) \to \Phi_i) = 1$; si $\Phi_j \to \Phi_i = \Phi_k$ et si $j > k > i$ alors $\Phi_j \to (\Phi_k \to \Phi_i) = 1$ et d'après ce qui précède $\Phi_k \to \Phi_i = \Phi_\ell$ avec $k > \ell \geq i$ donc $j > \ell$ et $\Phi_j \to \Phi_\ell \neq 1$.

Donc nous ne pouvons pas prendre $\Phi_j \to \Phi_i = \Phi_k$ si $k > i$, et, la seule possibilité est $\Phi_j \to \Phi_i = \Phi_i$ $(n-2 \geq j > i \geq 1)$.

Finallement, en prenant comme hypothèses que : $1 \to y \neq 1$ si $y \neq 1$; $\neg\neg\Phi_i = 1$ $(1 \leq i \leq n-2)$; si $\Phi_i \to \Phi_j = 1$ alors $\Phi_j \to \Phi_i \neq 1$ $(1 \leq i < j \leq n-2)$; $\Phi_i \to \Phi_j = 1$ $(1 \leq i < j \leq n-2)$, nous remarquons que nous obtenons une semi-valuation à n valeurs, généralisation de la valuation de la logique classique par l'algèbre de Boole. En effet, si nous posons $0 < \Phi_1 < \ldots < \Phi_{n-2} < 1$, nous pouvons écrire :

$$\begin{cases} x \vee y = \max(x, y). \\ x \wedge y = \min(x, y). \\ x \to y = 1 \quad \text{si} \quad x \leq y \;;\; x \to y = y \quad \text{si} \; x > y. \\ \neg x = 0 \quad \text{si} \quad x \neq 0 \text{ et } \neg 0 = 1. \end{cases}$$

Cette semi-valuation nous place d'ailleurs dans le cas de l'algèbre de Post à n valeurs (l'union et l'intersection correspondant respectivement à la somme et au produit ; 0 correspondant à 0, Φ_1 à 1, Φ_2 à 2,..., Φ_{n-2} à n-2, 1 à n-1).

Nous constatons facilement, qu'avec cette semi-valuation, tous les axiomes du système de Kleene par exemple, prennent la valeur 1, et, qu'évidemment la règle de déduction est vérifiée. Par conséquent tous les théorèmes de la logique intuitionniste pourront être identifiés à 1.

III - Propriétés de la semi-valuation choisie :

a) Cette semi-valuation valable pour un nombre fini quelconque de valeurs ne donnera certainement pas la valeur 1 pour des formules qui ne sont pas des théorèmes en logique classique.

En effet dans le cas contraire, la formule considérée serait égale à 1 lorsque nous associons aux propositions élémentaires les différentes combinaisons possibles des n valeurs, en particulier des valeurs 0 et 1 uniquement. Or d'après la semi-valuation choisie, les résultats sont les mêmes quand nous appliquons les opérateurs à ces deux valeurs particulières que nous soyons dans le cas de n valeurs ($n > 2$) ou de deux valeurs, donc la formule serait identique à 1 par la valuation de la logique classique, ce qui est impossible. (Les deux valeurs 0 et 1 sont effectivement les mêmes pour tout n, l'une représentant 'vrai' et l'autre 'faux' par convention).

b) A partir de $n = 3$, cette semi-valuation nous permet d'écrire de nombreux théorèmes de la logique classique comme non identiques à 1, (pour $n = 2$, tous les théorèmes de la logique classique prennent, évidemment, la valeur 1 et ce cas ne nous permet aucune différentiation entre les deux logiques) en particulier $\neg\neg A \rightarrow A$ et $A \vee \neg A$, axiomes ou théorèmes correspondant au principe du tiers exclus, ce qui présente un certain intérêt, étant donné l'obtention des systèmes d'axiomes de la logique intuitionniste à partir de ceux de la logique classique. Nous pouvons citer comme autres exemples de théorèmes classiques non identifiés à 1 :

$A \wedge (B \vee \neg B) \leftrightarrow A$; $A \vee B \vee \neg B \leftrightarrow B \vee \neg B$; $A \vee B \leftrightarrow \neg(\neg A \wedge \neg B)$;...

c) Nous avons vu que cette semi-valuation est telle que, pour un n donné, il existe au moins une formule qui n'est pas un théorème et qui n'est plus identique à 1, alors qu'elle l'était pour tout m inférieur à n.

Mais nous rencontrerons évidemment des formules logiques qui ne sont pas des théorèmes intuitionnistes et qui, cependant, restent identiques à 1 pour tout n fini, comme par exemple :

$\neg A \vee \neg\neg A$; $\neg(A \wedge B) \leftrightarrow \neg A \vee \neg B$;... Nous remarquons que, dans ces formules, l'opérateur de plus grande portée relie des expressions sous forme de négation. Il est donc évident que nous obtiendrons une identité à 1 puisque les expressions sous forme de négation prendront alors uniquement une des valeurs 0 et 1 et que les résultats dûs aux autres opérateurs seront alors obligatoirement 0 et 1 d'après la semi-valuation considérée. D'autre part le résultat ne peut être 0 puisque si E est la formule donnée, $\neg\neg E$ est un théorème en logique intuitionniste et puisque tout théorème intuitionniste est identique à 1 avec la semi-valuation utilisée.

- CHAPITRE IV -

- RECONNAISSANCE SYSTEMATIQUE D'EXPRESSIONS IDENTIQUEMENT VRAIES POUR UNE LOGIQUE A 3 VALEURS -

I - Lien entre une logique non classique et l'algèbre de Post :

Nous avons vu au chapitre précédent qu'il peut être établi une correspondance partielle entre la logique intuitionniste et l'algèbre de Post à n valeurs, notées $0, \Phi_1, \ldots, \Phi_{n-2}, 1$ pour conserver à la valeur 1 sa signification de 'vrai' habituellement considérée en algèbre de Boole. Cette même propriété peut encore être exacte dans d'autres cas de suppression d'axiomes de la logique classique. Un exemple de ceci est en particulier donné par Wang qui étudie l'indépendance de l'axiome $A \to (B \to A)$.

Il peut donc y avoir un certain intérêt à savoir déterminer systématiquement si une expression de l'algèbre de Post est identique à 1 ou non. Nous allons donc généraliser la méthode employée dans le cas de la logique classique et de l'algèbre de Boole. Nous le ferons, en particulier, pour 3 valeurs, ce qui s'étendra immédiatement à n valeurs (n fini et supérieur à 3).

II - Utilisation du théorème de Lagrange en algèbre de Post :

Toute fonction f, en algèbre de Post à n valeurs, peut s'écrire par rapport à une de ces variables sous la forme :

$$f(Y, x) = \sum_{k=0}^{n-1} f_k(Y)\, L_k(x).$$

où $L_k(x)$ est une fonction telle que $L_k(k) = n-1$ et $L_k(x) = 0$ si $x \neq k$.

Dans le cas de l'algèbre de Post à 3 valeurs $(0, \Phi, 1)$ ce théorème s'écrit :

$$f(Y, x) = f_0(Y)\, L_0(x) + f_\Phi(Y)\, L_\Phi(x) + f_1(Y)\, L_1(x).$$

Nous pouvons en raisonnant de même que dans le cas de l'algèbre de Boole, prouver qu'une expression quelconque est identique à 1 en nous servant de cette nouvelle forme de Lagrange.

En effet $f(x) = f_0\, L_0(x) + f_\Phi\, L_\Phi(x) + f_1\, L_1(x)$ ne peut être identique à 1 (ou à 0) que si et seulement si $f_0 \equiv f_\Phi \equiv f_1 \equiv 1$ (ou $f_0 \equiv f_\Phi \equiv f_1 \equiv 0$). D'autre part une transformation fonctionnelle quelconque de $f(x)$ se traduira, aussi en algèbre de Post, par la même transformation sur les coefficients puisque ceux-ci sont encore les valeurs de la fonction.

Donc pour identifier une expression à 1 en algèbre de Post, nous la ramènerons, avec une méthode analogue à celle utilisée en algèbre de Boole, systématiquement à la forme :

$$\boxed{1 \equiv 1\, L_0(x) + 1\, L_\Phi(x) + 1\, L_1(x)} \qquad (1)'$$

III - Relations entre les différents opérateurs utilisés en logique intuitionniste et exprimés en algèbre de Post :

Il serait intéressant d'obtenir entre les différents opérateurs des relations analogues à celles existant en algèbre de Boole, et de pouvoir exprimer toute expression, contenant ces opérateurs, en fonction de la somme, du produit et de la négation par exemple. Remarquons, tout de suite, que si ces relations existent, nous pourrons seulement affirmer qu'elles sont valables en algèbre de Post mais non obligatoirement en logique intuitionniste puisqu'il n'y a pas équivalance entre celle-ci et l'algèbre précédente.

Nous pouvons vérifier tout d'abord, en considérant les tables des opérateurs $\wedge$, $\vee$ et $\neg$, que les deux relations suivantes, valables en algèbre de Boole, se conservent :

$$\neg(A \wedge B) = \neg A \vee \neg B \quad ; \quad \neg(A \vee B) = \neg A \wedge \neg B$$

(ou $\neg AB = \neg A + \neg B$; $\neg(A + B) = \neg A \neg B$ en écriture habituelle de l'algèbre de Post. Ces égalités restent vraies pour n valeurs).

Ces deux relations permettent de considérer les expressions, de l'algèbre de Post, contenant uniquement ces trois opérateurs, sous forme de somme de produits de variables directes ou niées.

Soit maintenant une implication $A \rightarrow B$ et essayons de l'écrire sous forme de somme de produits. Par analogie avec l'algèbre de Boole nous allons comparer cette expression à $\neg A + B$. Ecrivons le tableau des valeurs :

A	B	$A \rightarrow B$	$\neg A + B$
0	0	1	1
0	Φ	1	1
0	1	1	1
Φ	0	0	0
Φ	Φ	1	Φ
Φ	1	1	1
1	0	0	0
1	Φ	Φ	Φ
1	1	1	1

Les valeurs de $A \rightarrow B$ et de $\neg A + B$ différent seulement pour $A = B = \Phi$. $A \rightarrow B$ doit être égale, dans ce cas à 1, il faudrait donc ajouter à $\neg A + B$ une quantité égale à 1, pour $A = B = \Phi$, et à 0 pour $A = \Phi$, $B = 0$ et pour $A = 1$, $B = 0$. Or la fonction $L_\Phi(A) = 1$ si et seulement si $A = \Phi$, et, est égale à 0 pour les autres valeurs de A. Donc le produit $L_\Phi(A)\ L_\Phi(B)$ sera égale à 1 si et seulement si $A = B = \Phi$ et à 0 dans tous les autres cas. Nous pouvons donc écrire : $\underline{A \rightarrow B = \neg A + B + L_\Phi(A)\ L_\Phi(B)}$.

(Pour n valeurs nous obtiendrions de même

$$A \rightarrow B = \neg A + B + \sum_{1 \leq i \leq j \leq n-2} L_{\Phi_i}(A)\ L_{\Phi_j}(B).).$$

Remarquons que :

$$\neg A \rightarrow B = \neg\neg A + B + L_\Phi(\neg A)\ L_\Phi(B).$$

Or $L_\Phi(\neg A) = 0$ donc $\neg A \rightarrow B = \neg\neg A + B$.

De même : $A \rightarrow \neg B = \neg A + \neg B + L_\Phi(A)\ L_\Phi(\neg B) = \neg A + \neg B$.

Donc dès que nous rencontrerons une implication, portant sur au moins une expression sous forme de négation, nous pourrons la remplacer par une somme.

IV - Méthode pour identifier une expression à 1 :

a) Une expression donnée, avec les opérateurs utilisés en logique, sera transformée sous forme de sommes de produits en utilisant les relations du paragraphe précédent plus celle relative à l'opérateur d'équivalence, c'est-à-dire :

$$A \leftrightarrow B = (A \rightarrow B)_\wedge(B \rightarrow A) = (\neg A + B + L_\Phi(A) L_\Phi(B))(\neg B + A + L_\Phi(A) L_\Phi(B)).$$

$$A \leftrightarrow B = AB + \neg A \neg B + L_\Phi(A)\ L_\Phi(B).$$

b) Nous nous ramènerons systématiquement à la forme (1)' en utilisant les égalités suivantes :

$$
(\mathrm{I}) \quad \left\{
\begin{array}{l}
f(A) * g(A) = (f_0 * g_0)\, L_0(A) + (f_\Phi * g_\Phi)\, L_\Phi(A) + (f_1 * g_1) L_1(A). \\
(* \text{ représentant une somme ou un produit}). \\
\neg\, f(A) = (\neg\, f_0)\, L_0(A) + (\neg\, f_\Phi)\, L_\Phi(A) + (\neg\, f_1)\, L_1(A). \\
L_\Phi(f(A)) = L_\Phi(f_0)\, L_0(A) + L_\Phi(f_\Phi)\, L_\Phi(A) + L_\Phi(f_1)\, L_1\,(A). \\
A = 0\, L_0\,(A) + \Phi\, L_\Phi\,(A) + 1\, L_1\,(A). \\
B = B\, L_0(A) + B\, L_\Phi(A) + B\, L_1\,(A). \qquad \text{si } B \neq A.
\end{array}
\right.
$$

c) Si la formule considérée contient plusieurs lettres nous l'écrivons sous forme d'expression de Lagrange par rapport à la première, A par exemple, à l'aide de (I). Nous étudions alors les coefficients obtenus par rapport à B et ainsi de suite jusqu'à ce que nous ayons épuisé toutes les lettres si nécessaire. Si l'expression de départ est identique à 1 les formules intermédiaires puis la forme finale seront alors identiques à (1)'.

Remarque : Une expression identique à 0, évidemment, se ramènera de même à une expression de la forme $0\, L_0(A) + 0\, L_\Phi(A) + 0\, L_1(A)$.

Une expression ni identique à 1 ni identique à 0 se trouvera, avec cette méthode, de même qu'en algèbre de Boole, écrite sous forme lexicographique en algèbre de Post.

V - Exemples d'utilisation de cette méthode :

1) Vérifions tout d'abord que certains théorèmes valables en logique classique et non en logique intuitionniste, ne se ramèneront pas à 1 en algèbre de Post :

a) $A \vee \neg A$ ou $A + \neg A$ en algèbre de Post.

$$A = 0\ L_0(A) + \Phi\ L_\Phi(A) + 1\ L_1(A).$$

$$\neg A = 1\ L_0(A) + 0\ L_\Phi(A) + 0\ L_1(A).$$

$$A + \neg A = 1\ L_0(A) + \Phi\ L_\Phi(A) + 1\ L_1(A) \not\equiv 1.$$

b) $(((A \to B) \to A) \to A)$:

$$A \to B = \neg A + B + L_\Phi(A)\ L_\Phi(B) = 1\ L_0(A) + (B + L_\Phi(B))\ L_\Phi(A) + B\ L_1(A).$$

$$(A \to B) \to A = \neg (A \to B) + A + L_\Phi(A)\ L_\Phi(A \to B).$$

$$= 0\ L_0(A) + (\neg (B + L_\Phi(B)) + \Phi)\ L_\Phi(A) + 1\ L_1(A).$$

$$= 0\ L_0(A) + (\neg B \neg L_\Phi(B) + \Phi)\ L_\Phi(A) + 1\ L_1(A).$$

$$((A \to B) \to A) \to A = \neg ((A \to B) \to A) + A + L_\Phi(A)\ L_\Phi((A \to B) \to A)$$

$$= 1\ L_0(A) + ((\neg \neg B + L_\Phi(B))\Phi + \Phi + L_\Phi(\neg B \neg L_\Phi(B) + \Phi))L_\Phi(A) + 1\ L_1(A).$$

$$= 1\ L_0(A) + (\Phi + L_\Phi(\neg B \neg L_\Phi(B) + \Phi))L_\Phi(A) + 1\ L_1(A).$$

Etudions $\Phi + L_\Phi(\neg B \neg L_\Phi(B) + \Phi)$ par rapport à B :

$$\neg B \neg L_\Phi(B) + \Phi = 1\ L_0(B) + \Phi\ L_\Phi(B) + \Phi\ L_1(B).$$

$$\Phi + L_\Phi(\neg B \neg L_\Phi(B) + \Phi) = \Phi\ L_0(B) + 1\ L_\Phi(B) + 1\ L_1(B).$$

Et :

$$(((A \to B) \to A) \to A) = 1\ L_0(A) + (\Phi L_0(B) + 1\ L_\Phi(B) + 1\ L_1(B))L_\Phi(A) + 1\ L_1(A) \not\equiv 1.$$

<u>Remarque</u> : si une formule E est un théorème en logique classique, mais n'en est plus un en logique intuitionniste, nous n'obtiendrons aucun coefficient égal à 0 car $\neg\neg E$ doit être identique à 1. Et réciproquement si aucun coefficient n'est égal à 0, la formule étudiée est un théorème en logique classique car, alors, $\neg\neg E \equiv 1$ et $\neg\neg E$ est un théorème en logique intuitionniste (et en logique classique) si et seulement si E est un théorème en logique classique.

2) Considérons maintenant un théorème de la logique classique contenant seulement les opérateurs $\neg$ et $\wedge$: $\neg(A \wedge \neg A)$.

En algèbre de Post nous avons :

$\neg(A.\neg A) = \neg A + \neg\neg A$

$\neg A = 1\ L_0(A) + 0\ L_\Phi(A) + 0\ L_1(A).$

$\neg\neg A = 0\ L_0(A) + 1\ L_\Phi(A) + 1\ L_1(A).$

$\neg A + \neg\neg A = 1\ L_0(A) + 1\ L_\Phi(A) + 1\ L_1(A) \equiv 1.$

Ce qui nous permet effectivement de dire que $\neg(A \wedge \neg A)$ est un théorème classique mais non de conclure que cette formule est ou non un théorème en logique intuitionniste. Cependant dans ce cas nous pourrons répondre par l'affirmative grâce à un théorème cité par Kleene :

Si E est une formule de propositions contenant aucun symbole logique excepté $\wedge$ et $\neg$ et si E est un théorème dans le calcul classique des propositions, alors E est un théorème dans le calcul intuitionniste des propositions.

Remarquons que, bien que $\neg(A \wedge \neg A)$ et $\neg A \vee \neg\neg A$, du point de vue de l'algèbre de Post, peuvent être considérées comme des formes différentes d'une même expression, elles ne sont cependant pas toutes les deux des théorèmes en logique intuitionniste.

3) Pour vérifier une règle de déduction nous pourrions en algèbre de Post, pour les mêmes raisons qu'en calcul booléen, la transformer en implication généralisée. Mais il est évident que si nous obtenons une identité à 1 nous pourrons seulement conclure que la règle est valable en logique classique et que, dans le cas contraire, elle n'est valable dans aucune des deux logiques.

Soit $\neg A \vdash A \rightarrow B \leftrightarrow \neg A$. Nous obtenons en algèbre de Post :

$\neg A \rightarrow (A \rightarrow B \leftrightarrow \neg A)$ et sous-forme de somme de produits :

$$R = \neg\neg A + (\neg A + B + L_{\Phi}(A)L_{\Phi}(B))\neg A + \neg(\neg A + B + L_{\Phi}(A)L_{\Phi}(B))\neg\neg A + L_{\Phi}(\neg A + B + L_{\Phi}(A)L_{\Phi}(B))L_{\Phi}(\neg A)$$

$$+ L_{\Phi}(\neg A)L_{\Phi}((\neg A + B + L_{\Phi}(A)L_{\Phi}(B))\neg A + \neg(\neg A + B + L_{\Phi}(A)L_{\Phi}(B))\neg\neg A + L_{\Phi}(\neg A + B + L_{\Phi}(A)L_{\Phi}(B))L_{\Phi}(\neg A))$$

Or : $\neg A = 1\ L_0(A) + 0\ L_{\Phi}(A) + 0\ L_1(A)$.

$\neg\neg A = 0\ L_0(A) + 1\ L_{\Phi}(A) + 1\ L_1(A)$.

$P = \neg A + B + L_{\Phi}(A)L_{\Phi}(B) = 1\ L_0(A) + (B + L_{\Phi}(B))\ L_{\Phi}(A) + B\ L_1(A)$.

$P\neg A = 1\ L_0(A) + 0\ L_{\Phi}(A) + 0\ L_1(A)$.

$\neg P\neg\neg A = 0\ L_0(A) + \neg(B + L_{\Phi}(B))\ L_{\Phi}(A) + \neg B\ L_1(A)$.

$L_{\Phi}(P)\ L_{\Phi}(\neg A) = 0\ L_0(A) + 0\ L_{\Phi}(A) + 0\ L_1(A)$.

$P\neg A + \neg P\neg\neg A + L_{\Phi}(P)L_{\Phi}(\neg A) = 1\ L_0(A) + \neg(B + L_{\Phi}(B))\ L_{\Phi}(A) + \neg B\ L_1(A)$.

$R = 1\ L_0(A) + 1\ L_{\Phi}(A) + 1\ L_1(A) \equiv 1$

La règle de déduction considérée est donc valable en logique classique mais le fait d'avoir obtenu une identité à 1 n'entraîne pas sa validité en logique intuitionniste. Nous ne pouvons rien dire non plus sur la formule

$\neg A \rightarrow (A \rightarrow B \leftrightarrow \neg A)$, à laquelle nous ne pouvons pas appliquer un théorème analogue à celui cité à l'exemple précédent.

<u>Remarques</u> : 1 - Nous n'avons étudié l'identité à 1 de formules logiques que dans le cas de trois valeurs. Nous aurions pu faire de même pour un nombre n quelconque de valeurs mais la recherche, à chaque pas, de n coefficients deviendrait vite très longue. De plus ceci ne paraît pas très utile car, bien que la semi-valuation, par l'algèbre de Post, soit de plus en plus puissante si n croît, le nombre de formules, qui ne sont pas des théorèmes et qui ne restent pas identiques à 1 pour n alors qu'elles le sont pour tout m inférieur à n, semble assez faible. Par exemple sur 97 théorèmes de la logique classique, cités dans Introduction to Metamathematics de Kleene, quatorze ne le sont plus en logique intuitionniste, douze ne s'écrivent pas comme identiques à 1 en algèbre de Post et les deux autres le restent pour tout n.

2 - Si nous ne voulons pas uniquement chercher à identifier à 1 des formules logiques écrites en algèbre de Post, mais aussi des expressions quelconques de cette algèbre, l'opérateur de glissement pourra intervenir. Cet opérateur est défini de la manière suivante :

A	A_1
0	Φ
Φ	1
1	0

Nous ne pouvons pas l'exprimer simplement en fonction des autres opérateurs considérés. Aussi quand nous étudierons des expressions le contenant nous devrons ajouter la règle supplémentaire :

$$A_1 = \Phi\ L_0(A) + 1\ L_\Phi(A) + 0\ L_1(A).$$

D'autre part nous aurions pu introduire cet opérateur dans l'écriture de l'implication en fonction de la somme et du produit. En effet $A \rightarrow B = \neg A + B + \neg A_2 \neg B_2$. Cependant il semble préférable de considérer les fonctions $L_\Phi(A)$ et $L_\Phi(B)$ puisque, de toute manière, elles apparaîssent dans l'écriture de l'expression sous forme de Lagrange.

- BIBLIOGRAPHIE -

1 - CARVALLO M : Monographie des treillis et algèbre de Boole. Gauthier - Villars - 1962.

2 - CHURCH A : Introduction to Mathematical Logic. Princeton - 1956.

3 - GÖDEL K : Zum intuitionionistischen Aussagenkalkül. Akademie der Wissenschaften in Wien, Mathematisch - naturwissenschaftliche Klasse, Anzeiger, Vol. 69 (1932), pp. 65-66. Reprinted in Ergebnisse eines mathematischen Kolloquiums, Helft 4 (for 1931-2, pub. 1933), p. 40.

4 - KLEENE S.C : Introduction to Metamathematics. New-York - 1952.

5 - KUNTZMANN J : Algèbre de Boole - Dunod - 1964.

6 - KUNTZMANN J : Théorie des réseaux-Cours de la faculté des Sciences de Grenoble.

7 - LORENZEN P : Formal Logic - Dordrecht-Holland - 1965.

8 - QUINE W Van Orman : Méthods of Logic. Revised Edition. New-York - Septembre 1962.

9 - RASIOWA H et SIKORSKI R : The Mathematics of Metamathematics. Varsovie 1963.

10 - VAUQOIS B : Cours de Logique et Programmation. Faculté des Sciences de Grenoble.

11 - WANG Hao : A survey of Mathematical Logic-North Holland - 1963.

12 - WANG Hao : Toward Mechanical Mathematics-I.B.M. journal janvier 1960. pp. 2-22.

VU

Grenoble, le

Le Président de la Thèse

VU

Grenoble, le

Le Doyen de la Faculté des Sciences

VU, et permis d'imprimer,

Le Recteur de l'Académie de GRENOBLE