Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Etude, résolution et optimisation de modèles macro-économiques. Application au modèle belge Sandy

NOEL, Marie-Cécile

Award date:
1981

Awarding institution:
Universite de Namur

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 02. May. 2026

FACULTES UNIVERSITAIRES N.D. DE LA PAIX
NAMUR

FACULTE DES SCIENCES

ETUDE, RESOLUTION ET OPTIMISATION DE MODELES MACRO-ECONOMIQUES.

APPLICATION AU MODELE BELGE SANDY.

Mémoire présenté pour l'obtention
du grade de Licencié en Sciences
mathématiques
par

Marie-Cécile NOEL

Année Académique 80-81

Promoteur : NGUYEN Van Hien

La réalisation d'un mémoire de type "applications réelles" fut pour moi un travail agréable plutôt qu'une contrainte.

Elle me permit d'avoir un premier contact réel avec la vie professionnelle et en même temps de mieux saisir des problèmes économiques.

Je tiens à remercier vivement toutes les personnes qui m'ont aidée au cours de ce travail enrichissant :

Monsieur Henri BOGAERT, qui au nom du Bureau du Plan, a proposé ce sujet et m'a fourni les explications et renseignements d'ordre économique nécessaires au cours de son évolution;

Monsieur NGUYEN Van Hien, qui m'a fourni les méthodes mathématiques efficaces pour traiter le sujet proposé par le Bureau du Plan et encouragée tout au cours de la réalisation;

Monsieur Hubert CLAES, qui m'a aidée au niveau informatique;

Madame MARCHOUL, pour le soin qu'elle a apporté à la dactylographie de ce travail.

AVERTISSEMENT.

Le problème résolu dans ce travail se pose concrètement au Bureau du Plan.

C'est donc sous la direction de Monsieur NGUYEN Van Hien, Professeur à l'Unité d'Optimisation aux Facultés Notre-Dame de la Paix et Monsieur BOGAERT Henri, Expert du Bureau du Plan, que ce mémoire a été réalisé.

TABLE DES MATIERES

		Page
INTRODUCTION		1
CHAPITRE 1 :	FORMULATION DU PROBLEME	3
	1) Formulation économique générale	3
	2) Application au modèle SANDY	4
CHAPITRE 2 :	ETUDE DE LA STRUCTURE DU MODELE	8
	1. Notations	8
	a) Généralités	8
	b) Cas particulier de SANDY	9
	2. Décomposition d'un système d'équations en blocs d'interdépendance	10
	a) Rappel de la théorie des graphes	10
	b) Décomposition d'un système d'équations	11
	1°) Graphe associé à un système d'équations	12
	2°) Interprétation du graphe associé	12
	3°) Décomposition en niveaux d'un graphe sans circuit	13
	4°) Décomposition en blocs d'un système dont le graphe associé contient des circuits	14
	5°) Notion de matrice d'incidence	17
	c) Décomposition en blocs du modèle SANDY	19
	3. Structure quasi-triangulaire d'un modèle macroéconomique	21
	a) Notion de variables de bouclage	21
	b) Différentes parties d'un modèle	23
	c) Mise du modèle sous forme quasi-triangulaire	25
	1°) Algorithme de Ravelli	26
	2°) Commentaires	29
	3°) Exemples	33
	4°) Résultats sur SANDY	36

Page

CHAPITRE 3 : RESOLUTION DU MODELE

1. Réduction de la taille du système 38
2. Résolution d'un système non linéaire par une méthode itérative 39
   a) Généralités 39
   b) Méthode de Newton et ses variantes 41
   c) Méthode de Gauss-Seidel 49
   d) Méthode des sécantes 43
   e) Méthode de Brown 44
   f) Conclusions 46
3. Résolution du modèle SANDY 46
4. Suggestions 49

CHAPITRE 4 : OPTIMISATION DU MODELE 51

1. Conditions d'existence et d'unicité de la solution 51
2. Conditions d'optimalité 55
3. Méthodes numériques d'optimisation sans contrainte 57
   a) Idée générale 57
   b) Méthode du gradient 59
   c) Méthode du gradient conjugué 60
   d) Méthodes quasi-newton 61
4. Calcul du gradient 66
5. Méthode de pénalisation 67
6. Suggestions 69

CHAPITRE 5 : RESULTATS 71

1. Référence initiale 72
2. Simulation : 3 objectifs 76
3. Simulation ou l'objectif, exportations nettes, est privilégié 80
4. Simulation où l'objectif, chômage, est privilégié 83
5. Simulation où l'objectif, déficit de l'Etat, est privilégié 86
6. Tableau récapitulatif 89

ANNEXES

REFERENCES

INTRODUCTION

L'étude, la résolution et l'optimisation des modèles macroéconomiques, sont l'objet de ce travail dont le but est essentiellement l'application des méthodes proposées par Gabay et al [2] . Elles sont générales et peuvent être appliquées à n'importe quel modèle macroéconomique. Nous avons, en particulier, étudié, résolu et optimisé le modèle SANDY, developpé pour la Belgique par T. de Biolley.

L'étude de la politique économique devrait permettre de déterminer la valeur de certains instruments (taux de taxations, taux de cotisations de sécurité sociale...) en vue de réaliser des objectifs (minimiser le chômage, le déficit de l'Etat, celui de la balance des paiements).

Le comportement de l'économie est décrit par un modèle, qui est un système d'équations dans lequel apparaissent deux sortes de variables : des variables endogènes (variables d'état) et des instruments (variables de contrôle).

Les techniques de simulation des modèles permettent de répondre à la question : "Quelles valeurs obtient-on pour les variables endogènes, si on donne des valeurs aux intruments?". Elles ne fournissent pas de réponse à la question inverse "Quelles valeurs optimales faut-il donner aux instruments, si on veut que les variables endogènes atteignent certaines valeurs désirées?".

Le but de ce travail est précisément de répondre à la deuxième question, grâce à des techniques d'optimisation.
Nous définissons le critère comme la traduction d'un objectif de stabilisation économique défini a priori comme la trajectoire de certaines variables clés, par exemple, détermination d'un objectif de chômage à 5 ans, interpolation entre ce chiffre et l'observation actuelle et minimisation de l'écart entre la trajectoire ainsi obtenue et les chiffres de chômage qui sortent de la résolution du modèle.

Dans le premier chapitre, le problème de stabilisation économique est posé. Il s'agit d'un problème de contrôle optimal, discret, déterministe, non linéaire, portant sur le modèle SANDY formé de 37 équations.
C'est assez récemment, en 1973, que Pindyck [15] a proposé un algorithme d'optimisation d'un modèle économétrique dont les équations (= contraintes) étaient linéaires et peu nombreuses. Cet algorithme n'est pas opérationnel pour deux raisons :

- les modèles macroéconomiques, en particulier SANDY, sont non linéaires et une linéarisation est souvent impossible vu leur taille (parfois plus de 1000 équations),
- le nombre d'équations et de variables demande un algorithme particulièrement performant pour que les coûts informatiques ne deviennent pas prohibitifs.

Nous avons donc choisi une méthode d'optimisation pour des modèles non linéaires. La technique que nous avons adoptée exige la résolution du modèle et le calcul du gradient du critère.
Les modèles macroéconomiques comportant en général un grand nombre d'équations, il importe de les rendre aussi triangulaires que possible. C'est le but du chapitre 2.
En nous basant sur les travaux de Gabay, Népomiastchy , Rachdi, Ravelli [2], nous décrivons une manière de décomposer le modèle en quatre blocs : prologue, coeur, équations

de bouclage, épilogue. Dès que les équations de bouclage sont résolues, les autres blocs sont récursifs.

L'application de l'algorithme proposé au modèle SANDY fournit seulement quatre équations de bouclage. Dans ce chapitre l'algorithme proposé par Ravelli est détaillé de façon à faciliter son utilisation et son application à SANDY. Nous avons aussi établi des liens entre cet algorithme et les techniques de la théorie des graphes qui étudient la structure d'un système d'équations : condensation et décomposition en niveaux d'un graphe.

Dans le chapitre 3, nous avons opéré un choix parmi les techniques de résolution. Celle qui a été retenue est basée sur la structure quasi-triangulaire du modèle et inspirée de Gabay et al[1]. Le système non linéaire à résoudre par une méthode itérative comporte seulement quatre équations (autant que d'équations de bouclage) pour SANDY.

Par ailleurs, nous décrivons quelques méthodes numériques : Newton et ses variantes. Gauss-Seidel, sécantes, Brown.

En nous basant sur les tests effectués par Gabay et al [1], nous comparons l'efficacité de ces méthodes.

Le chapitre 4 est consacré à l'optimisation du modèle.
En nous référant à Gabay et al [1] nous ramenons le problème initial d'optimisation sans contrainte à un problème non contraint.
Nous détaillons des techniques de minimisation sans contrainte : Steepest descent, Fletcher-Reeves, Quasi-Newton.
Nous calculons le gradient du critère par différences finies. Cette méthode exige la résolution du modèle à chaque itération. Nous fournissons une interprétation du gradient.
Pour conclure ce chapitre, nous suggérons une méthode de pénalisation, en mentionnant une technique, proposée par Gabay et al [1], adaptée à la structure quasi-triangulaire du modèle (pénalisation uniquement des équations de bouclage).

Dans le chapitre 5, nous présentons les résultats des cinq simulations que nous avons effectuées sur SANDY, en privilégiant chaque fois des objectifs différents. Pour l'optimisation de SANDY, nous avons choisi une méthode quasi-newton (BFGS) et pour sa résolution, une variante de la méthode de Newton (Broyden).

Ce travail ne clôture certainement pas l'analyse de l'application de l'optimisation aux modèles macroéconomiques.
Il serait particulièrement intéressant de rechercher des méthodes de calcul du gradient plus efficaces, qui n'exigent pas la résolution du modèle, de tester les différentes méthodes d'optimisation proposées dans les chapitres 3 et 4, pour choisir les plus efficaces et enfin les appliquer à d'autres modèles.

L'intérêt de ces perspectives de recherche nécessiterait un travail approfondi tant au niveau des algorithmes qu'au niveau de la programmation.
Il n'a pas été possible dans le temps qui nous était imparti d'entreprendre cette étude.

CHAPITRE I : FORMULATION DU PROBLEME

1. FORMULATION ECONOMIQUE GENERALE

Confrontés à l'évaluation des mesures de politique économique, les économistes, qui ont pour rôle d'éclairer les choix des acteurs de la vie économique et sociale, utilisent des modèles mathématiques qui traduisent les comportements des agents économiques. Parmi les modèles économiques, les modèles économétriques macroéconomiques ou macro-sectoriels connaissent actuellement un très grand intérêt tant de la part des institutions de planification publiques et privées que de la part des centres d'études et de prévision internationales et nationale.

Formellement, leur modèle macroéconomique est un système d'équations aux différences dont la solution est supposée décrire le comportement des grands agrégats économiques et financiers d'une nation : Produit national brut, consommation privée, investissements privés et publics, exportations, importations, stock monétaire ...
Utilisée dans le secteur public une modélisation détaillée est faite des recettes et dépenses de l'Etat Central, de la Sécurité sociale et des pouvoirs locaux.

Le modèle comporte un certain nombre d'équations dans lesquelles apparaissent deux sortes de variables : les variables exogènes et les variables endogènes.

Utilisée pour prévoir le futur, la solution du modèle doit se baser sur des hypothèses d'évolution des variables exogènes.

1°) les variables exogènes

Ces variables sont classées en deux groupes :

a) les instruments :

Ce sont les grandeurs économiques sur lesquelles les responsables de la politique économique agissent pour atteindre les objectifs qu'ils se fixent. Par exemple : le taux de cotisation de sécurité sociale.

b) les données :

Ce sont les événements extérieurs. Ils ne sont pas définis par les équations du modèle. Les responsables de la politique économique n'ont pas d'action sur eux. Par exemple : le temps, le taux de change du franc belge en dollar ...

2°) les variables endogènes:

Ce sont les grandeurs susceptibles de modifications sous l'influence des variables exogènes. Elles sont définies par les équations du modèle. Par exemple : le produit intérieur brut, les exportations, ...

Dans la terminologie du contrôle optimal, les instruments correspondent aux variables de contrôle ; les variables endogènes, aux variables d'état et les variables exogènes restantes sont considérées comme des données.

Les responsables de la politique peuvent poser deux types de problèmes :

(1) "Que se passe-t-il si ..." on augmente les taux de TVA par exemple. Ou, de façon plus précise ; quelle sera la moins-value fiscale si on bloque pendant six mois l'indexation des salaires et prestations sociales ?

(2) Quelles valeurs optimales faut-il donner aux instruments pour réaliser les objectifs fixés par les responsables ?

C'est en ces termes que se sont posées d'abord les tentatives d'application du contrôle optimal aux problèmes de politique économique. Depuis quelques années, toutefois, les économètres ont renoncé à ce type d'approche parce qu'elle suppose la définition d'une fonction de préférence sur les différents objectifs susceptibles d'être poursuivis. Même en régime totalitaire ceci n'est pas évident. Les conditions économiques elles-mêmes sont venues enlever beaucoup d'intérêt à ce problème : dans la crise longue que traversent les économies occidentales, il ne s'agit plus de répartir les fruits de la croissance mais d'adopter des politiques de stabilisation et d'assainissement des déséquilibres économiques : déficit de la balance des paiements, déficit des finances publiques, chômage. Dans ce contexte, les questions posées au modèle sont du type 1, c'est-à-dire, "What if ..." non plus sur l'exogène mais sur un ou plusieurs objectifs. En d'autres termes : de combien doit-on augmenter les impôts directs pour assainir le déficit public ou de combien faut-il diminuer les revenus réels pour assainir la balance des paiements ou encore de combien faut-il diminuer les cotisations patronales à la sécurité sociale pour créer 50.000 emplois dans le secteur privé.

C'est dans ces termes que le problème sera posé dans ce travail. Des objectifs très schématiques, irréalistes même dans leur ambition, sont imposés à échéance de 5 ans à la balance des paiements, au déficit public et au niveau du chômage.

2. APPLICATION DU MODELE SANDY

Le modèle SANDY est un petit modèle macroéconomique dont on pourrait dire qu'il est une maquette du modèle MARIBEL, utilisé au Bureau du Plan.

Ce modèle réduit comporte 37 équations qui définissent les variables endogènes suivantes :

1) AD : demande agrégée
2) CG : consommation publique en milliards de francs 1970
3) CK : coût du capital
4) CP : consommation privée en milliards de francs 1970
5) DUC : degré d'utilisation de la capacité (1970 = 0.8)
6) GDP : produit intérieur brut en milliards de francs 1970
7) ICX : index des coûts d'exportation (1970 = 1)
8) INVF : investissements privés en milliards de francs 1970
9) KAP : stock de capital du secteur privé
10) LAB : emploi en milliers de personnes
11) LP : emploi total du secteur non gouvernemental en milliers de personnes
12) LSE : travailleurs indépendants en milliers de personnes
13) LW : salariés en milliers de personnes
14) MPORTS : importations de biens et services en milliards de francs 1970
15) OP : Offre potentielle ou Capacité de production
16) PC : prix de la consommation privée (1970 = 1)
17) PG : prix de la consommation publique (fixe et stocks) (1970 = 1)
18) PGDP : prix du produit intérieur brut (1970 = 1)
19) PI : prix des investissements (1970 = 1)
20) PID : index des prix de la demande interne (1970 = 1)
21) PIG : prix des investissements publics (1970 = 1)
22) PM : prix des importations (1970 = 1)
23) PX : prix des exportations (1970 = 1)
24) RR : taux d'intérêt (moyenne annuelle)
25) UN : chômage (milliers de personnes)
26) VARS : variations des stocks
27) WR : salaire moyen en milliers de francs par an et par personne
28) WSE : salaire moyen des indépendants (idem)
29) XPORTS : exportations de biens et services (milliards 1970)
30) YDISP : revenu disponible des ménages en milliards de francs crt.
31) YK : revenu ne provenant pas du travail en milliards de francs crt.
32) YSE : revenu nominal des indépendants (,,)
33) YW : revenu nominal des salariés (,,)
34) M1 : masse monétaire en milliards de francs crt.
35) DEF : déficit de l'état (milliards de francs crt)
36) TAX : taxes (milliards de francs crt)
37) BP : exportations nettes (milliards de francs crt.)

Les variables exogènes considérées comme des données sont :

1) DEP : taux de dépréciation du capital
2) ER : taux de change du FB. en dollars
3) LF : offre de travail
4) LG : emploi total du secteur gouvernemental en milliers de personnes
5) PWX : index des prix mondiaux d'exportations en dollars
6) REX : taux d'escompte de la Banque Nationale
7) TFRM : transferts nets du reste du monde aux ménages en milliards de frs. crt.
8) TFTSR : taux de transferts de l'état aux ménages
9) WT : volume du commerce mondial en dollars
10) TIME : temps

Les instruments sont :

1) IG : investissements publics en milliards de francs 1970
2) X : accroissement du salaire réel
3) COTSSR : taux de cotisation de sécurité sociale sur le revenu professionnel
4) DTAXR : taux de taxation directe sur le revenu personnel
5) TINDR : taux de taxation indirecte sur la consommation privée

Les équations du modèle SANDY sont données en annexe ; il est, en effet, hors de propos de ce travail d'entrer dans l'explication d'un modèle économétrique.

Pour formuler le problème de stabilisation économique évoqué plus haut, il fallait, d'une part, se donner les instruments de politique économique utilisables et d'autre part, déterminer les variables correspondant aux objectifs retenus.

En ce qui concerne les instruments, tous ceux énoncés ci-dessus existaient comme variables exogènes dans la forme originale du modèle, sauf la variable X représentative de la modération salariale, c'est-à-dire, du pouvoir d'achat réel des salariés. Cette formulation du problème imposait dès lors de remplacer l'équation originale de salaire basée sur le comportement de ceux-ci observé dans le passé, par l'identité suivante :

$$\frac{WR_t}{WR_{t-1}} = \frac{PC_t}{PC_{t-1}} \cdot X$$

qui signifie que l'accroissement des salaires est égal à l'accroissement des prix multiplié par l'instrument : hausse des salaires réels.

Nous avons de plus ajouté trois équations qui définissent les variables endogènes BP, TAX, DEF, (M1), c'est-à-dire, les variables qui correspondent aux objectifs.

1°) BP(I) = XPORTS(I) * PX(I) - MPORTS(I) * PM(I)
2°) TAX(I) = COTSSR(I) * (YW(I) + YSE (I)) + TINDR(I) * PC(I) * CP(I) + DTAXR(I) * ((YW(I) + YSE(I) + YK(I))
3°) DEF(I) = CG(I) * PG(I) - TAX(I) + PIG(I) * IG(I) + TFTSR(I) * (YW(I) + YSE (I) + YK(I))

Le problème est posé comme suit :

> Quelles valeurs optimales faut-il donner aux instruments énumérés ci-dessus, à chaque période de 1979 à 1985, pour minimiser l'écart par rapport à une trajectoire désirée donnée, des exportations nettes (BP) (1), du déficit de l'état (DEF), du chômage (UN) ?

Puisqu'une période dure un an, l'indice de temps I varie de 1 à 7.
I = 1 correspond à l'année 1979 tandis que I = 7 signifie qu'il s'agit de l'année 1985.

Notre problème est donc un problème de minimisation sous contraintes.
Le critère a la forme suivante :

$$C = \frac{1}{2} \sum_{i=1}^{T} \left\{ \left(BP(I) - \overline{BP(I)}\right)^2 + \left(UN(I) - \overline{UN(I)}\right)^2 + \left(DEF(I) - \overline{DEF(I)}\right)^2 \right\}$$

Les 37 équations du modèle constituent les contraintes. Elles sont non linéaires. (log, exp, produits, quotients).

Précisons que BP(I) est la valeur de BP à la période I. tandis que $\overline{BP}(I)$, $\overline{UN}(I)$, $\overline{DEF}(I)$ sont les valeurs désirées de BP(I), UN(I), DEF(I).

Données du problème :

a) Calcul des trajectoires désirées

L'objectif poursuivi consiste à obtenir en 1985 :

- les exportations nettes ($\overline{BP}(7)$) égales à zéro
- le déficit de l'état ($\overline{DEF}(7)$) égal à zéro
- le chômage (UN(7) égal à cinquante (exprimé en milliers de personnes)

Pour les périodes intermédiaires, nous avons, par hypothèse de travail, déterminé les valeurs par interpolation linéaire :
Partant des valeurs données de ces variables en 1978 (i=0), nous avons :

- calculé l'écart entre les valeurs en 1978 et en 1985 (i=7)
- retiré le septième de cet écart de la valeur obtenue chaque année.

Ainsi pour le chômage, nous obtenons :

en 1978 : $\overline{UN}(0) = 299.997 = UN(0)$
$\overline{UN}(7) = 50.$
Ecart = 249.997

en 1979 : $\overline{UN}(1) = \overline{UN}(0) - \frac{\text{Ecart}}{7} = 264.283$

en 1980 : $\overline{UN}(2) = \overline{UN}(1) - \frac{\text{Ecart}}{7} = 228.569$

---

(1) Les exportations nettes sont ici considérées comme le solde de la balance des paiements.

Les valeurs des trajectoires désirées sont reprises dans le tableau suivant :

	1979	1980	1981	1982	1983	1984	1985
BP (exportations nettes)	-34.907	-29.089	-23.271	-17.453	-11.636	-5.818	0.
UN (chômage)	264.283	228.569	192.855	157.142	121.428	85.714	50.
DEF (déficit état)	104.431	87.026	69.621	52.215	34.810	17.405	0.

b) Valeurs des variables exogènes (données)

Elles apparaissent avec un retard maximal de deux périodes dans les équations. Il faut donc fournir les valeurs pour les périodes de 1977 à 1985.

	1977	1978	1979	1980	1981	1982	1983	1984	1985
DEP	0.065	0.065	0.065	0.065	0.065	0.065	0.065	0.065	0.065
ER	35.841	31.410	30.782	30.166	29.563	28.972	28.392	27.824	27.268
LF	4057.017	4080.666	4102.100	4117.800	4144.700	4173.597	4200.143	4230.269	4253.620
LG	569.598	594.676	626.518	628.000	628.000	628.000	628.000	628.000	628.000
TWX	2.096	2.385	2.752	3.137	3.451	3.796	4.175	4.593	5.052
REX	0.067	0.061	0.082	0.114	0.125	0.124	0.118	0.114	0.113
TFRM	-6.200	-6.100	-6.661	-7.108	-7.567	-8.074	-8.619	-9.274	-10.059
TFTSR	0.246	0.252	0.252	0.252	2.252	0.252	0.252	0.252	0.252
WT	323.375	341.319	362.481	376.980	388.289	407.704	428.089	449.493	471.968
TIME	77.	78.	79.	80.	81.	82.	83.	84.	85.

c) Autres données

- les valeurs des variables endogènes en 1976 et en 1977 seront nécessaires pour résoudre le modèle
- de même, il faudra disposer des valeurs des instruments en 1978
- pour l'optimisation du modèle, nous aurons besoin d'une approximation initiale des instruments pour les périodes de 1979 à 1985.

CHAPITRE 2 : ETUDE DE LA STRUCTURE DU MODELE

Les modèles macroéconomiques comportent en général un grand nombre d'équations (parfois plus de mille). Il importe d'étudier la structure de ces systèmes, afin de les rendre les plus triangulaires possible et diminuer ainsi le temps de calcul lors des simulations.

Nous exposerons une technique de décomposition d'un système d'équations en blocs constitués d'équations et variables interdépendantes.
Cette méthode est basée sur l'analyse du graphe associé au système d'équations. C'est une application de la condensation d'un graphe.

Nous fournirons ensuite un algorithme de renumérotation automatique des équations, proposé par RAVELLI (voir références[2]). Il permute les équations du modèle, en vue d'obtenir une structure quasi-triangulaire. Nous avons appliqué cette méthode au modèle SANDY.

1. NOTATIONS

a. Généralités

Soit un modèle dynamique en temps discret, pour les périodes k allant de 1 à T.

Appelons n et r, respectivement, le nombre de variables endogènes et d'instruments pour une période.

Notons .y_k^i, la valeur à la période k, de la variable endogène définie par l'équation i

.y_k, le vecteur $\{ y_k^1 \ldots\ldots y_k^n \}$.

Désignons par u_k^i, la valeur de l'intrument i, à la période k,

par u_k, le vecteur $\{u_k^1, \ldots\ldots u_k^r\}$

et par u , le vecteur $\{u_1, \ldots\ldots u_T\}$

Soient p et q, les retards maxima apparaissant dans le modèle, respectivement sur les variables endogènes et sur les instruments.

Si nous considérons que les variables exogènes (données) sont incluses dans les fonctions f_k^i, le modèle est écrit sous la forme générale :

$$(2.1a)\quad k_i\, y_k^i = f_k^i\,(y_k, y_{k-1}, \cdots y_{k-p}, u_k, u_{k-1} \cdots u_{k-q})$$

$$i = 1 \ldots n,\ k = 1 \ldots T$$

$$(2.1b)\quad y_k^i \text{ donné pour } i=1\ldots n;\ k=0,-1,\ldots,\ -p+1$$

$$(2.1c)\quad u_k^i \text{ donné pour } i=1\ldots r;\ k=0,-1\ldots\ -q+1$$

Les k_i sont des paramètres valant :

- .0 si l'équation i n'est pas résoluble analytiquement en y_k^i (k fixé) ou si y_k^i n'intervient pas dans l'équation i.
- .1 sinon

Notons K, la matrice diagonale dont le i-ème élément diagonal est ki :

$$(2.2.)\quad K = \text{diag}\ (ki)$$

En introduisant le vecteur de dimension (nxp) + (rxq),

(2.3) $d_k = \begin{matrix} y_{k-p} \\ \vdots \\ y_{k-1} \\ u_{k-q} \\ \vdots \\ u_k \end{matrix}$

qui contient toute l'information connue au début de la période k, le modèle s'écrit sous la forme condensée :

(2.4) $K\, y_k = f_k\ (y_k,\ d_k)$ avec K, f_k, d_k connus

b. Cas particulier de SANDY

Pour SANDY, puisqu'une période dure un an et que l'étude débute en 1979 et finit en 1985, le nombre T de périodes égale 7.

A chaque période, le modèle comporte 37 équations qui définissent chacune une variable endogène : N vaut 37.
A chaque période, le nombre r d'instruments égale 5.

Chaque équation i étant résolue analytiquement par rapport à la variable endogène i, la matrice K est l'identité (k=1).
Les variables endogènes apparaissent dans SANDY, avec un retard maximal de 2 périodes tandis que sur les instruments, le retard maximal est d'une période.

Suivant la numérotation des équations fournie par le Bureau du Plan, nous pouvons rebaptiser les variables de la manière suivante :

y_k^1 = CG(k)	y_k^{11} = PC (k)	y_k^{20} = YW(k)	y_k^{29} = MPORTS(k)
y_k^2 = PM(k)	y_k^{12} = PG (k)	y_k^{21} = WSE(k)	y_k^{30} = GDP(k)
y_k^3 = OP(k)	y_k^{13} = PI(k)	y_k^{22} = YSE(k)	y_k^{31} = PGDP(k)
y_k^4 = DUC(k)	y_k^{14} = PIG(k)	y_k^{23} = YK (k)	y_k^{32} = RR(k)
y_k^5 = INVF(k)	y_k^{15} = ICX(k)	y_k^{24} = YDISP(k)	y_k^{33} = CK(k)
y_k^6 = KAP (k)	y_k^{16} = PX(k)	y_k^{25} = VARS(k)	y_k^{34} = BP(k)
y_k^7 = LP(k)	y_k^{17} = XPORTS(k)	y_k^{26} = CP(k)	y_k^{35} = DEF(k)
y_k^8 = LAB (k)	y_k^{18} = LW(k)	y_k^{27} = AD(k)	y_k^{36} = TAX(k)
y_k^9 = UN(k)	y_k^{19} = LSE(k)	y_k^{28} = PID(k)	y_k^{37} = M1(k)
y_k^{10} = WR (k)			

$$u_k^1 = IG(k)$$

$$u_k^2 = X(k)$$

$$u_k^3 = COTSSR\ (k)$$

$$u_k^4 = DTAXR(k)$$

$$u_k^5 = TINDR(k)$$

Le modèle SANDY peut être traduit sous la forme générale :

$$(2.1a)' \quad y_k^i = f_k^i\ (y_k, y_{k-1}, y_{k-2}, u_k, u_{k-1}) \qquad i = 1 \ldots 37 \quad k = 1 \ldots 7$$

$$(2.1b)' \quad y_k^i \text{ donné pour } i = 1 \ldots 37 \quad k = 0, -1 \text{ (années 1978,1977)}$$

$$(2.1c)' \quad u_k^i \text{ donné pour } i = 1 \ldots .5 \quad k = 0 \text{ (année 1978)}$$

2. DECOMPOSITION D'UN SYSTEME D'EQUATIONS EN BLOCS D'INTERPDEPENDANCE

a. Rappel de la théorie des graphes

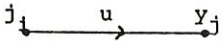
Un graphe est un couple G (Y,U) constitué par :

- un ensemble fini $Y = \{ y_1, y_2 \ldots . y_n\}$ dont les éléments sont appelés sommets;
- une famille $U = (u_1, u_2 \ldots . u_m)$ d'éléments du produit cartésien

$$Y \times Y = \{ (x,y) : x \in Y, y \in Y\} \text{ appelés arcs}$$

Nous pouvons représenter un graphe par un schéma :
par exemple:

$$Y = \{ y_i, y_j\} \ ; \ U = \{ u = (y_i, y_j)\}$$



. L'arc u est représenté par une flèche orientée de y_i à y_j.

y_i est l'extrémité initiale de l'arc u

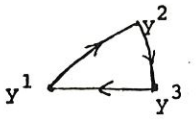
y_j est l'extrémité terminale de l'arc u

Un chemin $\mu = (u_1, u_2 \ldots u_i, u_{i+1} \ldots u_q)$ est une suite d'arcs $(u_1 \ldots u_q)$ telle que l'extrémité terminale de u_i coïncide avec l'extrémité initiale de $\underbrace{u_i+1}_{u_{i+1}}$; q est la longueur du chemin.

Un circuit σ est un chemin $\sigma = (u_1 \ldots u_q) = [y_1, y_2 \ldots y_r]$ telle que l'extrémité terminale du dernier arc u_q coïncide avec l'extrémité initiale du premier arc u_1.

Exemple $Y = \{ y_1, y_2, y_3 \}$

$U = \{u_1 = (y_1, y_2) ; u_2 = (y_2, y_3); u_3 = (y_3, y_1)\}$



$\mu = (u_1, u_2)$ est un chemin de longueur 2.

$\sigma = (u_1, u_2, u_3) = [y_1, y_2, y_3, y_1]$ est un circuit.

Nous définirons encore :

* un <u>prédécesseur</u>

  Le sommet y_i est un prédécesseur du sommet y_j s'il existe un chemin reliant y_i à y_j.

  Dans l'exemple ci-dessus, y_2 est un prédécesseur de y_3 et y_1.

* un <u>sucesseur</u>

  Le sommet y_i est un successeur de y_j s'il existe un chemin reliant y_j à y_i.

  Dans l'ensemble y_2 est un successeur de y_1 et y_3.

b. <u>Décomposition d'un système d'équations</u>

Considérons un système de n équations à n inconnues y_i appartenant à R.

Soit $g_j (y_1, y_2, \ldots y_n) = 0 \quad j = 1,\ldots n$

où

$g_j : R^n \longrightarrow R$

Nous cherchons à subdiviser le système initial en k blocs de dimensions inférieures à n, résolubles les uns après les autres.

Chaque bloc est constitué d'un groupe d'équations qui doivent être résolues simultanément.

La résolution du système de dimension n revient alors à la résolution successive de k sous-systèmes de dimensions inférieures à n (opération en général plus simple).

Pour cela, nous imposons les hypothèses suivantes :

1) Chaque équation g_j admet une <u>variable de sortie</u> y_i, c'est-à-dire une variable dont la valeur peut être déterminée par g_j moyennant la connaissance des valeurs des autres variables intervenant dans l'équation j. Ces dernières sont appelées <u>variables d'entrée</u>.

2) Il existe une bijection entre les équations et les variables de sortie. Cela signifie que chaque variable de sortie est déterminée par une et une seule équation et réciproquement chaque équation fournit une et une seule variable de sortie.

L'ensemble des n couples $\{(g_{j1}, y_{i1}), \ldots.(g_{jn}, y_{jn})\}$ où y_{i1} est la variable de sortie de g_{j1} est appelé <u>ensemble de paires de sorties</u>.

Nous pouvons encore supposer qu'après une éventuelle renumérotation des équations, les paires de sortie sont $\{(g_1, y_1)\ldots. (g_n, y_n)\}$.

Ces hypothèses sont évidemment vérifiées dans SANDY : la matrice K, définie par (2), étant l'identité, le modèle s'écrit $y_k^i = f_k^i (y_k, d_k)$, i=1...n.
Pour tout i, y_k^i est la variable de sortie de l'équation $g_k^i (y_k) = y_k^i - f_k^i (y_k, d_k) = 0$.

1° Graphe associé à un système d'équations

A chaque système de n équations à n inconnues, nous pouvons associer un graphe de la manière suivante :

- L'ensemble Y = {1,... n} des sommets représente l'ensemble des paires de sortie $\{(y_i, g_i)\}$.
- L'ensemble U des arcs est obtenu en associant 2 sommets i et j par un arc orienté de i à j si la variable y_i est une variable d'entrée de l'équation j. Le graphe ainsi obtenu est sans boucle.

L'existence d'une boucle (i,i) signifierait celle de l'arc (i,i), c'est-à-dire que y_i, variable de sortie de l'équation i, serait en même temps variable d'entrée de cette équation.

Considérons une équation dans laquelle y_i est à la fois variable d'entrée et variable de sortie, comme par exemple dans SANDY.

```
LSE(I) = LSE(I) * (LOGIC (TIME(I). LE. 1978DO))
       +  LOGIC(TIME(I). GT. 1978D0) * (LSE(I-1) - 1.500).
```

Nous pourrions traduire cette particularité en imaginant deux modèles :

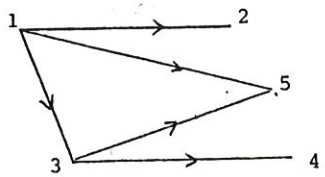
- L'un,utilisé après 1978, contiendrait l'équation LSE(I) = LSE(I-1) - 1.5.
  y_i = LSE(I) n'est plus variable d'entrée et l'arc (i,i) n'existe plus.
- L'autre, dans lequel nous aurions supprimé l'identité LSE(I) = LSE(I) des équations serait utilisé avant 1978.
  Ceci revient à enlever l'arc (i,i) du graphe associé qui est alors sans boucle.

Nous considérons que l'arc (i,i) n'existe pas et que le graphe est sans boucle.

A titre d'exemple nous allons reproduire le graphe associé au système (2.5) :

Equations		Variables de sortie
(2.5)	$g_1(y_1) = 0$	y_1
	$g_2(y_2, y_1) = 0$	y_2
	$g_3(y_1, y_3) = 0$	y_3
	$g_4(y_3, y_4) = 0$	y_4
	$g_5(y_1, y_3, y_5) = 0$	y_5

Le schéma suivant représente le graphe en question:



2° Interprétation du graphe associé

L'existence d'un chemin allant de i à j indique qu'il faut connaître la valeur de y_i pour pouvoir résoudre l'équation j.

Dans le système (2.5), il faut connaître y_1 pour résoudre l'équation 2 :

$g_2(y_2, y_1) = 0$.

Si un sommet j n'a pas de prédécesseur, il n'existe pas de chemin aboutissant en j.

Cela signifie que l'équation j ne nécessite pas la connaissance des valeurs d'autres variables pour sa résolution. Nous commencerons donc toujours par résoudre les équations correspondant aux sommets qui n'ont pas de prédécesseur.

L'existence d'un circuit passant par i et j (c'est-à-dire d'un chemin de i à j et d'un chemin de j à i) signifie que les équations i et j doivent être résolues simultanément.

Si le graphe est sans circuit, aucun couple d'équations ne doivent être résolues simultanément. Nous allons étudier au paragraphe 3°) ci-dessous, une manière d'effectuer la résolution du système, dans ce cas.

3°) Décomposition en niveaux d'un graphe sans circuit

Définitions

Le niveau de rang zéro, N_0, est l'ensemble des sommets sans précédent (i est un précédent de j s'il existe un chemin de longueur 1, de i à j). Nous retirons du graphe ces sommets et les arcs les ayant pour extrémités.

Nous considérons dans l'ensemble restant les sommets sans précédent : ils forment le niveau de rang 1, N1.

Nous continuons ainsi jusqu'à ce qu'il ne reste plus de sommets. On peut démontrer que c'est possible si le graphe est sans circuit.

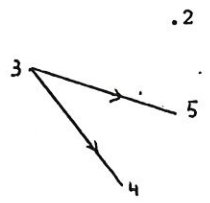
Nous renvoyons à CALLIER [5] pour la description détaillée de l'algorithme de décomposition en niveaux d'un graphe sans circuit et pour la démonstration des propriétés suivantes :

- un graphe est sans circuit si et seulement si il est décomposable en niveaux,
- si le sommet y fait partie du niveau i, ses successeurs sont dans un niveau de rang supérieur et ses prédécesseurs dans un niveau de rang inférieur. Ceci justifie la résolution d'un système récursif dans l'ordre croissant du rang des niveaux.

Exemple : décomposition en niveau du graphe associé au système (2.5)

*) No = { 1 } : sommets sans précédent

*) Le graphe réduit est représenté par :



N1 = {2, 3} : sommets sans précédent dans le graphe réduit.

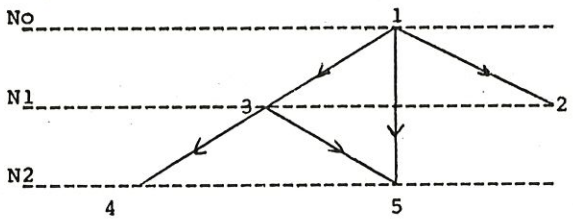
*) Le graphe réduit est

.5

.4

N2 = {4, 5}

Le schéma suivant représente la décomposition en niveaux du graphe associé au système (2.5)



La résolution du système (2.5) s'effectue dans l'ordre croissant du rang des niveaux c'est-à-dire :

1° Résoudre $g_1\ (y_1) = 0$. Cela fournit y_1

2° y_1 étant donné, résoudre les équations 2 et 3.

$g_2\ (y_2, y_1) = 0$. Nous obtenons y_2

$g_3\ (y_3, y_1) = 0$. Nous obtenons y_3

3° Résoudre les équations 4 et 5.

$g_4\ (y_3, y_4) = 0$. Nous obtenons y_4

$g_5\ (y_1, y_3, y_5) = 0$. Nous obtenons y_5.

Pour être complet, il nous reste à envisager le cas d'un graphe associé contenant des circuits.

4°) Décomposition en blocs d'un système dont le graphe associé contient des circuits

Dans ce cas, deux équations au moins doivent être résolues simultanément.

Nous groupons donc dans un même bloc toutes les équations qui doivent être résolues ensemble.

Il est clair qu'il existe une façon d'ordonner les blocs de sorte qu'ils puissent être résolus l'un après l'autre.

Nous sommes confrontés au double problème de :

1° déterminer les blocs,

2° définir l'ordre de résolution des blocs.

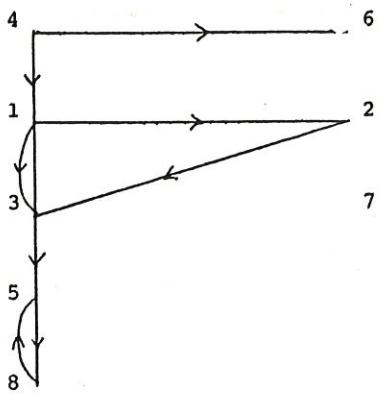
Nous allons définir une procédure de décomposition. Pour les justifications théoriques, nous renvoyons à CALLIER[5].

Exemple

Considérons le système suivant :

$$(2.6)\quad \begin{cases} g_1\ (y_1, y_3, y_4) = 0 \\ g_2\ (y_1, y_2, y_6) = 0 \\ g_3\ (y_1, y_2, y_3) = 0 \\ g_4\ (y_4, y_6) = 0 \\ g_5\ (y_3, y_5, y_8) = 0 \\ g_6\ (y_4, y_6) = 0 \\ g_7\ (y_2, y_7) = 0 \\ g_8\ (y_5, y_8) = 0 \end{cases}$$

Le graphe associé est représenté par :



Nous définissons une composante fortement connexe d'un graphe G(Y,U) comme un ensemble formé par tous les sommets liés entre eux par des circuits".

Les composantes fortement connexes C_i du graphe associé au système (2.6) sont :

$$\begin{cases} C_1 = \{1, 2, 3\} \\ C_2 \quad \{4, 6\} \\ C_3 \quad \{5, 8\} \\ C_4 \quad \{7\} \end{cases}$$

De manière plus précise, la composante fortement connexe du sommet 1 est formée de 1 et de tous les sommets qui forment un circuit avec 1.

Supposons que Y possède k composantes fortement connexes C_i, i = 1 .... k

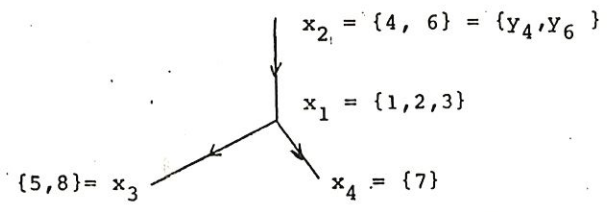
Elles forment une partition de l'ensemble Y des sommets. Cela signifie qu'elles sont non vides, disjointes deux à deux et que leur réunion donne Y.

Nous formons alors le graphe $G_t = (X,V)$ appelé condensation du graphe G=(Y,U) de la manière suivante :

- X est constitué de points x_i représentant la composante fortement connexe C_i de Y.
- V est constitué d'arcs $v = (x_i, x_j)$.
  L'arc v existe s'il existe un arc $u=(y_i, y_j)$ (noté aussi (i,j)) dans le graphe initial avec y_i dans C_i et y_j dans C_j.

Autrement dit, on fond tous les arcs liant les sommets de C_i aux sommets de C_j en un seul arc unissant le point x_i, représentatif de C_i au point x_j, représentatif de C_j.

La condensation G_c du graphe G associé au système (2.6) est



La bijection entre les variables de sortie et les équations permet de décomposer le système en blocs. Il suffit de transposer le graphe G_c aux équations :

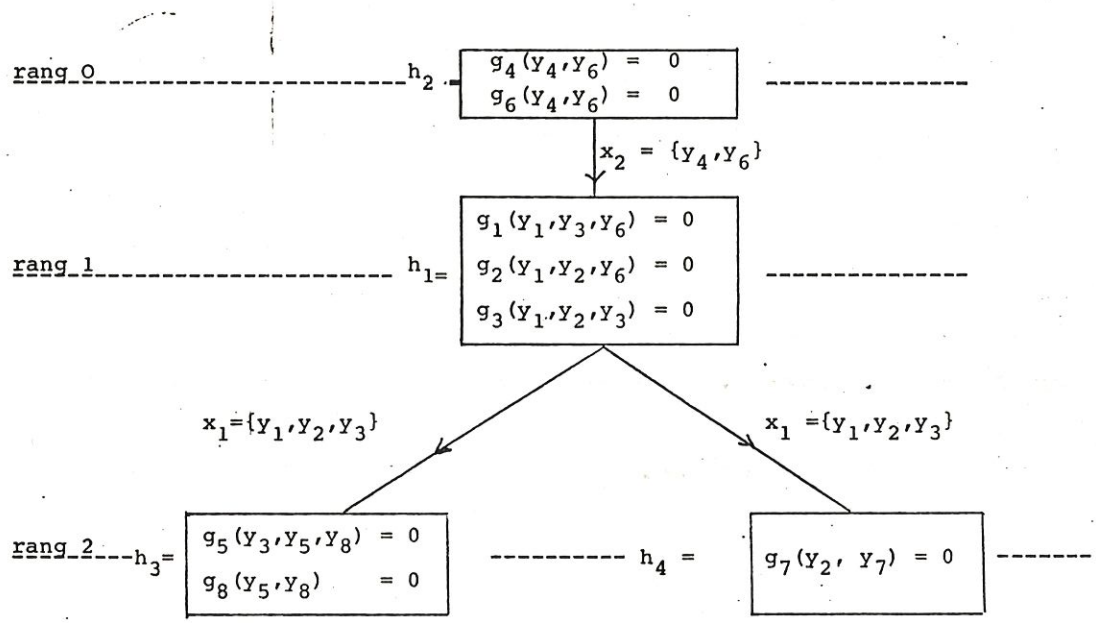
$$\begin{cases} h_1 = \{g_1, g_2, g_3\} \\ h_2 = \{g_4, g_6\} \\ h_3 = \{g_5, g_8\} \\ h_4 = \{g_7\} \end{cases}$$

A ce stade, nous avons déterminé les blocs d'équations à résoudre simultanément.

Il reste à établir une hiérarchie dans l'ordre de ces blocs afin de permettre leur résolution successive.

Pour cela, il suffit de décomposer en niveaux le graphe sans circuit qu'est la condensation G_c et de résoudre les blocs dans l'ordre croissant du rang des niveaux de G_c.

La résolution du système (2.6) s'effectue dans l'ordre suivant :



La solution de h_2 fournit $x_2 = \{y_4, y_6\}$ qui est transmis à h_1. La solution de h_1 fournit $x_1 = \{y_1, y_2, y_3\}$ qui est transmis à h_3 et h_4. La solution de h_3 fournit $x_3 = \{y_5, y_8\}$; celle de h_4 fournit $x_4 = \{y_7\}$.

Résumé :

Pour obtenir la solution d'un système d'équations de dimension n, par substitution successive, en considérant des blocs d'équations de taille minimale, il faut :

1° établir le graphe associé G = (Y,U)

2° établir la consensation $G_c = (X,V)$ de G = (Y,U)

3° établir les niveaux dans G_c.

Lorsque la taille du modèle augmente, la décomposition, basée sur l'étude du graphe associé est compliquée : le graphe devient rapidement indéchiffrable.

Par contre, une analyse du couplage des équations s'appuyant sur la matrice d'incidence est réalisable.

5°) Notion de matrice d'incidence

La matrice d'incidence du modèle (2.4), est la matrice E, de dimension n x n, d'éléments :

$i \neq j : e_{ij} = 1$ si la variable y_k^j est une variable d'entrée de l'équation i (à la période k fixée) c'est-à-dire si la dérivée

$$\frac{\partial f_k^i (y_k, d_k}{\partial y_k^i}$$ n'est pas identiquement nulle.

$= 0$, sinon

$i = j : e_{ii} = 1 - k_i \quad i = 1 \ldots n$ (eii = 0 pour SANDY)

Propriétés

1) Puisque la matrice de JACOBI F_k (période k fixée) du système (2.1a) (ou (2.4)) est formée des éléments

$F_{ij} = (\frac{\partial f_k^i}{\partial y_k^j}$ i, j = 1... n, il est évident que les éléments hors diagonaux non identiquement nuls de E (matrice d'incidence) et F_k sont situés aux mêmes endroits.

2) Le calcul automatique de E est une opération compliquée, surtout en Fortran. Il comporte deux parties :

1° Remplissage adéquat de la diagonale. Rappelons que si la variable y_i apparait à la fois dans le membre de gauche et dans celui de droite de l'équation, il faut mettre un zéro à la place (i,i) de la matrice E. (ce qui équivaut à supprimer l'arc (i,i) du graphe associé).

2° Calcul des éléments hors diagonaux par analyse syntaxique.

Ceci consiste en une analyse du texte de l'équation en vue de reconnaître les variables endogènes y figurant et le numéro, l, porté par l'équation. Il faut aussi découvrir le numéro j, de la variable endogène repérée dans l'équation l et mettre alors un 1 à la position (l, j) de la matrice E. Par exemple, pour le modèle SANDY, les équations sont données par la routine MODEL (I). Une variable endogène précède toujours les caractères (I). Lorsque nous avons repéré (I), il faut identifier les caractères qui précèdent, jusqu'à ce que l'obtention de l'un des signes suivants (, = , *, +, -, **. A ce moment, il faut vérifier si le"mot" obtenu est une variable endogène. (Cela pourrait être un instrument ou une variable exogène). Il faut ensuite trouver ses coordonnées (l,j) pour mettre 1 à la position (l,j) de la matrice E.

3) Par définition de graphe associé et de matrice d'incidence, l'élément $e_{ij} = 1$ $i \neq j$ signifie que la variable j est une variable d'entrée de l'équation i, ou encore qu'il existe un arc reliant le somment j au sommet i, ou encore que le sommet j est un prédécesseur du sommet i ou encore i est un successeur de j.

4) L'étude de cette matrice permet de déterminer :

- Les sommets qui n'ont pas de prédécesseurs dans le graphe associé. Ce sont les sommets i tels que la ligne i de la matrice d'incidence est nulle.
- Les sommets qui n'ont pas de successeurs. La colonne correspondante de E est nulle.
- Les circuits. Les sommets i et j appartiennent à un circuit si i est à la fois successeur et prédécesseur de j.

Exemple (proposé dans la référence [2])

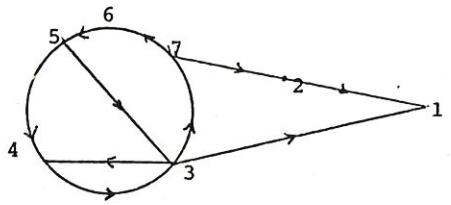
Considérons le système suivant :

1. S = 0.45 * A * (1+L) - 1 1.9
2. A = D + 26.69
3. L = 0.22 * P + 0.395 * H + 0.087
4. P = 0.25 * (L-M) + 0.04
5. M = (37.25/B) - 1
6. B = D + 36.02
7. D = 0.297 - 0.1036 * L

où

S = variation de salaire disponible
A = Emploi salarié de l'industrie
L = Taux de salaire horaire
P = Prix à la consommation
M = Productivité de la main d'oeuvre dans l'industrie.
B = Emploi dans l'industrie
D = Variation de l'emploi dans l'industrie.

Le graphe associé est représenté par la figure suivante :



La matrice d'incidence du modèle (2.7) est :

	1	2	3	4	5	6	7
1		1	1				
2							1
3				1	1		
4			1		1		
5						1	
6							1
7			1				

Figure 1

Les cases blanches représentent des zéros
Le sommet 1 n'a pas de successeur car la première colonne est nulle.
Les variables 3 et 4 appartiennent à un circuit puisque e_{34} et $e_{43} = 1$.
Les variables 7 et 3 appartiennent à un circuit car
. 7 est un successeur de 3 puisque $e_{73} = 1$
. 7 est un prédécesseur de 3 puisque $e_{34}=1=e_{45}=e_{56}=e_{67}$

c. Décomposition en blocs du modèle SANDY

Partant de la matrice d'incidence obtenue à partir de la numérotation initiale des équations de SANDY, nous obtenons une décomposition de notre modèle en 11 blocs : 5 blocs à une équation, 1 bloc de 27 équations, 5 blocs à une équation.

La résolution des cinq premiers blocs, pour la période k, fournit les valeurs y_k^1, y_k^2, y_k^{16}, y_k^{17}, y_k^{19} de la numérotation initiale (respectivement CG(k), PM(k), PX(k), XPORT(k), LSE(k) qui seront des données pour les blocs suivants.
La réunion de ces cinq blocs fournit un groupe d'équations que nous appellerons plus loin prologue.

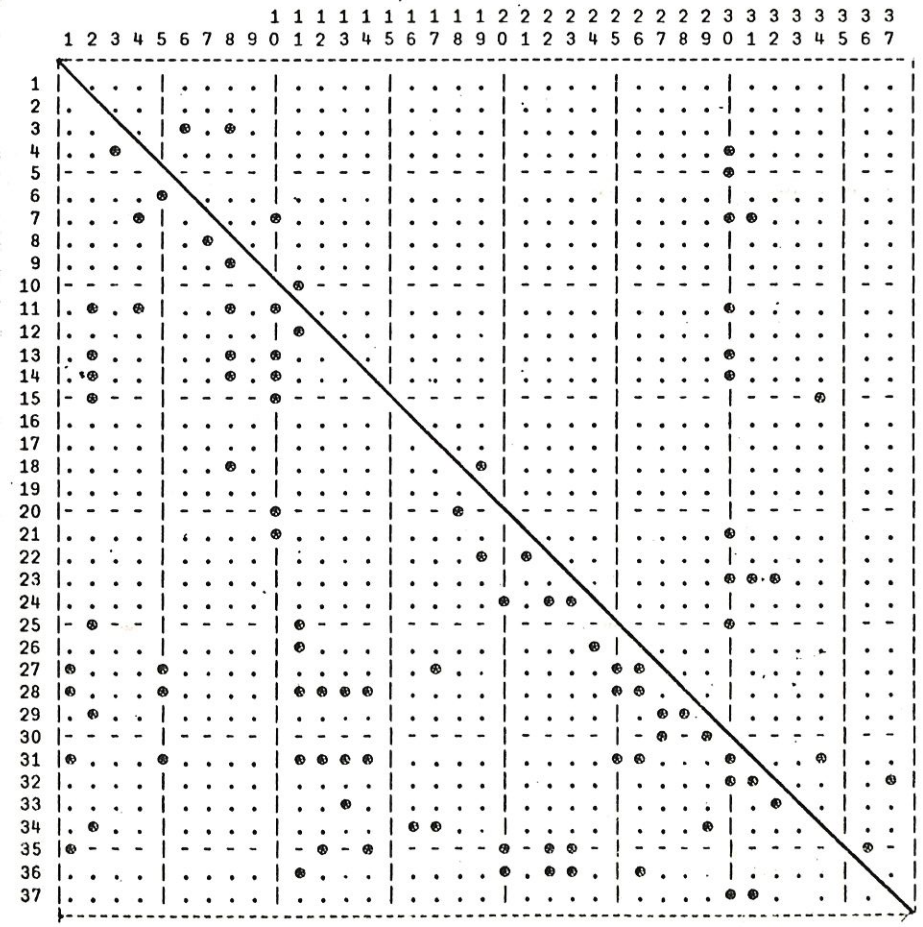
Nous avons ensuite un bloc géant de 27 équations qui doivent être résolues simultanément.

Après résolution, les variables endogènes définies par ces équations deviennent des données pour les blocs suivants.

Puis nous pouvons résoudre de manière récursive le dernier groupe, formé par cinq blocs à une équation. Nous appellerons ce groupe épilogue (voir plus loin). Il comporte les équations 36, 33, 35, 15, 9 définissant respectivement : TAX(k), CK(k), DEF(k), ICX(k), UN(k) .

Ces différentes remarques sont illustrées par les figures 2 et 3 représentant respectivement les matrices d'incidence du modèle SANDY, avec la numérotation initiale, et avec la numérotation résultant de la décomposition en blocs.

Matrice d'incidence initiale de SANDY

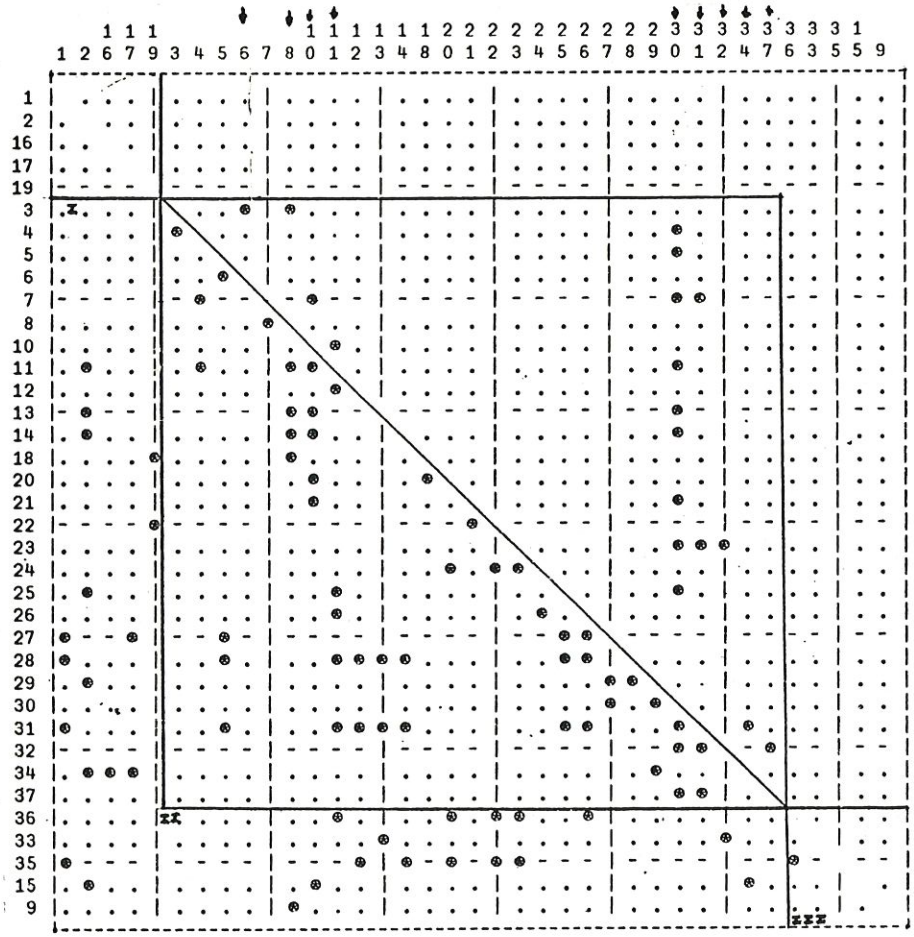


- figure 2 -

● représente 1

Les autres éléments sont nuls

Matrice d'incidence obtenue après la décomposition en blocs



- figure 3 -

Les numéros indiquent l'ordre de départ

⊛ représente 1

Les autres éléments sont nuls.

Nous avons ramené la résolution de notre système de 37 équations non linéaires à la résolution d'un système de 27 équations non linéaires et de deux sous-systèmes récursifs de 5 équations chacun.

En examinant de plus près le groupe II de 27 équations, nous constatons qu'en donnant une valeur aux variables fléchées (y_k^6, y_k^8, y_k^{10}, y_k^{11}, y_k^{30}, y_k^{31}, y_k^{32}, y_k^{34}, y_k^{37} de la numérotation initiale), nous rendons le reste du système récursif.

Ceci permet de décomposer le bloc II en deux sous-systèmes :

- l'un récursif comporte 18 équations. Nous l'appellerons coeur plus loin.
- l'autre non récursif contient 9 équations. Il correspond aux équations de bouclage. (voir plus loin).

Grâce à des permutations, nous pourrons encore ramener ce sous-système de dimension 9, à un groupe de seulement quatre équations et rendre le modèle initial quasi-triangulaire. Ce sera l'objet du paragraphe suivant.

3. STRUCTURE QUASI-TRIANGULAIRE D'UN MODELE MACROECONOMIQUE

a. Notion de variable de bouclage

Quelle définition intuitive pouvons-nous donner d'une variable de bouclage d'un système d'équations?

C'est une variable telle que si nous lui attribuons une valeur, le reste du système devient récursif.
Autrement dit, considérant la variable de bouclage comme une donnée, nous pouvons déduire la valeur des autres variables, par simple application sucessive des équations.

Un système peut être résolu entièrement de manière récursive si le graphe associé est sans circuit. Les variables de bouclage peuvent être interprétées comme des singularités dans cette récursivité..

L'existence d'une variable de bouclage signifie la présence d'un circuit (ou boucle) dans le graphe associé. Donner une valeur à la variable de bouclage peut être interprété comme casser le circuit.

Ainsi, si pour un certain i, ki vaut zéro (ce n'est pas le cas pour SANDY), cela signifie que la variable i, ne peut être une variable de sortie de l'équation i. C'est donc une variable d'entrée pour cette équation.

Dès lors, l'élément e_{ii} de la matrice d'incidence vaut un et le graphe associé contient une boucle (= arc(i,i)).

Les variables i telles que ki=0, ce qui est équivalent à $e_{ii}=1$, seront automatiquement des variables de bouclage.

La définition formelle est liée à la matrice d'incidence.

La variable j est une variable de bouclage du système (2.1a) si et seulement le produit

$$\prod_{i=1}^{j} (1 - e_{ij}) = 0$$

Ceci équivaut à $k_j \prod_{i=1}^{j-1} (1-e_{ij}) = 0$ puisque $e_{jj} = 1 - k_j$

Cela revient à dire que :

- soit k_j est nul
- soit la j^e colonne de la matrice d'incidence de (1a) possède au moins un élément non nul au-dessus de la diagonale principale.

Une équation de bouclage porte le même numéro qu'une variable de bouclage.

Un système est <u>récursif</u> s'il ne contient pas de variable de bouclage, c'est-à-dire si sa matrice d'incidence est triangulaire inférieure et si pour tout i, l'équation i est résolue analytiquement par rapport à y_k^i (k fixé).

Ceci est équivalent à :

$$S \text{ récursif si et seulement si } \begin{cases} - e_{ij} = 0 & \text{pour tout } i \\ - e_{ij} = 0 & \text{pour tout } j > i \\ - k_i = 1 & \text{pour tout } i \end{cases}$$

Nous pouvons écrire un tel système :

$$(2.(1a)") \quad y_k^i = f_k^i \ (y_k^1, \ \ldots \ y_k^{i-1}, \ d_k) \qquad i = 1 \ \ldots \ n \quad (k \text{ fixé})$$

Le théorème suivant justifie la définition intuitive de variable de bouclage, introduite au début de ce paragraphe.
Il s'avèrera fondamental au chapitre 3.

<u>Théorème</u>

<u>Si</u> : . $\bar{B}$ est l'ensemble des indices des variables qui ne sont pas variables de bouclage

. $S(\bar{B})$ = sous-système obtenu en ne conservant que les équations i, pour i dans $\bar{B}$ et en considérant les variables y^j pour j n'appartenant pas à $\bar{B}$ comme des données (nous avonsainsi l'indice k de la période).

<u>Alors</u>: $S(\bar{B})$ est un <u>système récursif</u>

<u>Démonstration</u>

1°) $k_i = 1$ pour tout i dans $\bar{B}$ sinon l'équation i serait équation de bouclage et i ne serait pas dans $\bar{B}$

2°) $e_{ii} = 0$ pour tout i dans $\bar{B}$. Sinon ki serait nul et il y aurait une contradiction avec 1°.

3°) $e_{ij} = 0$ pour tout j > i (j dans $\bar{B}$ s'il existait j > i tel que $e_{ij} = 1$, le produit $k_j \prod_{i=1}^{j-1} (1-1_{ij})$ serait nul et la variable j serait une variable de bouclage.
Ceci est en contradiction avec "j est dans $\bar{B}$".

c.q.f.d.

<u>Propriétés</u>

Il convient de remarquer que le nombre de variables de bouclage dépend de l'ordre des équations.
Une matrice d'incidence se réfère à une numérotation.

Ainsi l'examen de la matrice d'incidence associée au modèle (7) (figure 1) fournit

5 variables de bouclage : y^2, y^3, y^4, y^5, y^7.

L'écriture des équations de ce modèle dans l'ordre 6 - 5 - 3 - 7 - 4 - 2 - 1, ce qui correspond aux variables (B, M, L, D, P, A, S) fournit la matrice d'incidence, reproduite à la figure 4.

		6	5	3	7	4	2	1	← numérotation initiale
		1	2	3	4	5	6	7	← nouvelle numérotation
6	1				1				
5	2	1							
3	3		1			1			
7	4			1					
4	5		1	1					
2	6				1				
1	7			1			1		

- figure 4 -

Le nombre de variable de bouclage s, initialement égal à cinq, se réduit à 2 : les variables y_4 et y_7 de l'ancien numérotage.

b. Différentes parties d'un modèle

La définition de variable de bouclage permet de découper le modèle en quatre blocs successifs, après des permutations des équations. Il s'agit du prologue, du coeur, des équations de bouclage, de l'épilogue.

1° Les l premières équations du modèle ne contiennent pas de variables de bouclage. Elles forment un bloc récursif, appelé prologue

$$(2.8)\ \text{Prologue} : \begin{cases} y_k^1 = f_k^1\ (dk) \\ \vdots \\ y_k^i = f_k^i\ (y_k^1, \ldots y_k^{i-1}, d_k) \\ \vdots \\ y_k^l = f_k^l\ (y_k^1, \ldots y_k^{l-1}, d_k) \end{cases}$$

Un exemple tout simple d'équations du prologue est donné dans SANDY par :

$$LSE_k = LSE_{k-1} - 1.5$$

2° Après le prologue, deux groupes d'équations apparaissent dans le modèle :

a) le coeur constitué par les (m-l) équations de définition des variables endogènes $y_k^{l+1} \ldots y_k^m$

b) les équations de bouclage qui définissent les variables de bouclage $y_k^{m+1} \ldots y_k^{m+s}$

Désignons par YB_k, le vecteur de dimension s, formé par $YB_k = \{y_k^{m+1} \ldots y_k^{m+1}\}$

En vertu de la définition des variables de bouclage, le coeur devient récursif dès que nous attribuons une valeur à YB_k.

Il peut s'écrire :

$$(2.9)\quad \underline{\text{Coeur}} \qquad y_k^i = f_k^i\ (y_k^1,\ y_k^2 \ \ldots\ y_k^{i-1},\ YB_k,\ dk)$$

$$i = 1 + 1 \ldots\ m$$

3°) Les équations de bouclage ne constituent pas un système récursif. De plus, elles ne sont pas nécessairement résolubles analytiquement par rapport à la variable correspondante. Pour cette raison, nous maintenons le coefficient k_i dans l'écriture:

$$(2.10)\ \underline{\text{Equations de bouclage}} : ki\,y_k^i = f_k^i\ (y_k^1 \ \ldots\ y_k^{m+s},\ d_k)$$

$$i = m + 1, \ldots\ m + s.$$

Nous savons que dans SANDY, ki = 1 pour tout i.

4°) Enfin, le modèle contient un dernier bloc, l'épilogue. Il est récursif. Les variables se déduisent de celles qui précèdent par simple application des formules. Il s'écrit :

$$(2.11)\ \underline{\text{Epilogue}} :\ y_k^i = f_k^i\ (y_k^1 \ \ldots\ y_k^{i-1},\ d_k)$$

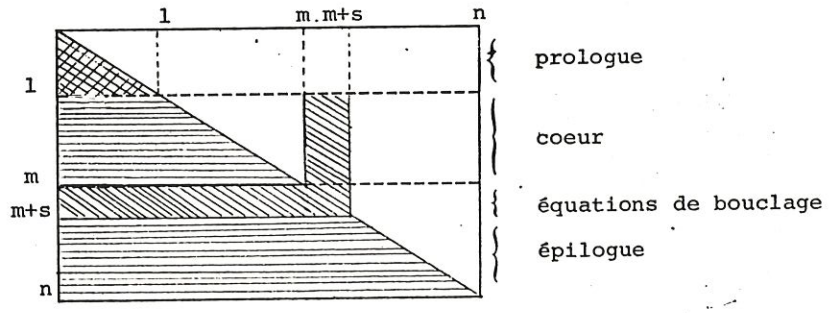
$$i = m + s + 1, \ \ldots\ n.$$

Par exemple, dans SANDY, l'équation définissant le chômage, $UN_k = LF_k - LAB_k$, fait partie de l'épilogue.

L'offre de travail LF_k est une variable exogène donnée tandis que l'emploi LAB_k est déterminé précédemment par les équations de bouclage.

Les coefficients ki des équations du prologue, du coeur, de l'épilogue valent toujours un quel que soit le modèle. Si ki était nul, nous aurions immédiatement une variable de bouclage. Pour cette raison, nous l'avons omis en écrivant ces équations.

La mise du modèle sous forme de quatre blocs se traduit sur la matrice d'incidence qui prend la forme suivante .



- figure 5 -

Le prologue et l'épilogue peuvent être vides. Nous avons alors 1 = 0 et m+s = n.

La partie non hachurée de la matrice d'incidence est entièrement constituée de zéros.

Le tableau suivant montre que la partie hachurée est en général fort creuse pour les modèles macroéconomiques.

Nom du modèle	STAR	FAIR	MPS	SANDY
Dimension de la matrice E	139	83	400	37
Nombre d'éléments non nuls de $(\frac{\partial f_k}{\partial y_i}$	173	218	865	100
% d'éléments non nuls	0,9%	3,16%	0,54%	7,3%

E = matrice d'incidence

SOURCES : Ces chiffres sont tirés de [1] pour les modèles STAR, FAIR, DMS. Pour STAR, nous avons ajoutés les 48 équations du prologue, non comptées par GABAY et al[2]

Pour MPS, ils sont fournis par GABAY et al[2]

Pour SANDY, ils proviennent du Bureau du Plan.

La mise du modèle sous forme quasi-triangulaire est réalisable sur n'importe quel modèle.
Nous proposons un algorithme pour exécuter cette décomposition au paragraphe suivant.

c. Mise du modèle sous forme quasi-triangulaire

Cette décomposition est surtout intéressante lorsque s est petit par rapport à n.
(s = nombre de variables de bouclage; n = nombre d'équations).
Cette propriété est vérifiée pour les modèles macroéconomiques.

Modèle	Nombre d'équations	Nombre de variables de bouclage	s/n %
ANDOMINI	4	1	25 %
PIMPON	14	2	14 %
STAR	139	3	2 %
FAIR	83	6	7 %
MPS	400	20	5 %
DMS	plus de 1000	moins de 100	- de 10%
SANDY	37	4	10 %

SOURCES : - Gabay et al [2]
- Bureau du Plan.

Pour SANDY, les 4 variables de bouclage résultent de l'application de l'algorithme de RAVELLI [3], décrit ci-dessous.

C'est un algorithme basé sur une renumérotation des équations qui permet de donner une structure quasi-triangulaire au modèle.

Après application des quatre premières étapes, nous obtenons le prologue et l'épilogue du modèle. Les étapes 5 à 8 déterminent les variables de bouclage.

1°) Algorithme de Ravelli

Etape 1 :

Déterminer E = matrice d'incidence du modèle

NL = nombre d'éléments non nuls sur chaque ligne (vecteur)

NC = nombre d'éléments non nuls sur chaque colonne (vecteur)

Initialiser NI(I) = I pour chaque I = 1,...,N.

Poser l = 0. l indique le nombre d'équations déjà placées dans le prologue.

Remarque : NI est un vecteur qui conserve l'ancienne numérotation après les permutations. Ainsi NI(I) = J signifie la ième équation de la nouvelle numérotation portait le numéro J au départ.

Etape 2

Si aucun des NL (I) n'est nul pour I=l+1 jusque N, aller à étape 3.

Sinon : a) poser l = l+1

b) permuter les lignes et colonnes de E de la manière suivante :

La ligne I telle que NL(I) est nul est momentanément stockée dans un vecteur auxiliaire.

La ligne I-1 est amenée alors en I

La ligne I-2 est amenée en I-1

La ligne 1 est amenée en l+1

Le vecteur auxiliaire contenant la ligne I est recopié ensuite à la position 1.
Nous effectuons la même opération sur les colonnes de E.

c) Permuter de la même manière les vecteurs NI et NC.

NI(I) est stocké dans IA. Puis NI(I) = NI(I-1) ... NI(l+1) = NI(l),

NI(l) = IA.

d) Déterminer NL (J) pour J = l+1... N où NL(J) = nb d'éléments non nuls sur la ligne J, comptés entre les colonnes l+1 et n.

e) Retourner à l'étape 2.

N.B. : Si I (avec NL(I)=0) était égal à 1, nous n'effectuerions pas les étapes 2B) ni 2C).

Etape 3

Si l=n . STOP. Le système est récursif. Nous résolvons les équations dans l'ordre NI(1)...NI(n).

Sinon : le prologue contient les équations NI(l)...NI(l) dans cet ordre:

- poser m=n (m indique le dernier numéro avant la première équation de l'épilogue;
- aller à l'étape 4.

Etape 4

Si aucun des NC(I), pour I allant de l+1 à m n'est nul, aller à l'étape 5

Sinon : soit I tel que NC(I) = 0. SI I=n, aller en c).

Sinon a) Permuter les lignes et colonnes de E comme suit :

- copier la ligne I dans un vecteur auxiliaire,
- copier ensuite les lignes I+1 à la place de la ligne I, (I+2) à la place de (I+1)...,m à la place de m-1,
- puis copier le vecteur auxiliaire à la place de la ligne m.
- faire la même chose sur les colonnes.

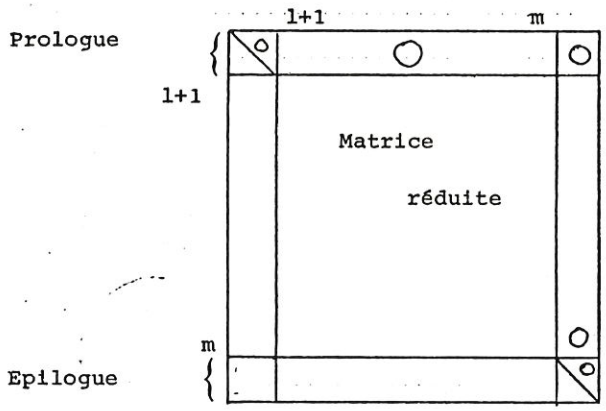
b) Permuter de la même manière NI,

c) Déterminer NC(J) pour J = l+1... m-1 en ne comptant pas les éléments 1 sur les lignes m à n.

d) Poser m=m-1.

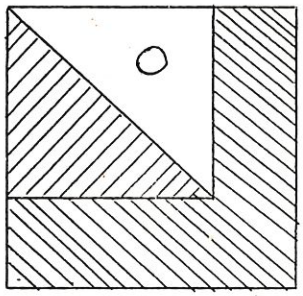
e) Aller à l'étape 4.

La matrice d'incidence a maintenant la forme suivante



La matrice réduite comprend les équations de l+1 à m (nouvelle numérotation) et les variables de l+1 à m.

Dans les étapes suivantes, nous allons déterminer les équations de bouclage sur cette matrice réduite, avec s le plus petit possible, c'est-à-dire donner la forme suivante à la matière réduite:



Etape 5

Déterminer un premier ensemble de s variables de bouclage.

Pour cela :

. poser s=0

. poser il=m

(*1) . pour chaque colonne i de il à l+1 (par pas de -1), examiner s'il existe un j avec l+1 $\leq$ j $\leq$ i tel que E (j,i) = 1.

Si oui . poser s=s+1 (nous avons une variable de bouclage en plus.
. si i = il, . poser m = m-1
. traiter la colonne suivante (i-1) : retourner en *1

sinon : i<il

*). mettre la colonne i de E (non réduite) dans un vecteur auxiliaire;

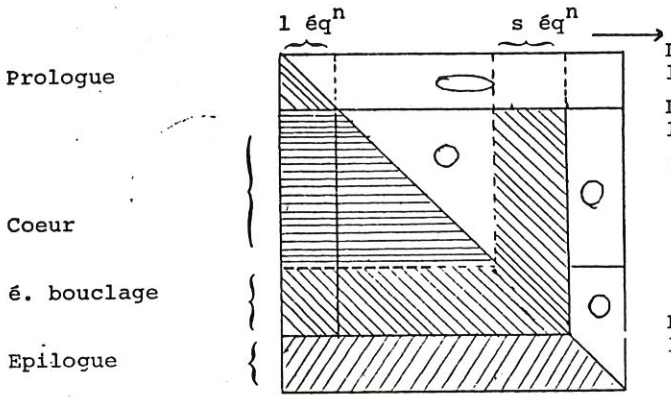
. amener la colonne i+1 en position i,

. amener la colonne i+2 en position i+1
. amener la colonne m en position m-1
. copier le vecteur auxiliaire à la place de la colonne m.

*) . permuter les lignes de E de la même façon
*) . permuter de même le vecteur NI
*) . poser m=m-1
*) . aller en *1 : traiter la colonne i-1

Sinon aller en *1 : traiter la colonne i-1

A ce stade, la matrice E a la forme suivante:



La première de ces s équations porte le numéro m+1

La dernière de ces s équations porte le numéro m+s

(Nous adoptons la nouvelle numérotation).

La dernière variable du coeur porte le numéro m

Le nombre s d'équations de bouclage peut encore être réduit. Nous allons tester chacune des variables de bouclage y_k^i, i=m+1,... m+s.

Pour déterminer les successeurs et prédécesseurs (étape 7), nous travaillons sur le coeur (lignes et colonnes 1+1 à m).

m indique la dernière équation du coeur, i, la prochaine variable à tester, il, la dernière position avant les variables de bouclage déjà testées (épilogue, si nous commençons à tester).

Etape 6

Posons i = m+1. Soit y_k^i, la variable de bouclage correspondante que nous testons. (Nous avons il = m+s à la fin de l'étape 5).

*1) Si E(i,i)=1 (c'est-à-dire ki=0), y_k^i est une vraie variable de bouclage:

. Permuter les lignes et colonnes de la matrice E (non réduite) comme suit :
- stocker la ligne i dans un vecteur auxiliaire,
- amener la ligne i+1 en position i,
- amener la ligne i+2 en position i+1,
- amener la ligne il en position il-1
- copier le vecteur auxiliaire en position il
- répéter ces permutations sur les colonnes.

. Permuter de manière similaire le vecteur NI
. Poser il = il-1.
. Si il = m. STOP . Toutes les variables de bouclage ont été testées.
Sinon : retourner en *1, étape 6
(i n'est pas modifié : il indique toujours la prochaine variable à tester).

*2) Sinon (E(i,i)=0))

- déterminer P = l'ensemble des prédécesseurs de y_k^i
- déterminer S = l'ensemble des successeurs de y_k^i
- Si l'intersection de S et P est vide, aller à l'étape 7.
  Sinon, aller à l'étape 8.

Nous détaillerons ce passage (*2) dans les commentaires qui suivent au § 2°).

Etape 7

La variable y_k^i est une fausse variable de bouclage.

- Ordonner E de manière à avoir dans l'ordre, les équations définissant P, l'équation i, les équations définissant S (voir § 2° ci-dessous).
- Permuter (NI) de la même manière.
- Poser s = s-1
- Poser i=i+1 (le nombre de variable de bouclage étant diminué de un, la position i de la prochaine variable à tester recule d'une place vers la droite).
- Poser m=m+1 (le coeur possède une équation en plus).
- Si il=m. STOP. Toutes les variables de bouclage ont été testées.
  Sinon retourner à l'étape 6 en (*1).

Etape 8

La variable y_k^i est une vraie variable de bouclage.
Nous effectuons les mêmes opérations qu'à l'étape 6 *1.

- permuter les lignes et colonnes de E pour amener la variable y_k^i en position il.
- permuter de la même façon NI
- poser il = il-1
- si il = m. STOP : toutes les variables ont été testées
  sinon : aller à l'étape 6, *1

Nous fournissons dans l'annexe 1, une copie du programme exécutant l'algorithme de Ravelli. Ce programme a été écrit en APL, par Jean-Marc PAUL au Bureau du Plan.

2°) Commentaires

La détermination d'une structure quasi-triangulaire équivaut à la détermination d'un ensemble de sommets dans le graphe associé, tels que le graphe réduit correspondant soit sans circuit. Par graphe réduit, nous entendons le graphe obtenu en retirant les sommets déterminés et les arcs dont ils sont les extrémités.

L'étape 2 construit le prologue. La méthode est analogue au procédé de décomposition en niveaux du graphe associé.

Nous déterminons les sommets qui n'ont pas de précédent (niveau zéro). Ils correspondent à une ligne nulle (NL(i)=0).
Nous amenons les équations correspondantes dans le prologue aux places indiquées par 1.
Nous réduisons le graphe en ne considérant plus que la matrice d'incidence, sans les lignes et colonnes correspondant aux équations du prologue. Ceci équivaut à supprimer les sommets sans précédent du graphe associé et les arcs les joignant.
Nous répétons l'opération jusqu'àce que ce ne soit plus possible.

L'étape 3 teste la récursivité du système.

Si nous avons un système non récursif, nous allons déterminer l'épilogue à l'étape 4. L'étape 3 initialise alors m=n. m indique ici le numéro de la dernière équation avant l'épilogue (partiel).

L'étape 4 détermine l'épilogue, par un procédé semblable à celui de l'étape 2.

Le sommet j est un suivant du sommet i, dans un graphe, s'il existe un arc(i,j). Ceci équivaut au fait que l'élément e_{ji} égale un.

Nous déterminons les sommets qui n'ont pas de suivant, c'est-à-dire les colonnes nulles de la matrice d'incidence.

Nous plaçons les équations correspondantes dans l'épilogue, en position m.

Au premier passage dans l'étape 4 nous travaillons sur le graphe réduit (dans lequel les sommets correspondant aux variables du prologue ont été enlevés ainsi que les arcs les joignant). En déterminant alors les sommets sans suivant, nous découvrons des variables endogènes qui ne doivent pas être connues pour la résolution des autres équations. Il est donc naturel de les placer à la fin de la matrice d'incidence, pour qu'elles soient résolues en dernier lieu.

Nous ne considérons plus ces sommets lors des passages suivants : nous réduisons la matrice en posant m = m-1. Il suffira de ne tenir compte désormais que des lignes et colonnes 1+1 à m.

Nous répétons l'opération jusqu'à ce que ce ne soit plus possible.

Les sommets sans suivant déterminés dans le graphe réduit, lors des passages ultérieurs dans l'étape 4, correspondent à des variables endogènes dont la valeur ne doit pas être connue pour résoudre les équations de la matrice réduite. Elles peuvent éventuellement devoir être déterminées avant de résoudre l'épilogue partiel construit antérieurement. Nous les plaçons donc juste avant l'épilogue partiel (en position m).

Les étapes 5 à 8 déterminent les variables de bouclage, c'est-à-dire les variables qui vont être un endroit de cassure des circuits du graphe associé.

Les étapes 6 à 8 vont réduire le premier ensemble de variables de bouclage obtenu à l'étape 5.

L'étape 6 détermine les variables de bouclage telles que ki égale zéro et réduit la matrice d'incidence.
Ces variables sont de vraies variables de bouclage.

Parmi les variables de bouclage restantes, certaines forment des circuits avec d'autres variables. Ce sont les vraies variables de bouclage. Déterminons-les Considérons la variable i, appartenant au premier ensemble de variables de bouclage. Il faut la tester. Pour vérifier, si elle forme un circuit, il faut déterminer ses prédécesseurs et ses sucesseurs et voir si l'intersection de ces deux ensembles est non vide(étapes 6 et 8).

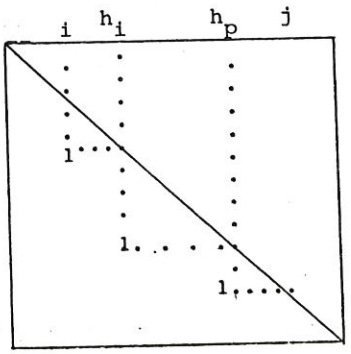
Si cette intersection est vide, la variable i est une fausse variable de bouclage. Par des permutations, nous l'amenons dans le coeur (étape 7).

La détermination des successeurs et des prédécesseurs s'effectue sur la matrice d'incidence.

a. Prédécesseurs

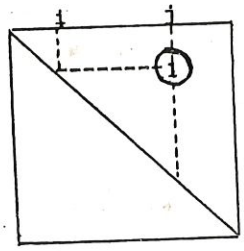
. i est un prédécesseur de j si $i < j$ et s'il existe $h_1, \ldots h_p$ tel que $i < h_1 < h_2 < \ldots < hp < j$ et les éléments $e_{h_1 i}$, $e_{h_2 h_1}$, $\ldots$ $e_{j h_p}$ la matrice d'incidence valent un.

E =



b. Successeurs

i est un successeur de j si i< j et $e_{ij} = 1$.

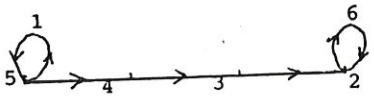


A l'étape 6, nous posons i=m+1. C'est la position de la première variable de bouclage dans la nouvelle numérotation.

Cet indice i indiquera toujours la variable de bouclage que nous allons tester. Vu les permutations et réductions, que nous allons effectuer, il n'y a pas de danger de tester deux fois la même variable de bouclage. Dès qu'une variable est testée, nous amenons la variable de bouclage suivante en position i et la variable testée est amenée à la fin de la matrice que nous réduisons d'une dimension. Nous allons éclaircir la manière de permuter les équations lorsque nous avons une fausse variable de bouclage (étape 8).

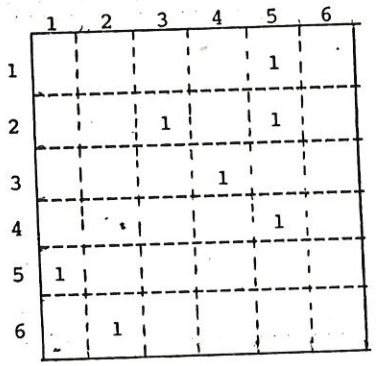
Rappelons qu'à ce moment, l'intersection de S et P est vide.

Ceci correspond au graphe suivant :



Nous n'avons pas de sommet sans précédent, ni de sommet sans suivant.

Dès lors, ces équations font partie de la matrice réduite = E (prologue et épilogue vides).



Les variables 3, 4, 5, 6 sont de bouclage.

Mais pour 3, 4, les ensembles S et P ont une intersection vide.

3 et 4 sont de fausses variables de bouclage car elles n'appartiennent pas à un circuit.

Nous pouvons résoudre le système de manière récursive sans attribuer de valeur à 3 et 4(avec 5 et 6 connus). 5 connu permet de déterminer 4 pour résolution de l'équation 4, puis 3, puis 2 (puisque 6 et 3 sont alors connus). Il suffit de changer l'ordre des équations.

Pour permuter les équations, nous travaillons avec la nouvelle numérotation. Pour déterminer les successeurs et prédécesseurs de i, nous travaillons sur la matrice formée des colonnes et lignes l+1 à i et utilisons la nouvelle numérotation.

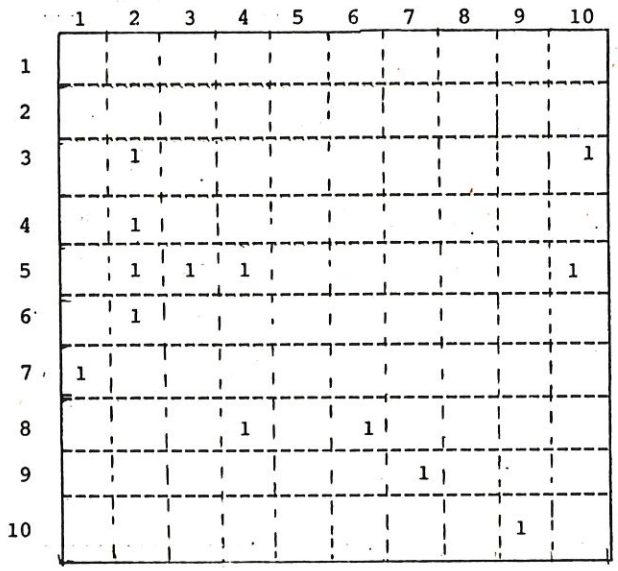
Considérons la variable i. Pour déterminer ses successeurs, nous cherchons les éléments 1 dans la colonne i, en remontant à partir de i - 1 jusque l+1.

Pour les prédécesseurs, nous travaillons "en escalier", c'est-à-dire, partant de l'élément (i,i), nous cherchons un élément 1 à gauche dans la ligne i. Si nous en trouvons un, soit j, nous repartons de la même manière à partir de l'élément (j, j).

Examinons la manière d'ordonner les éléments de P, l'équation i, les élements de S, à l'étape 7.

Remarquons d'abord que si P est vide (les éléments à gauche de (i,i) dans la ligne i sont nuls, alors automatiquement i est une fausse variable de bouclage. Nous plaçons la ligne i et la colonne i juste avant l'indice du prmier successeur de i (l'élément eji = 1 avec j minimal (i fixé) fournit l'indice j en question).

Si S est vide, il n'y a pas de permutation à faire.
Considérons l'exemple suivant pour lequel la matrice E est déjà réduite.



Nous testons la variable 10. (nouvelle numérotation).
Les variables de bouclage non testées portent les numéros 11, 12... Par conséquent, 1 et 2 n'appartiennent pas au prologue mais au coeur (imaginons par exemple que les éléments (1, 11) et (2, 12) valent un).
Pour la même raison, 5 et 8 n'appartiennent pas à l'épilogue.

Nous testons la variable 10.

$S = \{3, 5\}$
$P = \{1, 7, 9\}$
$S \cap P = \emptyset$; 10 est une fausse variable de bouclage.

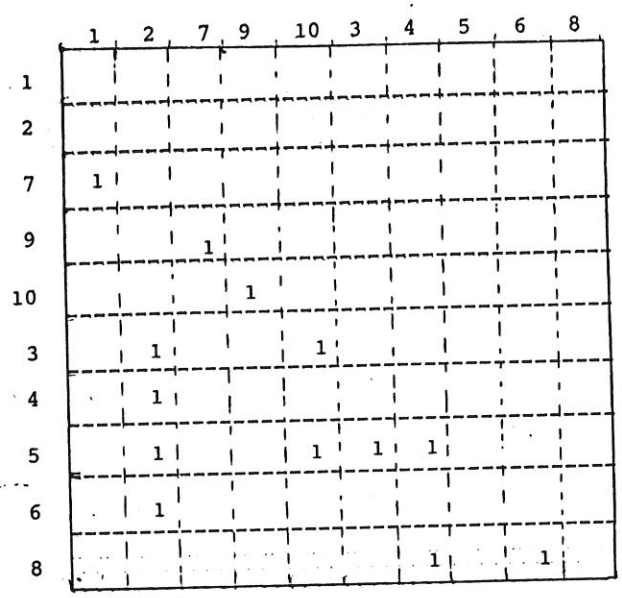
Il faudra avoir dans l'ordre 1, 7, 9, 10, 3, 5 en perturbant au minimum la matrice pour ne pas créer de nouvelles variables de bouclage.
(D'autres numéros s'intercaleront éventuellement entre les indices 1, 7, 9, 10, 3, 5).

Pour cela nous détectons le plus petit élément de l'ensemble des successeurs : s*. Comme nous travaillons avec la nouvelle numérotation, les équations sont rangées dans l'ordre croissant.

Ici, c'est 3.
Nous amenons (10) juste avant 3 (permuter les lignes et les colonnes).
Nous cherchons alors parmi les prédécesseurs ceux qui sont plus grands que s*.
Ici, 7 et 9 sont supérieurs à s* = 3.
Nous amenons les lignes et colonnes 7 et 9 juste avant l'équation i, laquelle précède 3 immédiatement) en maintenant l'ordre croissant.
Ici nous ordonnons donc dans l'ordre :

1 2 7 9 10 3 4 5 6 8

Ceci donne la matrice d'incidence :



Pour montrer que la manière d'ordonner, proposée ci-dessus, n'ajoute pas de variables de bouclage il faut seulement examiner le cas où il y a eu perturbation de l'ordre croissant des équations.

Pour 10, il n'y a pas de problème : les successeurs 3 et 5 sont placés après et les prédécesseurs 1, 7, 9, avant par construction.

Nous avons détruit l'ordre croissant pour 3, 4, 5, 6 qui suivent 7, 9, 10.

Pour que 3 ne devienne pas variable de bouclage, il faut que les éléments (7, 3), (9, 3), (10, 3) ne soient pas nuls dans la matrice de départ.

C'est le cas. Si (7, 3) était nul, 3 aurait été un prédécesseur de 10. De même pour (9, 3) et (10, 3). Le même raisonnement s'applique pour les variables 4, 5, 6.

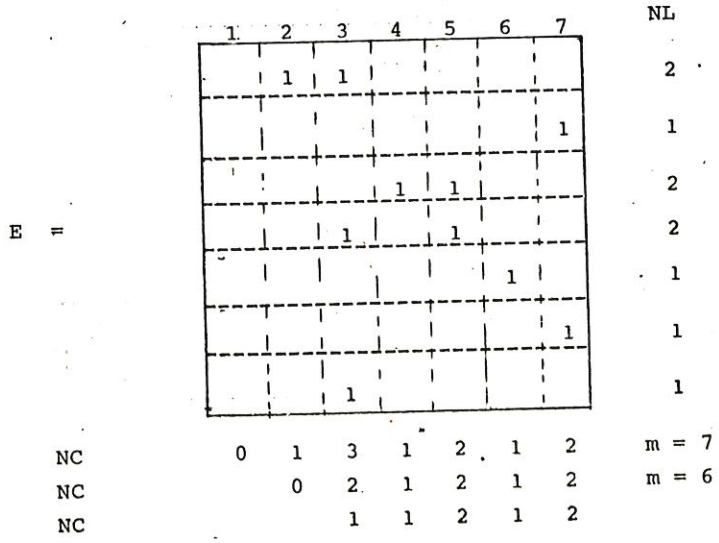
Pour la variable 8, l'ordre croissant a été brisé puisque 8 vient après 9 et 10.

De nouveau (9, 8) et (10, 8) ne pouvaient être nuls dans E, sinon 8 aurait été prédécesseur de 10.

3° Exemple

Nous allons illustrer les différentes étapes de l'algorithme à la main, sur le modèle (2.7).

A la figure 6, nous indiquons les différentes valeurs de NL et NC, au cours des étapes de 1 à 4.



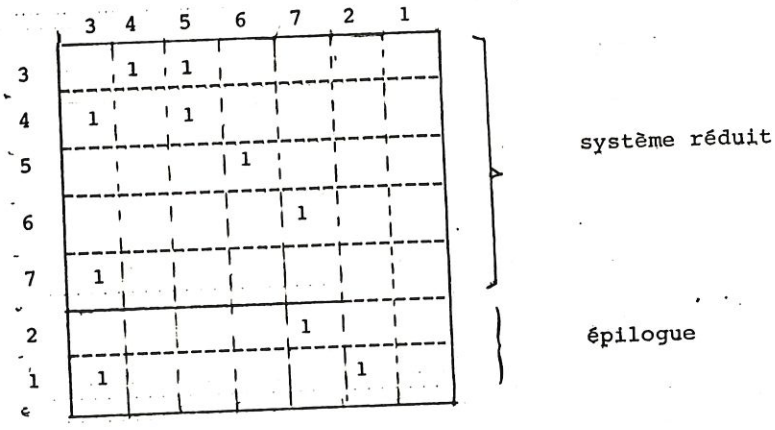
- figure 6 -

E ne possède pas de ligne nulle. Le prologue est donc vide (étape 2 et 3).

A l'étape 4, nous constatons que la première colonne de E est nulle; d'où, l'équation 1 est amenée en position 7 (dans l'épilogue).

Nous déterminons l'épilogue qui contient les équations {2, 1}

A l'étape 5, nous obtenons la matrice E' suivante :



- figure 7 -

Les étapes 6 à 9 sont effectuées sur le système réduit.

Etape 6

- Déterminer les variables de bouclage sur la matrice réduite réduite : 4, 5, 6, 7. Donc s=4.
- Les équations correspondantes étant à la fin de la matrice réduite, il ne faut rien permuter.
- $m^* = 1$

Nous allons tester les variables de bouclage : 4, 5, 6, 7

Etape 7

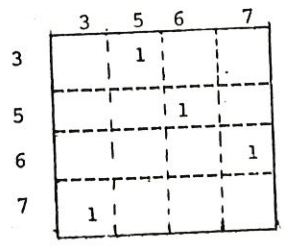
Pour tout i, ki = 1. D'où ki n'intervient pas dans la détermination des vraies variables de bouclage.

Variable 4

Prédécesseur : 3

Successeur : 3

La variable 4 est une vraie variable de bouclage. La matrice réduite devient :



Variable 5

Prédécesseur : vide

Successeur : 3

La variable 5 est une fausse variable de bouclage.
Nous l'amenons dans le coeur en ordonnant les équations dans l'ordre
5 - 3 - 6 - 7

La matrice d'incidence devient :

	5	3	6	7
5			1	
3	1			
6				1
7		1		

Variable 6

Prédécesseur : vide

Successeur : 5

En ordonnant les équations, comme suit : 6, 5, 3, 7, nous amenons l'équation 6 dans le coeur.
La matrice réduite devient:

	6	5	3	7
6				1
5	1			
3.		1		
7			1	

Variable 7

Prédécesseurs : 3 5 6

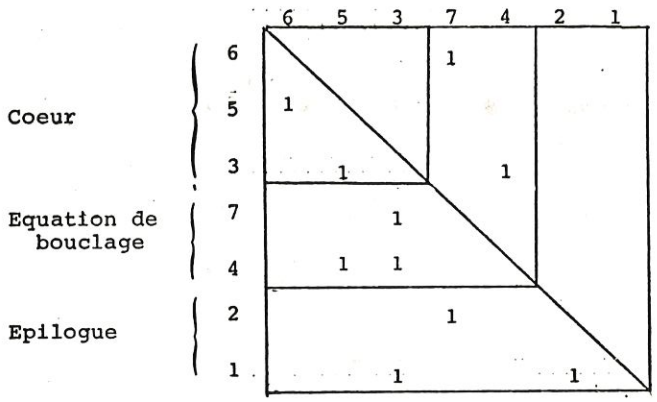
Successeurs : 6

La variable 7 est une vraie variable de bouclage.

Toutes les varaibles de bouclage ont été testées. STOP.
Nous avons déterminé :

- le prologue : vide
- le coeur : équations 6 5 3
- les équations de bouclage : 7 et 4
- l'épilogue : équations 1 et 2.

La matrice d'incidence du système global (figure 8) a la structure quasi-triangulaire :



- figure 8 -

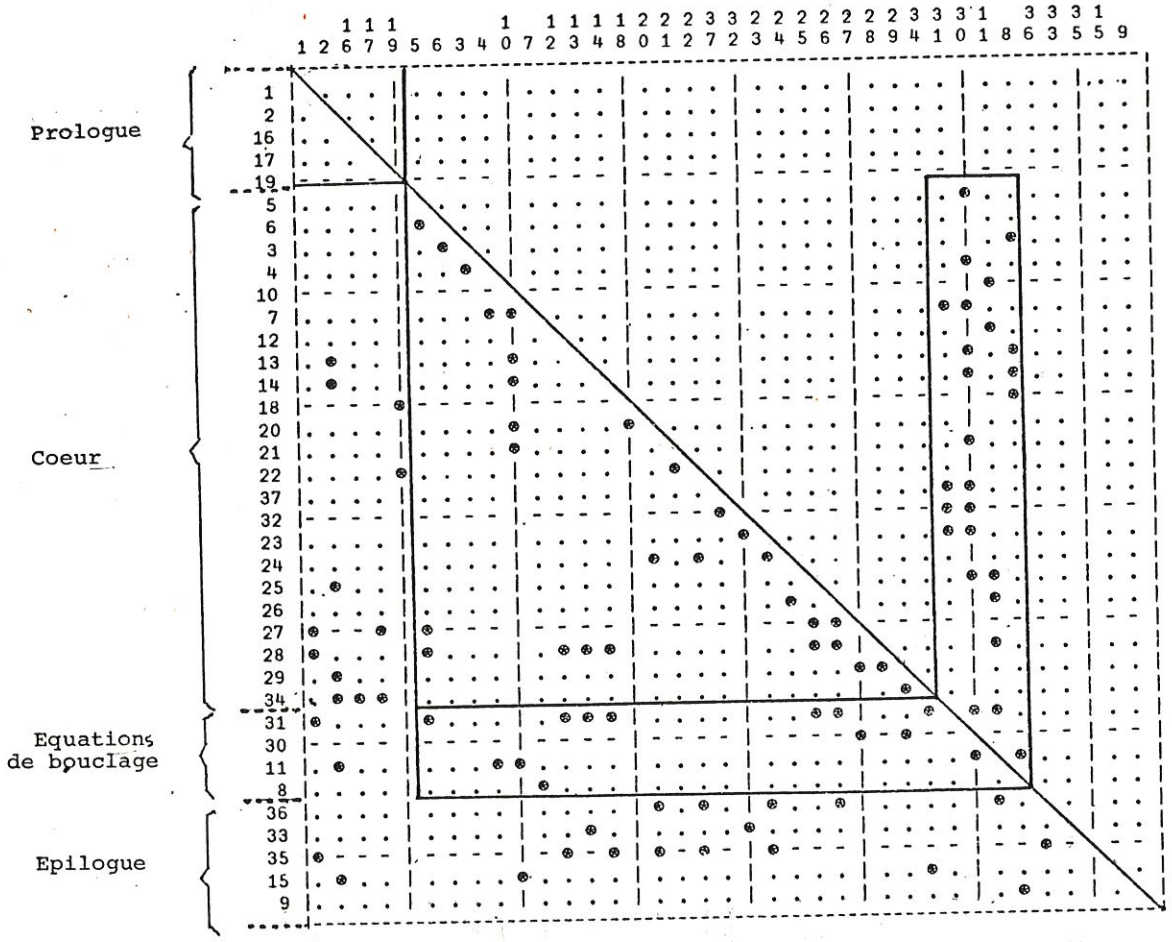
4° Résultats sur SANDY

L'application de l'algorithme de Ravelli au modèle SANDY permet de le rendre quasi-triangulaire. La matrice d'incidence associée à la structure quasi-triangulaire & est reproduite à la figure 9.

Nous avons déterminé dans l'ordre suivant :

- le prologue, constitué des 5 équations définissant :CG, PM, PX, XPORTS, LSE (dans cet ordre);
- le coeur, constitué des 23 équations définissant : INVF, KAP, OP, DUC, WR, LP, PG, PI, PIG, LW, YW, WSE, YSE, M1, RR, YK, YDISP, VARS, CP, AD, PID, MPORTS, BP (dans cet ordre).
- les 4 équations de bouclage définissant PGDP, GDP, PC, LAB.
- l'épilogue, constitué des 5 équations définissant TAX, CK, DEF, ICX, UN (dans cet ordre).

Matrice d'incidence de SANDY, après la mise sous forme quasi-triangulaire



- figure 9 -

Les numéros indiquent l'ordre de départ

⊛ représente 1

Les autres éléments sont nuls.

CHAPITRE 3 - RESOLUTION DU MODELE

La résolution du modèle fait partie d'un programme plus général qui résout le problème d'optimisation formulé au chapitre 1.

Résoudre le modèle signifie résoudre le système non linéaire (2.4): $Ky_k = f_k(y_k, d_k)$ pour chaque période k = 1,... T.

Plutôt que de travailler sur le système global, de 37 équations pour SANDY, nous allons exploiter la structure quasi-triangulaire, obtenue au chapitre 2, afin de réduire la taille du système non linéaire, non récursif à résoudre.

Nous décrirons ensuite brièvement quelques techniques de résolution d'un système non linéaire et en bous basant sur les travaux de Rachdi [6] et Gabay et al[1] nous comparerons l'efficacité des diverses méthodes proposées.

Dans un dernier temps, nous détaillerons les programmes de résolution du modèle SANDY.

1. REDUCTION DE LA TAILLE DU SYSTEME

Notations

Nous écrivons dans le sous-programme Fortran FUNCTION F (K, I, Z, D, YB, Y), les équations de SANDY, avec les notations du chapitre 2, paragraphe 3b et dans l'ordre qui rend le modèle quasi triangulaire (voir chapitre 2, figure 9).

Pour les équations du coeur et de bouclage, nous remplaçons de plus Y(K,M+I),I=1,... S, respectivement par YB(1), YB(2)* 1000, YB(3), YB(4)*2000, pour tenir compte du scaling(voir §3 de ce chapitre).

Pour cette même raison, nous divisons les équations de définition des variables endogènes Y(K, M+2) et Y(K,M+4) respectivement par 1000 et 2000.

Signification des arguments de F

. K = Indice indiquant la période pour laquelle nous allons résoudre le modèle.

. I = Indice indiquant le numéro de la variable endogène.
Par exemple, I=1 désigne la variable endogène CG (voir chapitre 2, paragraphe 3.C.4°).

. Z = Matrice (9 x 10) des variables exogènes lues dans le programme principal.
Les deux premières lignes contiennent respectivement les valeurs des variables exogènes pour les années 1977 et 1978.(Le retard maximal est de 2 périodes sur ces variables dans SANDY).

La ligne K+2, K=1,...7 contient dans l'ordre : DEP(K), ER(K), LF(K), LG(K) PWX(K), REX(K), TFRM(K), TFTSR(K), WT(K), TIME(K).

. D = Vecteur de dimension (2 x n) + (2 x r) =(2 x 37) + (2 x 5) = 84 qui contient toute l'information connue au moment de la période K.

$D = (y_{K-2}, y_{K-1}, u_{K-1}, u_K)$.
Il est connu au moment de l'appel à F.

. YB = Vecteur de dimension S=4, qui contient les valeurs des variables de bouclage à la période K, avec un scaling sur YB(2) et YB(4) (voir plus loin).

. Y = Matrice (T*N) = (7*37) dont l'élément Y(K,I) représente la valeur de la variable endogène I, à la période K.

Résolution du modèle pour la période k (en général)

En nous basant sur les travaux de Gabay et al(référence 2), nous allons proposer une technique de résolution du modèle, exploitant sa structure quasi-triangulaire. Pour SANDY, ceci est exécuté par le sous-programme RESOMO, dont nous fournissons une copie dans l'annexe 2.

Le vecteur $d = (y_{k-2}, y_{k-1}, u_{k-1}, u_k)$ est connu au début de la période k.
Nous commençons par résoudre le prologue (2.8) qui est récursif.

Par définition des variables de bouclage, nous savons que le coeur devient récursif

lorsque nous attribuons une valeur au vecteur $YB\{y_k^{m+1} \ldots y_k^{m+s}\}$.

Nous fixons YB à la période k, comme étant le vecteur formé par les valeurs des variables de bouclage à la période k-1.

Nous résolvons le coeur (2.9).
Nous obtenons une erreur sur les équations de bouclage :

$$(3.1) \qquad E_i(YB) = YB(i) - f_k^{m+i}(y_k^1 \ldots y_k^m, YB, d) \qquad i = 1 \ldots s$$

Pour obtenir les vraies valeurs des composantes de YB et par voie de conséquence de $y_k^{l+1} \ldots y_k^m$, il faut annuler cette erreur, c'est-à-dire résoudre

$$(3.2) \qquad E(YB) = 0$$

système non linéaire de dimension s. Nous utilisons pour cela une méthode itérative. A chaque itération, la j-ième approximation YB^j (vecteur) étant connue, nous résolvons le coeur (2.9), système triangulaire, et nous obtenons $E(YB^j)$ par la formule (3.1).

Finalement, nous connaissons y_k^i, i=1... m+s et nous résolvons alors l'épilogue (2.11) récursif, par simple application des formules le constituant.

L'intérêt évident de cette méthode réside dans le fait que nous utilisons une méthode itérative uniquement pour s équations au lieu des n de départ (s<n). Cela signifie notamment que la convergence se fera sur s équations, que le coût d'une itération sera moins élevé

2. RESOLUTION D'UN SYSTEME NON LINEAIRE PAR UNE METHODE ITERATIVE

La méthode exposée ci-dessus exige la résolution du système non linéaire (3.2).

Pour traiter ce problème, il faut utiliser des méthodes itératives, si le système n'est pas récursif.
Nous allons donner l'idée de quelques-unes de ces méthodes : méthode de Newton, Gauss-Seidel, des sécantes, de Brown.

Pour plus de détails, énoncés et démonstrations des théorèmes de convergence, estimation de l'erreur... nous renvoyons à Ortega-Rheinboldt [7].

a) Généralités

Les méthodes que nous allons décrire pouvant être appliquées aussi bien au système global (2.4) qu'au système réduit (3.2), nous utilisons la notation habituellement rencontrée dans la littérature :

Soit à résoudre le système non linéaire :

$$(3.3) \qquad \begin{cases} f_1(x_1 \ldots x_n) = 0 \\ \vdots \\ f_n(x_1 \ldots x_n) = 0 \end{cases} \qquad fi : R^n \longrightarrow R, i=1 \ldots n$$

que nous écrivons en abrégé $f(x) = 0$.

Nous notons x^* une solution du système (3.3) : x^* signifie $f(x^*) = 0$.

Pour résoudre (3.3), nous utilisons un processus itératif qui est défini par la donnée :

- d'une approximation initiale x° de la solution x^*
- d'un algorithme : $A : R^n \longrightarrow R^n$ tel que l'approximation x^{k+1} de la solution x^* est donnée par $x^{k+1} = A(x^k)$

Il faut aussi vérifier la propriété suivante :
Partant de x°, la suite des itérés (x^k) construite par le processus converge vers x^*.
A chaque itération, nous définissons une erreur : $e^k = x^k - x^*$.

Signalons que pour certains processus, il faut fournir les valeurs des itérés x°, x^1 ... x^p et l'approximation x^{k+1} est donnée par $x^{k+1} = A(x^k, x^{k-1} \ldots x^{k-p})$.

La plupart des méthodes procèdent à une linéarisation du système (3.3) autour de (x^{k-1}) pour obtenir x^k. Il s'agit de résoudre alors le système linéaire.

$$A(x - x^{k-1}) = f(x^{k-1})$$

La rapidité de convergence d'une méthode est mesurée par son ordre de convergence.

Supposons que $\lim_{k\to\infty} x^k = x^*$

Cette convergence est d'ordre α ($\alpha > 0$) s'il existe une constante $c>0$ telle que :

$$(3.4) \qquad \| x_{k+1} - x^* \| \leq C \| x^k - x^* \|^\alpha \text{ pour } k = 0, 1, 2\ldots$$

Si $\alpha=1$, la convergence est linéaire et elle est quadratique, si $\alpha=2$.

Elle est superlinéaire si $\alpha=1$ et si nous remplaçons (3.4) par (3.4)'.

$$(3.4)' \qquad \| x^{k-1} - x^* \| \leq C_k \| x^k - x^* \| \quad \text{pour } k = 0, 1, \ldots$$

$$\text{avec } \lim_{k\to\infty} C_k = 0$$

La norme $\| . \|$ est une norme quelconque sur R^n, par exemple

$$\| x \| = \max_i \{|x_i|\}$$

Si nous choisissons comme critère de sélection d'une méthode, un temps de résolution minimal , il nous faut trouver un compromis entre :

- le nombre d'itérations exigées qui diminue avec l'augmentation de l'ordre de convergence,
- et le coût de chaque itération qui lui augmente en général avec l'ordre de convergence.

Le nombre d'itérations augmente avec la précision exigée. Il est aussi plus élevé lorsque l'approximation initiale x° est éloignée de la solution x^*. Dans certains cas la convergence n'est même plus obtenue si nous partons trop loin de x^*.

Pour les modèles macroéconomiques nous prenons, comme approximation initiale, les variables correspondantes retardées d'une période. Ce choix est proche de la solution.

Quant au test d'arrêt de la méthode, c'est par exemple

- $\sqrt{\sum_{i=1}^{n} fi^2 (x_1 \dots x_n)} < \varepsilon$ qui implique automatiquement

$$\sqrt{\frac{1}{n} \sum_{i=1}^{n} (fi\ (x_1 \dots x_n))^2} \leq \varepsilon$$

- $\max / fi\ (x_1 - . x_n) / \leq \varepsilon$

Plus ε est petit, plus le nombre d'itérations est élevé. (ε petit signifie une grande précision).

La matrice de Jacobi est notée f'(x).
L'élément (i,j) de f'(x) est $\frac{\partial fi(x)}{\partial x_j}$

b) Méthode de Newton et ses variantes

Nous supposons que f est de classe C^1 (= continue et admettant des dérivées premières continues).

A chaque itération k+1 nous approchons $f(x^{k+1})$ par son développement de Taylor du premier ordre, au voisinage du point x^k.

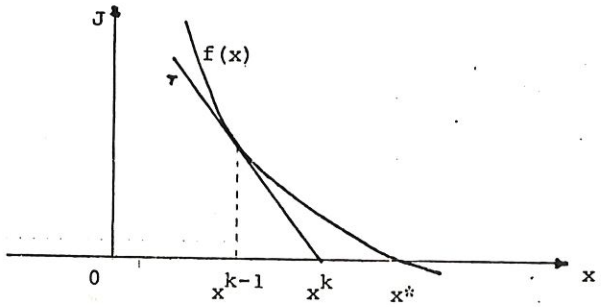
$$f(x^{k+1}) \simeq f(x^k) + f'(x^{k+1} - x^k)$$

Nous annulons cette approximation.
D'où $0 = f(x^k) + f'(x^k)\ (x^{k+1}) - x^k)$
et nous tirons x^{k+1} :

$$(3.5) \quad x^{k+1} = x^k - (f'(x^k))^{-1}\ f(x^k)$$

Il faut que la matrice jacobienne $(f'(x^k)$ soit non signulière pour tout k.

Nous allons illustrer géométriquement la construction des itérés dans le cas unidimensionnel. Soit à résoudre l'équation à une inconnue $f(x) = 0$.



x^k est l'intersection de la tangente T à la courbe y=f(x) au point d'abscisse x^{k-1} et de l'axe 0x.

On peut démontrer que pour avoir une convergence localement quadratique du processus (3.5), il suffit d'imposer que $f'(x^*)$ soit non singulière et de partir d'un x° assez proche de x^*.

Méthode de Newton modifiée

Le processus (3.5) converge en général en peu d'itérations mais il nécessite à chaque itération, le calcul et l'inversion de la matrice jacobienne $f'(x_k)$.

Le nombre d'opérations requis pour l'inversion d'une matrice par la méthode de Gauss-Jordan est de l'odre de n^3.

Pour diminuer ce coût élevé de temps calcul, lorsque x° est proche de x^*, nous pouvons remplacer (3.5) par (3.5)':

$$(3.5)' \qquad x^{k+1} = x^k - [f'(x^\circ)]^{-1} f(x^k)$$

Cette substitution permet d'éviter le calcul et l'inversion du jacobien de f à chaque itération.

Mais nous pouvons démontrer que dans ce cas, la convergence n'est plus que linéaire.

Autres variantes

1) L'algorithme (3.6) résulte d'un compromis entre (3.5) et (3.5)'. Nous calculons et inversons $f'(x^k)$ périodiquement.

$$(3.6) \qquad x^{k+1} = x^k - (f'(x^{p(k)}))^{-1} f(x^k)$$

avec $p(k) = [\frac{k}{po}] * po$ où [x] désigne la partie entière de x et po est donné.

2) La variante (3.7) est mise à profit dans le programme NS01A (NS01AD en double précision) de la bibliothèque de sous-routines scientifiques de Harwell[4].

$$(3.7) \qquad x^{k+1} = x^k - B_k f(x^k)$$

B_k est une approximation de $[f'(x^k)]^{-1}$

Nous partons de Bo = I (I = identité)

Pour la formule de Broyden, nous calculons

$$(3.8) \qquad B_{k+1} = B_k - B_k \frac{f(x^{k+1})\,(f(x^{k+1}) - f(x^k)^T}{(f(x^{k+1}) - f(x^k)^T\,(f(x^{k+1}) - f(x^k)}$$

où T indique une transposition du vecteur considéré.

On peut démontrer que la convergence est localement superlinéaire .

Newton discrétisée

Lorsque le jacobien $f'(x)$ est difficile à calculer analytiquement, nous utilisons l'algorithme

$$F(x^k, h_k)\,(x^{k+1} - x^k) = - f(x^k)$$

avec $$F_{ij}(x,h) = \frac{1}{h}\,[fi(x_1 \ldots x_{j-1};\ x_j + h \ldots x_n) - fi(x)]$$

$$i, j = 1 \ldots n$$

Moyennant des conditions sur f, sur x°, sur les pas de discrétisation, la méthode a une convergence localement quadratique [voir [1] pour un énoncé précis du théorème).

Nous économisons du temps de calcul en ne calculant que les éléments non nuls de la matrice jacobienne. Pour cela, nous pouvons utiliser la matrice d'incidence (voir chap. 2) puisque les éléments hors diagonaux non nuls de ces 2 matrices occupent les mêmes places.

c) La méthode de Gauss-Seidel

L'algorithme généralement utilisé est :

$$(3.9) \qquad x_j^{k+1} = \gamma_j \left(x_1^{k+1} \ldots x_{j-1}^{k+1}, x_j^k \ldots x_n^k\right) \qquad j = 1 \ldots n$$

Rachdi a montré [6] la convergence de cet algorithme.

La méthode de Gauss-Seidel présente l'avantage d'être facile à programmer et de ne pas faire intervenir de calculs de dérivées.

Mais sa convergence est en général lente et conditionnée par la forme des γ_i.

D'après les tests effectués par Gabay et al[1], sur PIMPON et STAR, elle est battue par toutes les méthodes de Newton, par celle des sécantes, pour le critère temps de calcul minimal pour la résolution.

d) Méthode des sécantes

Le principe de l'algorithme est facile à comprendre en dimension un.

Nous considérons L(x), interpolation de f entre x^{k-1} et x^k [$L(x^k) = f(x^k)$ et $L(x^{k-1}) = f(x^{k-1})$]

L(x) est donné par

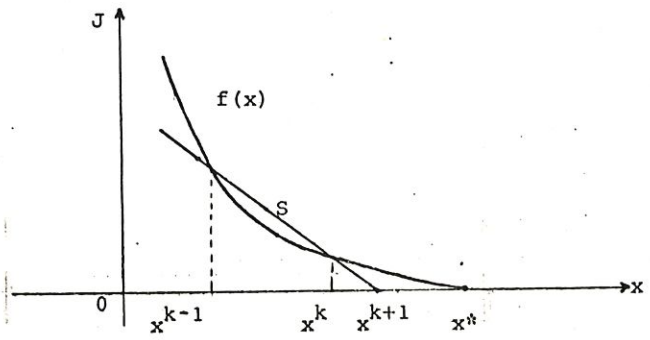
$$(3.10) \qquad L(x) = \frac{f(x^k) - f(x^{k-1})}{x^k - x^{k-1}} (x - x^k) + f(x^k)$$

x^{k+1} est solution de l'équation linéaire $L(x) = 0$ et est donc fourni par :

$$(3.11) \qquad x^{k+1} = x^k - \frac{x^k - x^{k-1}}{f(x^k) - f(x^{k-1})} f(x^k)$$

Comme $\frac{x^k - x^{k-1}}{f(x^k) - f(x^{k-1})}$ approche $[f'(x^k)]^{-1}$, nous avons une discrétisation de la méthode de Newton.

La méthode des sécantes est représentée schématiquement en dimension 1 par :



x^{k+1} est à l'intersection de l'axe 0x et de la corde S : $(x^k \quad x^{k-1})$.
Il faut fournir deux approximations, x° et x^1, au départ, la convergence est quasi-quadratique (en dimension 1).
Si le système à résoudre est de dimension μ, nous fournissons (n+1) points: $x^{k-n} \ldots x^k$ et nous remplaçons f(x) par (x) = Ax+b qui vérifie $L(x^{k-j}) = f(x^{k-j})$, j=0... n.

L'algorithme est :

$$x^{k+1} = x^k - H^k (G^k)^{-1} f(x^k) \tag{3.12}$$

avec

$$\begin{cases} \{H^k\}_{i,j} = x_i^{k-j+1} - x_i^{k-j} \\ \{G^k\}_{i,j} = f_i(x^{k-j+1}) - f_i(x^{k-j}) \end{cases}$$

$$i,j = 1 \ldots n$$

$H^k (G^k)^{-1}$ peut être considérée comme une approximation de $[f'(x_k)]^{-1}$. Ceci est évident en dimension 1.
La méthode des sécantes peut être interprétée, comme une discrétisation de celle de Newton.

Pour l'établissement du théorème de convergence, nous renvoyons à Rachdi[6]. Rachdi propose aussi une méthode de calcul de G_{k+1}^{-1} à partir de G_k^{-1}.

Contrairement à la méthode de Newton classique, la méthode des sécantes ne nécessite pas le calcul des dérivées.
La convergence est d'ordre α, compris entre 1 et 2 et tendant vers 1 lorsque n augmente (Rachdi[6]).

e) Méthode de Brown

Nous supposons que f est de classe C^1 et que $f'(x^*)$ est non singulière.

Soit x°, une approximation initiale de x^*.
En nous basant sur les travaux de Rachdi[6], nous allons décrire l'obtention de l'itéré x^1 par la méthode de Brown.

1ère étape

1) Nous linéarisons la première équation f_1 au voisinage de $x_\circ$.

$$f_1(x) \simeq f_1(x^\circ) + \sum_{i=1}^{n} \frac{\partial f_1(x^\circ)}{\partial x_i} (x_i - x_i^0) \tag{3.13}$$

2) Nous annulons cette approximation.
Nous résolvons l'équation ainsi obtenue par rapport à x_n.
x_n est tel que $\frac{\partial f_1}{\partial x_n}(x^\circ)$ a la valeur absolue maximale parmi les $\frac{\partial f_1}{\partial x_j}(x^\circ)$.
(Nous pouvons toujours permuter pour que ceci soit vrai).
Nous obtenons grâce à ces hypothèses que $(\frac{\partial f_1}{\partial x_n})(x_\circ) \neq 0$.

$$x_n = x_n^0 - \left[\sum_{i=1}^{n-1} \frac{\partial f_1}{\partial x_i} (x_i - x_i^0) + f_1(x^0) \right] \Big\backslash \left[\frac{\partial f_1(x^0)}{\partial x_n} \right] \tag{3.14}$$

3) La formule (3.14) définit une application affine

$L_n : R^{n-1} \longrightarrow R^n$ telle que $x_n = L_n (x_1 \cdots x_{n-1})$.

2ème étape

1) Nous définissons l'application $g_2 : R^{n-1} \longrightarrow R$ par

$$(3.15) \qquad g_2(x_1 \cdots x_{n-1}) = f_2(x_1 \cdots x_{n-1}, L_n(x_1 \cdots x_n))$$

avec

f_2 = 2ème équation du système $f(x) = 0$

2) Nous répétons le procédé :

- linéarisation de g_2 autour de $y^\circ = (x_1^\circ \cdots x_{n-1}^\circ)$
- obtention de x^{n-1} en annulant cette linéarisation et en extrayant x^{n-1} de la formule obtenue,
- définition de L_{n-1} telle que $x_{n-1} = L_{n-1} (x_1 \cdots x_{n-2})$.

Ne étape

Nous obtenons $g_n(x_1) = f_n (x_1, L_2 \cdots L_n)$

$= f_n (x_1, L_2(x_1), L_3(x_1, L_2(x_1)) \cdots)$

Nous avons à résoudre une équation à une inconnue x_1.

La méthode de Newton à une dimension s'écrit :

$$x_1 = x_1^0 - g_n(x_1^0) \,/\, \frac{\partial g_n}{\partial x_1} (x_1^0).$$

Elle nous fournit la première composante de l'itéré x^1.

Pour obtenir les autres composantes, nous remontons les équations, c'est-à-dire

$$\begin{cases} x_2^1 = L_2(x_1^1) \\ x_3^1 = L_3(x_1^1, x_2^1) \\ \vdots \\ x_n^1 = L_n(x_1^1 \cdots x_{n-1}^1) \end{cases}$$

Pour l'obtention de l'itéré $k+1$, à partir de la connaissance de x^k, nous répétons le procédé en remplaçant x^0 par x^k.

On peut démontrer que

si f est deux fois continûment différentiable, que $f'(x^*)$ est inversible et que x^0 est dans un voisinage de x^*, la méthode a une convergence locale quadratique.

f) Conclusions

Des tests effectués sur STARet FAIR, par Gabay et al[1] et Rachdi [6], il ressort que la méthode de Brown semble utiliser un minimum de temps de calcul, par rapport aux autres méthodes, dans le cas d'une moins bonne approximation initiale x^0.

Par contre, lorsque x^0 se rapproche de la solution, et pour une précision exigée élevée (de l'ordre de eps $\leq 10^{-7}$), la méthode des sécantes paraît la plus performante suivie par Newton modifiée (3.5)', Broyden (NSO1AD)(3.7, 3.8), Newton classique (3.5), puis loin derrière par Gauss-Seidel (3.9).

Pour une précision de l'ordre de 10^{-2} et une bonne approximation initiale, le classement par ordre d'efficacité décroissante est : méthode des sécantes, Broyden (NSO1AD), Newton modifiée, Newton classique, Gauss-Seidel.

Le test d'arrêt utilisé est $\sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_i(x_1 \ldots x_N))^2} \leq \text{eps}$

3. RESOLUTION DU MODELE SANDY

Pour un vecteur U donné, la sous-routine RESOMO (T, L, M, S, N, R, U) résout le modèle pour toutes les périodes 1 à T et crée la matrice Y des variables endogènes.

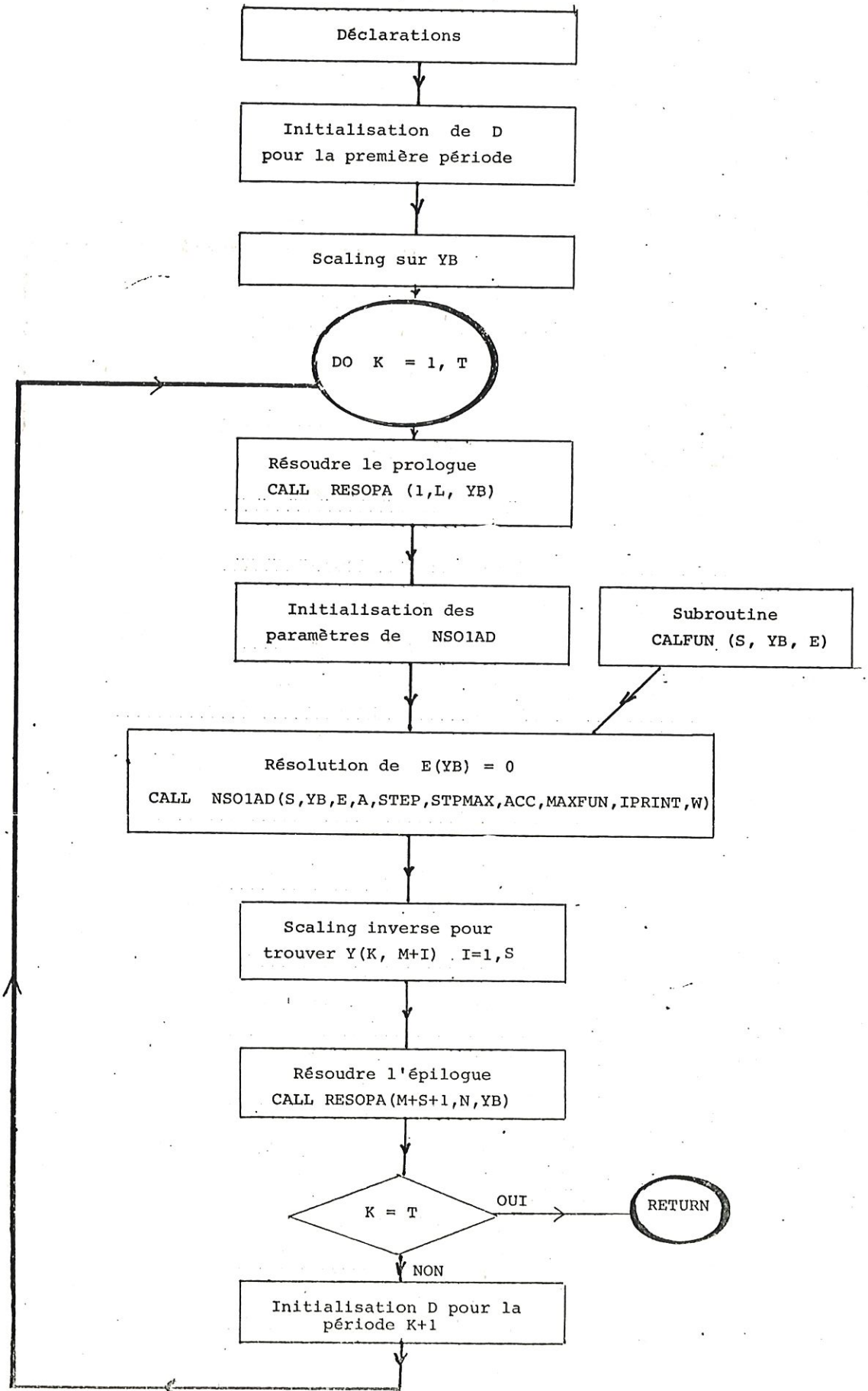
Table des variables de RESOMO

- Arguments : ils sont tous connus au moment de l'appel.

  T = nombre de périodes
  L = nombre d'équations du prologue
  M = numéro de la dernière équation du coeur
  S = nombre de variables de bouclage
  R = nombre d'instruments
  U = vecteur des instruments
  U(1) = IG(1); U(2) = x(1) ... (voir chapitre 2)
  U(6) = IG(2) ....
  .
  .
  .
  U(3/1)=IG(7) ....

- Les matrices Y et le vecteur D passent par COMMON et ont les mêmes significations que pour le sous-programme F (voir avant §1).
- D1 : vecteur de dimension 84, D1 = (y_{-1}, y_0, u_0, u_1) contient l'information connue au début de la première période.
- K : Indice de boucle, indiquant la période de résolution.
- YB : Vecteur des variables de bouclage après scaling. (voir plus loin)

ORGANIGRAMME DE RESOMO



La sous-routine RESOMO débute par une phase d'initialisation des vecteurs D et YB, pour la première période.

Puis, elle contient une boucle sur l'indice k.

A l'intérieur de cette boucle , RESOMO résout le modèle pour une période, suivant la technique décrite au paragrapge 1b de ce chapitre et ajuste le vecteur D pour la période suivante (sauf si k vaut T).

RESOMO appelle la sous-routine RESOPA pour résoudre les systèmes récursifs : prologue et épilogue.

La sous-routine RESOPA (I1, I2, YB) calcule les valeurs des variables endogènes : Y(K,J) pour J-I1, I1+1...., I2) par simple application de leur formule de définition.

```
      SUBROUTINE RESOPA (I1,I2,YB)
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION YB(4)
      COMMON/CLEN/Y(7,37),Z(9,10),D(84),K
C
      DO 10 I=I1,I2
      Y(K,I)=F(K,I,Z,D,YB,Y)
10    CONTINUE
C
      RETURN
      END
```

Pour l'annulation de l'erreur (E(YB) = 0), RESOMO appelle le programme NS01AD de Harwell [4].

Parmi les arguments en faveur de ce choix, nous citerons la facilité de programmation et la performance de la méthode.

Examinons les conditions d'utilisation de cette routine. Son but consiste à résoudre par la méthode de Newton, couplée avec la formule de Broyden (3.7 et 3.8), un système de N équations non linéaires à N inconnues :

$$f_i(x_1 \ldots x_N) = 0, \quad i=1 \ldots N.$$

L'utilisateur doit fournir une estimation initiale de $x_1 \ldots x_N$ dans un vecteur X et une sous-routine CALFUN (N, X, F) qui calcule chaque f_i c'est-à-dire établit $F(i) = f_i(x_1 \ldots x_N)$.

Parmi les exigences figure aussi la nécessité de travailler avec des variables $x_1 \ldots x_N$ du même ordre de grandeur.
Il en est de même pour les fonctions f_i.

Ceci est dû aux définitions du test d'arrêt (3.16) et de la distance entre deux estimations de la solution $(Y_1, \ldots Y_n)$ et $(Z_1 \ldots Z_N)$, donnée par $\sum_{i=1}^{N} (Y_i - Z_i)^2$

Si ces conditions ne sont pas remplies, la routine peut consommer plus de temps de calcul.

Comme approximation initiale YB, de la solution à la période k, nous conservons les valeurs des composantes de YB, obtenue à la période k-1. Les variables économiques fluctuant peu entre deux années consécutives, les solutions de l'année k-1 sont proches de celles de l'année k. Il suffit donc d'initialiser YB pour la première période (1979) puisqu'il sera automatiquement calculé par NS01AD pour les années suivantes.

Pour l'année 1979, nous l'initialisons avec les valeurs de 1978, à savoir :

PGDP = YB(1) = 1.8233

GDP = YB(2) = 1669.408

PC = YB(3) = 1.8041

λAB = YB(4) = 3749.667

Pour avoir des valeurs de YB(i),i=1,... S plus ou moins semblables, nous divisons YB(2) par 1000 et YB(4) par 2000.(= scaling).

Nous n'avons pas effectué de scaling sur les composantes E(i), définies dans CALFUN (S, E, YB) de la manière suivante : E(i) = YB(i) - F(K, M+1, Z, D, YB, Y),I = 1,S.

Nous ne divisons pas F par des constantes ici parce que nous l'avons déjà fait lors de l'écriture des équations dans le programme F.

La sous-routine CALFUN définit le vecteur erreur E(YB). La copie du programme CALFUN se trouve dans l'annexe 2. Pour cela, elle résout le coeur, système récursif puisque YB est déterminé, en appelant RESOPA(L+1,M,YB). Après le dernier appel de CALFUN par NS01AD, nous avons les vraies valeurs des variables du coeur dans la matrice Y et les valeurs des variables de bouclage, au scaling près, dans le vecteur YB.

Nous avons choisi une précision élevée ACC = $4*10^{-14}$ parce que la sous-routine RESOMO est utilisée par un programme d'optimisation. Si la précision était mauvaise nous pourrions avoir des problèmes au niveau de la convergence des méthodes d'optimisation.

En prenant ACC = $4*10^{14}$ pour le test d'arrêt (3.16) utilisé par NSO1AD,

(3.16) $\sum_{i=1}^{N} [f_i (x_1 \ldots x_n)]^2 \leq$ ACC nous obtenons une précision de l'ordre de $2*10^{-7}$ pour le test suivant :

$$\sqrt{\sum_{i=1}^{N} f_i (x_1, \ldots x_n)^2} \leq \quad eps = 2*10^{-7}$$

Après l'appel à NSO1AD, RESOMO définit Y(K, M+I), I=1,S à partir de YB, en effectuant le scaling inverse de celui décrit plus haut (Y(K,M+2) = YB(2) *1000

et Y(K,M+4) = YB(4)*2000).

Ceci exige une précision grande sur YB.

Si nous prenons successivement ACC = $4*10^{-10}$ et ACC = $4*10^{-14}$, le nombre d'itérations nécessaires à la résolution de E(YB) = 0 pour le modèle SANDY, passe de 9 à 11.

Nous avons défini le paramètre STPMAX de NSO1AD en lui attribuant la valeur 0.5.

Comme STPMAX représente une estimation généreuse de la distrance entre l'approximation initiale et la solution, nous avons examiné l'accroissement annuel des variables endogènes correspondant aux YB(i), pendant les période de 1953 à 1978. Nous avons constaté qu'il était toujours de loin inférieur à 0.5, valeur attribuée à STPMAX.

Après l'appel à NSO1AD, puis le scaling inverse sur YB, il ne reste plus qu'à résoudre l'épilogue récursif, en appelant RESOPA puis à définir D pour la période k+1.

4. SUGGESTIONS

Les seules contraintes sur la résolution du modèle sont sa mise sous forme quasi-triangulaire et un scaling sur les variables et équations de bouclage, si nous utilisons NS01AD.

Si nous voulons employer d'autres techniques pour résoudre un système non linéaire (Gauss-Seidel, sécantes,...) il suffit de remplacer l'appel à NS01AD par celui d'une sous-routine qui effectue la méthode en question.

Si nous voulons appliquer la méthode décrite au paragraphe 1, à un autre modèle, il faut au préalable le mettre sous forme quasi-triangulaire.

Dans le cas particulier des modèles macroéconomiques , le nombre d'évaluations de fonctions semble être le critère essentiel, pour le choix d'une méthode de résolution d'un système non linéaire, f(x)=0, de dimension n.

Dans la méthode de Newton discrétisée, il faut effectuer, à chaque itération, n évaluations de la fonction

$$f_i (x_1, \ldots x_{j-1}, x_j + h, x_{j+1} \ldots x_n) \text{ pour } j = 1, \ldots, n,$$

poue rechercher l'approximation du jacobien. La méthode de Broyden nous paraît plus intéressante parce qu'elle évite ces évaluations de fonctions supplémentaires, en introduisant la matrice B_k.

Le calcul de certaines variables du coeur peut être évité, lors de la recherche de l'erreur : E(YB) = 0.

A l'itération k+1 :

$$YB^{k+1}(i) = YB^{k}(i) - B^{k}E(YB) \qquad \text{(Broyden)}$$

où la définition de E(YB) exige la résolution du coeur (c'est notamment le cas lors de l'évaluation de la matrice jacobienne).

Si $YB^{k+1}(i)$ égale $YB^{k}(i)$, la variable de bouclage $YB^{k+1}(i)$ ne modifie pas les valeurs des variables du coeur, obtenues à l'itération précédente.

Nous cherchons alors le premier indice de la variable du coeur, modifiée par les autres variables de bouclage. Nous commençons la résolution du coeur, à partir de cet indice, pour calculer E^{k+1}.

Il est clair que de nombreuses améliorations pourraient être apportées au niveau de la programmation, mais nous pensons que cela dépasse le cadre de ce travail.

CHAPITRE 4 : OPTIMISATION DU MODELE

Dans ce chapitre, nous proposons une manière de ramener le problème avec contraintes, formulé au chapitre I, à un problème non contraint.

Nous examinons ensuite les conditions d'existence et d'unicité de la solution optimale.

Puis, nous rappelons les conditions d'extrêmalité d'un problème d'optimisation sans contrainte.

Nous détaillons des techniques de minimisation sans contrainte : steepest descent, Fletcher-Reeves, méthodes quasi-newton.

Nous concluons le chapitre en suggérant une méthode de pénalisation pour résoudre un problème contraint. Nous mentionnons une adaptation de cette méthode à la structure quasi-triangulaire du modèle.

1. CONDITIONS D'EXISTENCE ET D'UNICITE DE LA SOLUTION

Rappelons que nous sommes confrontés à un problème de minimisation sous contraintes, formulé au chapitre 1.
En utilisant les notations du chapitre 2, nous écrivons le critère, ou fonction objectif, ou fonction coût du modèle, sous la forme générale :

$$(4.1)\quad j(y,u) = \sum_{k=1}^{T} j_k\,(y_k, y_{k-1}, \cdots y_{k-p}, u_k, \cdots u_{k-q})$$

Les équations du modèle, données par (2.4), constituent les contraintes du problème d'optimisation.

Chaque fois que nous donnons une valeur au vecteur u, formé par les instruments pour toutes les périodes, nous pouvons résoudre le modèle (2.4) et obtenir la solution y comme fonction de u : y(u).

Nous pouvons supposer que y(u) est unique. Si ce n'était pas le cas, nous chercherions la solution $y_1(u)$ du modèle au temps 1, la plus proche de l'état initial connu $y_o = (y_o^1, \ldots y_o^n)$. $y_1(u)$ existe car d'un point de vue pratique il est inconcevable que le modèle n'ait pas de solution. Lorsque nous aurions obtenu $y_1(u)$, nous choisirions $y_2(u)$, solution du modèle au temps 2, la plus proche de $y_1(u)$ et nous continuerions de la sorte jusqu'à la période T.

Finalement, nous définirions $y(u) = (y_1(u), y_2(u) \ldots y_T(u))$ comme la solution du modèle. C'est un vecteur de dimension n*T.

Cette construction de la trajectoire y(u) repose sur l'hypothèse d'un comportement dynamiquement continu des variables économiques. Cela signifie que leurs valeurs varient peu entre deux périodes consécutives. Par conséquent, si les solutions du modèle ne sont pas uniques, il est naturel de construire y(u) comme ci-dessus.

L'unicité de y(u) permet de définir le critère.

$$(4.2)\quad J(u) = j(y(u),u) \quad \text{que nous minimiserons.}$$

Nous nous trouvons face à deux approches :

1) ou bien, nous considérons y et u comme des variables indépendantes et nous minimisons le critère j(y,u), soumis aux contraintes définies par les équations du modèle.
   Le critère j(y,u) étant connu explicitement, la méthode est alors une méthode explicite. Pour résoudre ce problème, nous proposerons la méthode de pénalisation (paragraphe 5);
2) ou bien, nous travaillons avec le critère J(u). Nous ramenons dans ce cas, le problème initial à un problème sans contrainte. y et u sont liés par les équations (2.4). Le critère J(u) n'étant connu qu'implicitement par l'intermédiaire du modèle, nous dirons que nous utilisons une méthode implicite.

Signalons que pour l'optimisation du modèle SANDY, nous avons choisi une méthode implicite, que nous détaillerons plus loin.

Existence d'une solution

Nous allons imposer les conditions suivantes qui nous assureront de l'existence d'une solution pour le problème de minimisation de J(u). Elles ne sont pas gênantes en pratique.

1) Nous introduisons un ensemble U formé par les vecteurs dont les composantes représentent des valeurs "raisonnables" des instruments.

Nous restreignons $y : U \longrightarrow y(U) = Y$.

Y est l'image par l'application y, de l'ensemble U, c'est-à-dire Y est l'ensemble des solutions du modèle pour tous les u dans U.

Ceci s'interprète en considérant Y comme l'ensemble des valeurs raisonnables des variables endogènes.
Dans U, nous ne gardons que des valeurs des instruments telles que toutes les fonctions f_k^i et j_k soient continues (et même de classe C^2 c'est-à-dire admettant des dérivées jusqu'au second ordre continues) par rapport à tous leurs arguments.

Par exemple, nous retirons de U, les valeurs des instruments qui produisent des singularités mathématiques sur les fonctions intervenant dans le modèle : dénominateurs qui s'annulent, logarithmes qui sont des arguments devenant négatifs....

2) Appelons R = r * T = le nombre total d'instruments pour toutes les périodes (R = 35 pour SANDY).

R^R est l'ensemble des vecteurs de dimension R à composantes réelles.
Nous définissons un compact de R^R comme une partie fermée et bornée.
Nous supposerons que l'ensemble U, introduit en 1), est un sous-ensemble compact et non vide de R^R.

Nous formulons notre problème comme suit :

(P) Trouver u* dans U tel que $J(u^*) \leq J(u)$ pour tout u dans U.

Avec les hypothèses mentionnées ci-dessus, nous pouvons appliquer la propriété bien connue suivante qui assure l'existence d'une solution au problème (P). (Théorème de Weierstrasse).

" Une fonction J continue d'une partie compacte U de R^R et à valeurs réelles est bor-
" née et atteint ses bornes en des points de U".

Par "J est bornée ", nous entendons qu'il existe m et M dans R (ensemble des réels) tels que $m \leq J(u) \leq M$ pour tout u dans U.

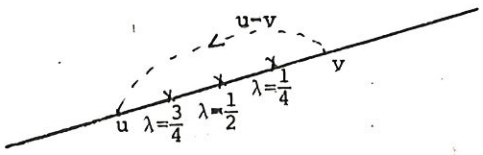
Par "J atteint ses bornes en des points de U" nous entendons qu'il existe des points u* et v* dans U tels que $m = J(u^*)$ et $M = J(v^*)$ c'est-à-dire il existe u* et v* dans U tels que

$$J(u^*) \leq J(u) \leq J(v^*) \quad \text{pour tout u dans U.}$$

Unicité de la solution

Pour examiner le problème d'unicité de la solution nous devons introduire quelques définitions.

1) Nous appelons combinaison convexe des vecteurs u et v, le vecteur $\lambda u + (1 - \lambda)v$ avec $0 \leq \lambda \leq 1$.
L'ensemble de toutes les combinaisons convexes des points u et v forment le segment de droite (uv).
Ceci est illustré par la figure 4.1.



- figure 4.1 -

Si $\lambda = 0$, nous obtenons v (0 u + 1 v = v).

Si $\lambda = 1$, nous obtenons u.

Si $0 < \lambda < 1$, $\lambda u + (1-\lambda) v$ équivaut à $v + \lambda(u-v)$.

Partant de v ($\lambda=0$), nous obtenons pour chaque valeur de $0 < \lambda < 1$, un point dans la direction u-v, situé à une distance λ du point v.

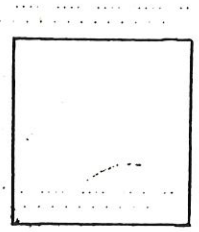
Si $\lambda = \frac{1}{2}$, nous avons le milieu du segment uv.

2) L'ensemble U est convexe si chaque fois que nous prenons deux points u et v dans U, toutes leurs combinaisons convexes sont dans U.

Ceci s'interprète géométriquement en disant : U est convexe, si le segment de droite qui unit deux points quelconques de U est entièrement dans U.

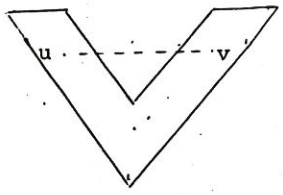
Exemples

U =



U est convexe

U' =



U' n'est pas convexe
Le segment (u v) n'est pas dans U'

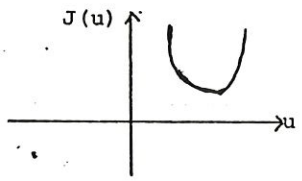
3) Une fonction J est strictement convexe si

$J = U \longrightarrow R$ (U est inclus dans R^R) est telle que pour tout u,v dans U, $u \neq v$, et pour tout λ avec $0 < \lambda < 1$, $J(\lambda u + (1-\lambda)v) < \lambda J(u) + (1-\lambda)J(v)$

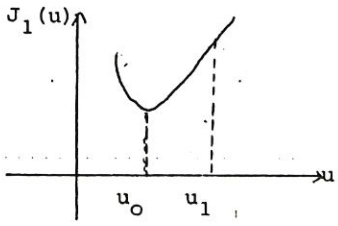
c'est-à-dire

l'image par J d'une combinaison convexe quelconque de points de U est strictement inférieure à la même combinaison convexe des images de ces points

Exemples :



(J(u) est strictement convexe.



$J_1(u)$ n'est pas strictement convexe car sa courbe est formée partiellement par un segment de droite $(J_1(u_o), J_1(u_1)$

Théorème 1.1. ("unicité de la solution")

> Si $J : U \longrightarrow R$ est strictement convexe.
> U est convexe
> Alors le problème min J(u), avec u dans U,
> admet au-plus une solution u^* dans U

Démonstration

1) Si le problème n'admet pas de solution, c.q.f.d.

2) S'il existe une solution, posons m = minimum de J(u)
   Introduisons Sm = {u dans U tels que $J(u) \leq m$ }
   Sm est convexe.

En effet :

Soit u et v dans Sm et λ tel que $0 < \lambda < 1$.

Il faut voir si $\lambda u + (1-\lambda)u$ est dans Sm?

$J(\lambda u + (1-\lambda)v) < \lambda J(u) + (1-\lambda) J(v)$ puisque J est strictement convexe.

Or, $J(u) \leq m$ et $J(v) \leq m$ pour définition de Sm.

Donc

$J(\lambda u + (1-\lambda)v) < \lambda J(u) + (1-\lambda) J(v) \leq \lambda m + (1-\lambda) m = m$

Donc

$\lambda u + (1-\lambda) v \in Sm$ par définition de Sm.

Nous allons démontrer l'unicité de la solution par l'absurde.
Supposons que u^* et v^* soient solution du problème.
Nous avons : $J(u^*) = m = J(v^*)$
Par conséquent u^* et v^* sont dans Sm.
Sm étant convexe $\frac{u^* + v^*}{2}$ est dans Sm.
Dès lors

(a) $$J\left(\frac{u^* + v^*}{2}\right) \leq m \leq J\left(\frac{u^* + v^*}{2}\right)$$

$\uparrow$ car m est minimum de J sur u et $\frac{u^* + v^*}{2}$ apaprtient à U puisque U est convexe.

Nous avons

$$m = J\left(\frac{u^* + v^*}{2}\right) < \frac{J(u^*) + J(v^*)}{2} = m$$

$\uparrow$ car J est strictement convexe.

Nous avons la contradiction : $m < m$.
Donc la solution du problème min J(u) avec u dans U est unique

C.Q.F.D.

Pour le problème (P) que nous devons résoudre, la solution u^* n'est en général pas unique. Même lorsque $j(y,u)$ est quadratique (convexe) comme dans SANDY, la fonction $J(u)$ ne sera en général pas convexe vu la non linéarité du modèle.

En pratique, les algorithmes itératifs d'optimisation convergeront vers l'optimum local (voir ci-dessous: définition) le plus proche de l'approximation initiale u° donnée.

Ceci est suffisant en pratique.

Nous allons définir un optimum local.

Définition

*) La boule ouverte $B(u,r)$ est l'ensemble des points situés à une distance de u strictement inférieure à r.

$$B(u,r) = \{ v : \| u - v \| < r \}$$

$\|\cdot\|$ désigne la norme euclidienne (par exemple) :

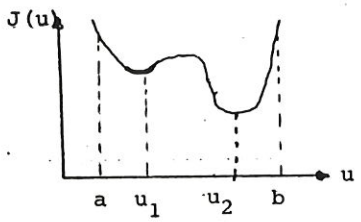
$$u = \{ \sum_{i=1}^{R} [u(i)]^2 \}^{1/2}$$

*) Un voisinage V de u est un ensemble comprenant une boule ouverte centrée en U.

*) Nous appelons minimum local de J sur U le point u^* tel qu'il existe un voisinage V de u avec $J(u^*) \leq J(u)$ pour tous les u dans l'intersection de V et U.

Nous appelons minimum global de J sur U le point u^* tel que $J(u^*) \leq J(u)$ pour tout u dans U.

Exemple : $J : U \longrightarrow R$ avec $U = [a,b] < R$



u_1 = minimum local de J

u_2 = minimum global de J

Pour SANDY, les instruments n'interviennent pas de manière explicite dans le critère qui s'écrit dès lors $J(u) = j(y(u))$.

Comme l'ensemble U n'est introduit que pour des raisons théoriques et qu'il peut être choisi aussi grand que nous voulons pourvu qu'il soit borné, la contrainte "u dans U" du problème (P) n'est pas active et nous sommes ramenés à un problème d'optimisation sans contrainte.

Nous allons examiner des conditions d'optimalité (paragraphe 2) puis nous exposerons quelques techniques numériques de minimisation sans contrainte (paragraphe 3).

2. CONDITIONS D'OPTIMALITE

Nous allons rappeler quelques notations puis deux théorèmes : le premier fournit des conditions nécessaires d'optimalité tandis que le second donne des conditions suffisantes. Pour plus de détails, nous renvoyons à Hestenes [8] qui démontre ces deux théorèmes.

a) Rappels et notations

*) J'(u) est le gradient du critère J au point u.

C'est un vecteur dont la i-ème composante $(J'(u))_i$ est la dérivée première de J par rapport à u_i.

$$(4.3) \quad (J'(u))_i = \lim_{h \to 0} \frac{J(u_1, \ldots, u_{i-1}, u_{i+h}, u_{i+1} \ldots u_R)}{h}$$

*) J'(u;h) est la dérivée directionnelle de J au point u dans la direction h (h est un vecteur de R^R.

$$(4.4) \quad J'u_i;h) = \lim_{\substack{t \to 0 \\ t>0}} \frac{J(u+th) - J(u)}{t}$$

Lorsque J est différentiable, alors

(4.5) J'(u;h) = (J'(u), h) où (.,.) désigne le produit scalaire de R^R.

*) J"(u) est la matrice hessienne (ou hessien) de J au point u.

L'élément (i,j) de J"(u) est donné par

$$(4.6) \quad (J"(u))_{i,j} = \frac{\partial^2 J(u)}{\partial ui \partial uj} \quad \text{(dérivée seconde de J par rapport à ui et uj)}$$

J étant de classe C^2, on peut démontrer que le hessien est symétrique (c'est-à-dire égal à sa transposée).

*) Nous notons J" (u;h) le produit matriciel

$$(4.7) \quad J"(u;h) = h^T J"(u) h.$$

*) Nous disons que la matrice A de dimensions (R,R) est définie positive si pour tout x, non nul dans R^R, le produit x^T A x est strictement positif.
(Si x^T Ax $\geq$ 0, on dit que A est semi-définie-positive)

b) Théorème (2.1) : (Conditions nécessaires d'optimalité)

> Si u* est un minimum local de J sur U
> J est de classe C^2.
> Alors pour tout h non nul dans R^R
> . J' (u*; h) = 0
> . J" (u*, h) $\geq$ 0

Ce théorème est une extension des conditions d'extrêmalité pour une fonction de R dans R (midimensionnelle) au cas d'une fonction de R^R dans R.

Démonstration

Idée : Rappelons, sans les démontrer, les conditions d'extrêmalité d'une fonction g d'une seule variable réelle et à valeurs réelles :

Si m est un minimum local de g, alors g'(m) = 0 et g"(m) $\geq$ 0.

Pour démontrer le théorème (2.1), nous allons nous ramener au cas unidimensionnel pour lequel les conditions sont admises.

Comme u^* est un minimum local, il existe un $\partial > 0$ tel que si $\|u-u^*\| < \partial$ alors $J(u^*) \leq J(u)$

Soit k non nul, quelconque dans R^R. Posons $\varepsilon = \frac{\partial}{k}$ réel.

Considérons $u = u^* + th$ avec $/t/ < \varepsilon$ (t est réel)

Ce point u est dans le voisinage de u^* défini par ∂.

Par conséquent $J(u) = J(u^*+th) \geq J(u^*)$ pour tout t avec $/t/ < \varepsilon$. (4.8)

Pour nous ramener au cas unidimensionnel, nous définirons la fonction $g(t)=J(u^*+ th)$ où $/t/ < \varepsilon$

(4.8) devient

(4.9) $\quad g(t) \geq g(0)$ pour $/t/ < \varepsilon$

(4.9) signifie que t=0 est un minimum local de la fonction d'une seule variable réelle g, sur l'intervalle $]-\varepsilon,\varepsilon[$

Appliquons les conditions d'extrêmalité pour cette fonction $\quad g'(0) = 0$

$g''(0) \geq 0$

Dès lors $g'(0) = 0$

$$\lim_{t\to 0} \frac{g(t)-g(0)}{0} = \frac{d}{dt} J(u^*+th) \Big|_{t=0} = (J'(u^*),h) = J'(u^*;h)$$

(Ces égalités résultent des définitions).

De même, $g''(0) = J''(u^*;h)$

C.Q.F.D.

Afin d'éviter une accumulation de détails techniques, nous allons énoncer le 2ème théorème sans démonstration. Le lecteur intéressé trouvera une démonstration dans [8].

c) Théorème 2.2. (Conditions suffisantes d'optimalité)

Supposons que u appartienne à U et vérifie

. $J'(u^*) = 0$ } pour tout h non nul

. $J''(u^*;h) > 0$ } dans R^R

Alors il existe $m>0$ et $\partial>0$ tel que

si $\|u-u^*\| < \partial$ alors $J(u) \geq J(u^*) + m \|u-u^*\|^2$

En supposant qu'au point optimal le gradient du critère soit nul et que son hessien soit défini positif, nous obtenons un minimum local strict.
Ce théorème n'est pas vraiment la réciproque du théorème 2.1. Les hypothèses sont renforcées et nous obtenons une conclusion plus forte.

3. METHODES NUMERIQUES D'OPTIMISATION SANS CONTRAINTE

a) Idée générale

Les algorithmes de résolution du problème non contraint (P) (min J(u) fonctionnent de la manière suivante :

A partir d'une approximation initiale u^0 de la solution u^* du problème, ils construisent une suite (u^k) d'itérés convergeant vers u^*.

Nous appelons <u>direction de descente d de la fonction J au point u</u>, le vecteur non nul d, de R^R, tel qu'il existe $\partial>0$ tel que $(u+\rho d) < J(u)$ pour tout $0<\rho<\partial$

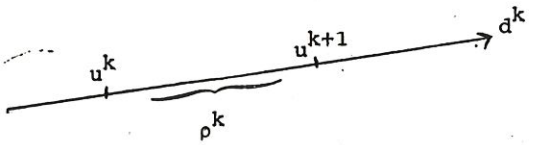
L'algorithme de minimisation construit alors u^{k+1} à partir de u^k par la formule :

$$(4.10) \quad u^{k+1} = u^k + \rho^k d^k$$

où . k est le numéro de l'itération

. d^k est une direction de descente de J au point u^k

. ρ^k est un pas de déplacement dans la direction d^k. Il mesure la distance entre u^{k+1} et u^k la direction d^k



ρ^k est positif

Partant de u^0, approximation initiale de u^*, nous nous déplaçons à chaque itération k , d'un pas ρ^k dans la direction d^k et nous diminuons la valeur du critère pour finalement arriver à une valeur proche de $J(u^*)$, valeur optimale du critère.

Il existe plusieurs manières de définir le test d'arrêt des méthodes (voir Bazaraa,Shetty [9]. Nous en citons quelques unes ci-dessous :

1) $\| u^{k+1} - u^k \| < \varepsilon$

ε est une tolérance fixée.
Ceci signifie que nous arrêtons parce que la distance parcourue après N itérations en partant de u^k est inférieure à ε.
"Nous restons pratiquement sur place au point u^k"

2) $\left\| \frac{u^{k+1} - u^k}{u^k} \right\| < \varepsilon$

La distance relative parcourue en une itération est inférieure à ε.

3) $J(u^k) - J(u^{k+N}) < \varepsilon$

Le critère ne diminue que d'une quantité inférieure à ε en N itérations.

Certains algorithmes ne nécessitent pas le calcul des dérivées du critère. Pour leur étude, nous renvoyons à Bazaraa,Shetty [9]. Nous allons proposer des algorithmes utilisant les dérivées parce qu'ils sont plus performants.

Nous allons montrer comment choisir la direction de descente d^k et le pas ρ^k d'une manière pratique.

*) Nous caractérisons une <u>direction de descente</u> d de J au point u comme un vecteur dont le produit scalaire avec le gradient J'(u) est négatif, c'est-à-dire que la dérivée directionnelle de J dans la direction d au point u est négative.

<u>En effet</u> :

Si $(d,J'(u)) < 0$

Alors $J(u+\lambda d) \simeq J(u) + \underbrace{\lambda}_{>0} \underbrace{(J'(u),d)}_{<0} < J(u)$

Par conséquent d est une direction de descente.

*) Quant au pas ρ, il est calculé par une minimisation unidirectionnelle.
Nous définissons la fonction $g(\rho) = J(u+\rho d)$.
Nous voulons diminuer la valeur de J.
Nous minimisons $g(\rho)$ en imposant $\rho>0$.

Ceci nous fournit ρ^k

ρ^k est donc la solution optimale du problème $\min_{\rho>0} J(u^k + \rho d^k)$. Cette minimisation est facile parce qu'il s'agit d'un problème unidimensionnel . Nous choisissons ρ^k qui annule la dérivée $g'(\rho)$.

Propriété

(4.11) Lorsque nous cherchons ρ^k par une minimisation unidirectionnelle, les directions de descente d^k sont orthogonales au gradient de l'itération k+1.

En effet

Il faut montrer que le produit scalaire $(d^k, J'(u^{k+1})) = (d^k)^T J'(u^{k+1})$ est nul, pour tout $k \geq 0$ (par définition d'orthogonalité).

Or nous calculons ρ^k tel que $g'(\rho)\Big|_{\rho^k} = 0$.

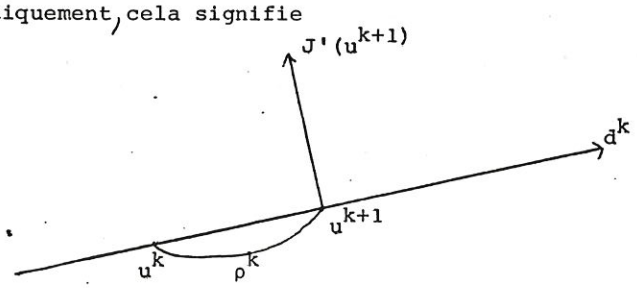
C'est-à-dire

$$0 = g'(\rho)_{\rho^k} = \frac{d}{d\rho} J(u^k+\rho\, d^k)\Big|_{\rho^k} = (d^k)^T J'(u^k+\rho d^k)\Big|_{\rho^k}$$

$$= (d^k)^T J'(u^k+\rho^k d^k) + (d^k)^T J'(u^{k+1})$$

C.Q.F.D.

Schématiquement, cela signifie



b) Méthode du gradient ("Steepest descent")

La direction de descente d^k est $-J'(u^k)$

Trivialement, $(d^k, J'(u^k)) < 0$ puisque $(d^k, J'(u^k)) = - \| J'(u^k) \|^2 < 0$.

L'algorithme s'écrit :

1) Initialisation

Poser $\varepsilon > 0$ (critère d'arrêt)
u° point de départ
$k = 0$
Aller à l'étape principale.

2) Etape principale

Si $\| J'(u^k) \| \leq \varepsilon$ STOP

Sinon

poser $d^k = -J'(u^k)$

chercher ρ^k tel que $J(u^k + \rho^k d^k) = \min_{\rho>0} J(u^k+\rho d^k)$

poser $u^{k+1} = u^k + \rho^k d^k$

poser $k = k+1$

répéter l'étape principale.

Dans la pratique, cette méthode semble moins intéressante et pas plus facile à programmer que la méthode de gradient conjugué (Fletcher-Reeves).

c) Méthode de gradient (direction) conjugué

Définition

Soit A une matrice symétrique d'ordre R.
Soit une suite de vecteurs de dimension R, indexée par i $\{p^{(i)}\}$
Nous dirons que les $p^{(i)}$ sont A-conjugués si le produit $p^{(i)^T} A\, p^{(j)} = 0$ pour tout i et j avec $i \neq j$.

Supposons que nous minimisons une fonction J(u) quadratique, c'est-à-dire s'écrivant sous la forme

$$(4.12) \quad \begin{cases} J(u) = \frac{1}{2}\, u^T Au + b^T u + c \\ \text{avec} \\ \quad \begin{cases} A & \text{matrice symétrique, définie positive} \\ b & \text{vecteur} \\ c & \text{constante} \end{cases} \end{cases}$$

Les itérés de l'algorithme des directions conjuguées sont fournis de la manière suivante :

1) Initialisation

Donner l'approximation initiale u°
Calculer $J'(u^\circ)$
Calculer d° tel que $(d^\circ, J'(u_\circ)) < 0$
(par exemple $d^\circ = J'(u_\circ)$)

Poser $k = 0$
Aller à l'étape principale.

2) Etape principale

Calculer ρ^k et u^{k+1} par les formules

$$J(u^k + \rho^k d^k) = \min_{\rho>0} J(u^k+\rho d^k)$$

$$u^{k+1} = u^k + \rho^k d^k$$

Calculer d^{k+1} de sorte que $(d^{k+1})^T Ad^j = 0$ pour $j=0,\ldots k$ (Les directions de descente sont A-conjuguées.

Poser $k = k+1$
Répéter l'étape principale.

On peut démontrer que si le critère J est quadratique (écrit sous la forme(4.12)) la méthode converge vers la solution optimale en au plus R itérations (R est la dimension de l'espace des vecteurs u), de manière quadratique.

Il est possible de construire les d^k comme combinaison linéaire de $J'(u^k)$ et d^{k-1} et aboutit finalement à la formule de Fletcher-Reeves.

$$(4.13) \quad d^k = J'(u^k) + \frac{\| J'(u^k) \|}{\| J'(u^{k-1}) \|} d^{k-1}$$

Nous pouvons appliquer cette méthode aux fonctions qui ne sont pas quadratiques.

Pour cela, nous posons $d^\circ = -J'(u^\circ)$, nous calculons $u^1 = u^\circ + \rho^\circ d^\circ$, puis $d^1 = -J'(u^\circ) + \frac{\| J'(u^1) \|}{\| J'(u^\circ) \|} d^\circ$.

Après R itérations, nous initialisons de nouveau $d^{R+1} = -J'(u^\circ)$ et nous répétons le procédé.

On peut démontrer [10] que la méthode converge aussi dans ce cas vers u^*, de façon superlinéaire. L'algorithme est implémenté dans la bibliothèque de Harwell [4] , sous le sigle VA14AD.

Cette méthode est surtout intéressante si la dimension R du problème est grande parce qu'elle n'exige qu'une faible réservation mémoire. Il faut stocker les vecteurs $J'(u^k)$, $J'(u^{k-1})$, d^k et d^{k-1}, tous de dimension R.

Si la dimension du problème n'est pas trop grande pour la mémoire centrale de l'ordinateur, nous utilisons plutôt une méthode quasi-newton (voir ci-dessous) pour résoudre le problème de minimisation sans contrainte. La raison majeure réside dans la plus grande efficacité de ces méthodes.

d) Méthodes quasi-newton

Dans ce paragraphe, nous allons expliquer l'idée des méthodes quasi-newton. Puis nous en proposerons deux : les méthodes DFP et BFGS. Nous admettrons sans démonstration que toutes ces méthodes convergent. Les démonstrations sont longues et techniques.

1° Idée

A chaque itération k+1, nous approchons la fonction à minimiser par une approximation quadratique autour du point u^k:

$$(4.14) : q(u) = J(u^k) + (J'(u^k), (u-u^k)) + \frac{1}{2} (u-u^k)^T J''(u^k) (u-u^k)$$

Nous minimisons la fonction q(u), c'est-à-dire nous cherchons u^{k+1} tel que $q'(u^{k+1})$ soit nul.

u^{k+1} est solution du système.

$$(4.15) : q'(u) = J'(u^k) + J''(u^k) (u-u^k) = 0$$

Ceci équivaut à définir u^{k+1} par

$$(4.16) : u^{k+1} = u^k - (J''(u^k))^{-1} J'(u^k).$$

La formule (4.6) est équivalente à l'algorithme de la méthode de Newton (voir chap. 3) appliquée à la condition d'extrêmalité $J'(u^k) = 0$.

Le calcul et l'inversion du hessien sont coûteux. Une inversion matricielle est de l'ordre de n^3 opération. Nous suggérons de remplacer $(J''(u^k))^{-1}$ par une matrice B^k de dimension R, telle que $\lim_{k\to\infty} B^k = (J''(u^k))^{-1}$.

Rappelons qu'une suite de matrices (A^k) converge vers la matrice A si les suites d'éléments (a_{ij}^k) de A^k converge vers l'élément (a_{ij}) de A pour tout i et j.

Nous obtenons alors

$$(4.17) \quad u^{k+1} = u^k - B^k J'(u^k) \text{ pour tout } k \geq 0.$$

Suivant le choix des matrices B^k, nous avons une série de méthodes, appelées quasi-newton pour une raison évidente.

Les matrices B^k sont définies de manière à vérifier les propriétés suivantes :

1) elles sont symétriques,
2) elles sont définies positives,
3) elles sont des approximations de l'inverse du hessien.
4) la matrice B^{k+1} s'ontient à partir de B^k en ajoutant une correction C^k.

$$B^{k+1} = B^k + C^k.$$

Le fait que B^k soit définie positive (propriété 2) nous permet de considérer la direction $d^k = -B^k J'(u^k)$ comme une direction de descente de J au point u^k.

En effet : le produit scalaire $(J'(u^k), -B^k J'(u^k)) = -(J'(u^k))^T B^k J'(u^k)$ est strictement négatif si B^k est définie positive.

Nous choisirons $d^k = -B^k J'(u^k)$ comme direction de descente. La pas ρ^k sera obtenu par une minimisation unidirectionnelle dans la direction d^k.

Nous définirons alors :

$$(4.18) \quad u^{k+1} = u^k - \rho^k B^k J'(u^k)$$

qui est l'algorithme des méthodes quasi-newton.

L'algorithme général des méthodes quasi-newton s'écrit :

u° donné; B° donné (en général B° = Identité)

≠1. Poser k = 0

≠2. Calculer $J(u^k)$ et $J'(u^k)$

≠3. Calculer $d^k = -B^k J'(u^k)$

≠4. Calculer ρ^k tel que $J(u^k + \rho^k d^k) = \min_{\rho>0} J(u^k + \rho d^k)$

≠5. Calculer u^{k+1} par la formule $u^{k+1} = u^k + \rho^k d^k$

≠6. Calculer $J'(u^{k+1})$

$$s^{k+1} = u^{k+1} - u^k$$

$$y^k = J'(u^{k+1}) - J'(u^k)$$

≠7. Calculer B^{k+1} tel que

- $B^{k+1} = B^k + C^k$

  C^k : matrice de correction ne dépend que de B^k, s^k, y^k
- B^{k+1} soit définie positive
- B^{k+1} vérifie l'équation quasi-newton définie par

  $B^{k+1} y^k = s^k$

" L'approximation de l'inverse du hessien appliquée à la différences des gradients entre deux itérations successives vaut la différence entre les itérés".

≠8. Poser k = k+1 et retourner au ≠3.

L'équation quasi-newton est en réalité le développement de Taylor du premier ordre autour de u^k, appliqué à la fonction $J' : R^R \longrightarrow R^R$ qui envoie u sur J'(u), gradient de J au point u dans lequel $(J''(u^k))^{-1}$ est remplacé par B^k.

En effet

Ecrivons ce développement.

$$J'(u^{k+1}) = J'(u^k) + J''(u^k)(u^{k+1}-u^k)$$

ou

$$[J''(u^k)]^{-1} \underbrace{[J'(u^{k+1}) - J'(u^k)]}_{y^k} = \underbrace{(u^{k+1} - u^k)}_{s^k}$$

Remplaçons $(J''(u^k))^{-1}$ par son approximation B^k :

$B^k\ y^k = s^k$: équation quasi-newton.

L'algorithme général ne peut être opérationnel que si le $\neq 7$ est spécifié. Il faut définir la matrice de correction C^k.
Nous allons détailler ce $\neq 7$. (voir 2° : méthode DFP et 3° : méthode BFGS).

2° Méthode DFP (Davidon - Fletcher - Powell)

Il reste à définir la matrice de correction C^k pour appliquer l'algorithme général (et y ajouter le test d'arrêt).

Lemme

La matrice $B^{k+1} = B^k + s^k (w^k)^T - H^k y^k (v^k)^T$
où

- s^k et y^k sont construits comme dans l'algorithme général
- $w^k \neq 0$ est tel que $(w^k)^T y^k = 1 = \sum_{i=1}^{R} (w_i^k)\ y_i$
- $v^k \neq 0$ est tel que $(v^k)^T y^k = 1$

vérifie l'équation quasi-newton.

Démonstration du lemme

Il faut voir si $B^{k+1} y^k = s^k$
or

$$B^{k+1} y^k = B^k y^k + s^k (w^k)^T y^k - B^k y^k (v^k)^T y^k$$

$$= B^k y^k + s^k - B^k y^k = s^k$$

C.Q.F.D.

Pour construire la matrice de correction C^k, il est naturel de choisir $C^k = s^k (w^k)^T - H^k y^k (w^k)^T$ après avoir précisé w^k et v^k.

Précisons la signification de $s^k (w^k)^T$:

$s^k = u^{k+1} - u^k$ est un vecteur $(s_1^k, \ldots s_R^k)$

w^k est un vecteur de dimension R.

Le produit $s^k (w^k)^T$ s'effectue comme suit :

$$\begin{pmatrix} s_1^k \\ s_2^k \\ \cdot \\ \cdot \\ \cdot \\ s_R^k \end{pmatrix} (w_1^k \ . \ . \ . \ w_R^k) = \begin{pmatrix} s_1^k w_1^k & s_1^k w_2^k \ \ldots\ldots & s_1^k w_R^k \\ s_2^k w_1^k & \ldots\ldots\ldots & \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ s_R^k w_1^k & \ldots\ldots\ldots & s_R^k w_R^k \end{pmatrix}$$

Davidon propose

$$w^k = \frac{s^k}{(s^k)^T y^k}$$

et

$$v^k = \frac{B^k y^k}{(y^k)^T B^k y^k}$$

Il est aisé de montrer que w^k et v^k vérifient les hypothèses du lemme et nous obtenons alors

$$(4.19) \qquad B^{k+1} = B^k + \frac{s^k (s^k)^T}{(s^k)^T y^k} - \frac{B^k y^k (B^k y^k)^T}{(y^k)^T B^k y^k}$$

vérifie l'équation quasi-newton.

On peut montrer que les matrices B^k construites par la formule 4.19) sont symétriques et définies positives.

Nous pouvons dès lors utiliser l'algorithme général en définissant B^{k+1} par la formule (4.19) au ≠7.

On peut montrer encore que si nous appliquons la méthode DFP à une fonction quadratique, $J(u) = \frac{1}{2} u^T Au + 1^T u + c$, les directions d^k, $k = 0,\ldots R-1$ générées sont A-conjuguées et que $B^R = A^{-1}$.

La méthode DFP étant alors une méthode de directions conjuguées, nous avons la convergence finale quadratique en au-plus R itérations.

3° Méthode BFGS

C'est une autre méthode quasi-newton qui est implémentée sous le sigle VA13AD de la bibliothèque de Harwell [4].

C'est cette méthode que nous avons utilisée pour l'optimisation du modèle SANDY parce qu'elle est connue pour être actuellement la plus efficace.

Cette méthode exige le stockage d'une demi-matrice de dimension R (=35 pour SANDY). En général, ceci ne pose pas de problème pour les modèles macroéconomiques, même si le nombre d'équations est élevé, car ici, c'est le nombre d'instruments, généralement faible qui importe. De toute manière, si la réservation mémoire est trop importante, nous utilisons alors VA14AD, basé sur la formule (4.13) de Fletcher-Reeves, qui demande un espace de travail de dimension 6R alors qu'il est de $\frac{R(R+13)}{2}$ pour VA13AD (voir plus loin).

Idée de la méthode

L'algorithme (4.20) de cette méthode est basé sur l'équation (4.17) et s'écrit :

$$(4.20) \quad A^k (u^{k+1} - u^k) + \rho^k J'(u^k) = 0$$

où A^k est une approximation du kessien $J''(u^k)$.

La matrice A^k est symétrique et stockée sous une forme factorisée $L^k D^k (L^k)^T$ où L^k et D^k sont respectivement des matrices triangulaires inférieure et diagonale. (stockage d'une demi-matrice de dimension R).

A^k est calculée à partir de A^{k-1} par une correction qui affecte L^{k-1} et D^{k-1}.

Quand A^k est connu, le pas ρ^k est calculé par une minimisation unidirectionnelle comme nous l'avons décrit antérieurement (chap. 3, § 3.a).

Utilisation du programme VA13AD : Optimisation de SANDY

Rappelons que pour le modèle SANDY, seules les variables endogènes non retardées interviennent dans le critère. Nous minimisons leur écart par rapport à une trajectoire désirée donnée.

Pour résoudre ce problème formulé au chapitre un, nous écrivons un programme qui :

*) lit les données nécessaires à l'optimisation :
   - approximation initiale du vecteur des instruments de dimension 35 : U. Rappelons que nous choisissons les valeurs des instruments constantes, à chaque période et égales à celles de 1978;
   - la matrice YD des valeurs désirées des variables endogènes intervenant dans le critère.

*) lit les autres données(utiles pour la résolution du modèle): D, Z, N, R, T, L, M, S dont la signification est donnée au chapitre 2;

*) initialise les paramètres de VA13AD : N4, SCALE, ACC (voir après);

*) appelle VA13AD(FUNC, N4, U, C, G, SCALE, ACC, W);

*) imprime les résultats.

Nous allons décrire les exigences pour utiliser VA13AD.

Cette routine poursuit le but de calculer la valeur minimale d'une fonction de plusieurs variables par la méthode BFGS. L'utilisateur doit fournir une sous-routine FUNC qui calcule la valeur du critère et de ses dérivées premières pour une valeur fournie des variables $(u_1 \ldots u_{N4})$. N4 est le nombre de variables.

Liste des arguments de VA13AD

. FUNC : nom de la sous-routine FUNC(N4,U,C,G) qui doit être fournie par l'utilisateur.

. N4, U, C, G ont la même signification pour FUNC et VA13AD :

  . N4 = nombre de variables.
  = R * T = 5 * 7 = 35 pour SANDY.

  . U = vecteur de dimension N4 qui au départ contient une approximation initiale de la solution (fournie par l'utilisateur).
  A chaque itération, U est modifié par VA13AD pour finalement contenir la solution optimale.

  . C = variable qui est automatiquement mise à la dernière valeur calculée du critère.

  . G = vecteur de dimension N4 qui est automatiquement mis aux valeurs du gradient du critère au point U.

. SCALE = vecteur de dimension N4, dont chaque composante doit être positive et indique un changement adéquat à apporter à U(i), initialement. Un bon choix est SCALE(i) = 10 % de la modification attendue pour U(i).

. ACC = variable qui définit la précision requise. La routine arrête les calculs quand des changements de l'ordre de ACC * SCALE(i) sur U(i) ne réduisent plus le critère.

. W = espace de travail de dimension $\frac{1}{2}$ N4 (N4+13) .
A la sortie, les $\frac{N4\,(N4+1)}{2}$ premières places contiennent l'approximation du hessien, sous forme factorisée.

La sous-routine FUNC, dont nous fournissons une copie à l'annexe 3, construit C et le vecteur G. Les dérivées premières du critère (composantes de G) doivent être précises sinon la méthode converge lentement et elle peut même ne pas converger vers un minimum du critère.

Nous détaillerons le calcul du gradient au paragraphe 4.
Le calcul du critère C=J(u) exige la résolution du modèle qui fournira y(u). Pour cela FUNC appelle la routine RESOMO (voir chap. 3), puis elle construit C en appelant les fonctions CNM et CT qui définissent le critère et dont nous fournissons une copie à l'annexe 3.

Nous avons exécuté le programme VA13AD avec ACC = 10^{-3}.

Pour obtenir SCALE(i), nous avons :

- pris la différence entre les valeurs de l'instrument U(i) (valeurs de 1978) et sa valeur en 1953;
- divisé le nombre obtenu par 25 pour avoir l'augmentation annuelle moyenne;
- multiplié par 7 le résultat pour avoir le changement attendu en 7 ans;
- divisé le résultat par 10 pour prendre 10 % de l'augmentation attendue.

Nous avons finalement :

SCALE(1) = 1.1073 représente 10 % du changement attendu pour IG(1979);
SCALE(2) = 0.0058 représente 10 % du changement attendu pour X (1979);
SCALE(3) = 0.00255 représente 10 % du changement attendu pour COTSSR(1979).

Et de même pour les 35 composantes.

4. CALCUL DU GRADIENT

Les méthodes décrites au paragraphe précédent demandent le calcul du gradient du critère.

Nous l'avons calculé par différences finies. Le calcul de son expression analytique était quasiment impossible puisque le critère J(u) = j(y(u)) n'était connu qu'implicitement à travers le modèle. Il aurait fallu dériver les équations du modèle.

Si nous appelons G le vecteur J'(u) (notation utilisée par VA13AD), la composante G(i) est définie par la formule suivante :

$$(4.21) \qquad G(i) = \lim_{k_i \to 0} \frac{J(u_1, \ldots u_{i-1}, u_i + k_i, u_{i+1} \ldots u_R) - J(u_1, \ldots u_R)}{k_i}$$

$$i = 1 \ldots R$$

Pour calculer G(i) numériquement, nous utilisons la formule approchée suivante :

$$(4.22) \qquad G(i) = \frac{J(u_1, \ldots u_{i-1}, u_i + k_i, \ldots u_R) - J(u_1 \ldots u_R)}{k_i}$$

Si k_i est petit, l'approximation du gradient fournie par la formule (4.22) est bonne.
Il reste à préciser le choix du pas k_i.
Nous définissons

$$k_i = \max (\text{DABS } U(i) * 10^{-5}, 10^{-5})$$

La calcul du gradient du critère est effectué par la sous-routine PARSH (U, C, NA, G) dont nous fournissons une copie à l'annexe 3.

C = valeur du critère au point U, connue au moment de l'appel.

NA = dimension du vecteur au point U, connue au moment de l'appel.

G = vecteur gradient au point U.

Interprétation du gradient

En examinant la formule (4.22), nous voyons que si k_1 vaut un, c'est-à-dire si nous augmentons la composante de l'instrument U(i) d'une unité, la composante i du gradient mesure l'accroissement correspondant du critère.

Par exemple, nous avons exécuté VA13AD sur le critère min $\frac{1}{2} \sum_{k=1}^{T} (UN(k)-\overline{UN(k)})^2$ où

UN(k) et $\overline{UN(k)}$ représentent respectivement le chômage et le chômage désiré en milliers de personnes.

VA13AD fournit comme résultats (entre autres) :

U(1) = IG(1979) = 60.779 (valeur optimale)

G(1) = 0.0019

J(u) = 0.000375

IG = investissements publics

G(1) = première composante du gradient

J(u) = valeur du critère à l'optimum

G(1) = 0.0019 signigie que si nous augmentons u(1) de 1 unité (c'est-dire u(1) = 61.779) alors le critère J(u) augmentera de 0.0019 et vaudra 0.000375 + 0.0019 = 0.002275.

La méthode du calcul du gradient par différences finies présente l'avantage d'une grande simplicité. L'écriture de la sous-routine PARSH comprend environ 10 lignes. C'est pour cette raison qu'elle est en général utilisée.

Mais le calcul du vecteur G nécessite le calcul du critère J(u), donc la résolution du modèle pour R=rT valeurs différentes du contrôle.
Comme il faut de plus calculer le gradient à chaque itération de la méthode d'optimisation, il apparaît clairement que la méthode peut être très coûteuse.
Nous pensons qu'il serait intéressant de poursuivre des recherches en vue de trouver d'autres méthodes de calcul du gradient qui n'exigeraient pas toutes ces résolutions du modèle.

Au cours des paragraphes précédents, nous avons étudié l'optimisation du modèle, par une méthode implicite. Il nous reste à proposer une méthode explicite : la méthode de pénalisation que nous décrivons au paragraphe suivant.

5. METHODE DE PENALISATION

Cette méthode permet de ramener un problème contraint à une série de problèmes non contraints. Elle pourra être appliquée au problème min j(y,u) soumis aux équations du modèle.

Elle restera applicable si nous ajoutons des contraintes sur les variables endogènes y et sur les instruments u.

Nous allons illustrer la méthode sur un exemple, puis nous en donnerons l'idée.
L'exemple est tiré de Bazaraa-Shetty [9].

Exemple

Soit à résoudre le problème (P1)

(P1) min x

sous-$x+2 \leq 0$ (c'est-à-dire $x \geq 2$)

Posons f(x) = x

g(x) = -x + 2

La solution du problème (P1) est $x = 2$.

Considérons $\alpha(x) = (\max(0, g(x)))^2$

c'est-à-dire

$$\alpha(x) = 0 \text{ si } g(x) \leq 0 \text{ (ou } x \geq 2)$$

$$\alpha(x) = (g(x))^2 = (-x+2)^2 \text{ si } g(x) > 0 \quad (x < 2)$$

Introduisons alors la fonction $f(x) + \alpha\, d(x)$

Ici, il s'agit de : . $x + \mu\,(-x+2)^2$ si $x < 2$

. x si $x \geq 2$

Nous allons minimiser cette fonction :

$$(4.23) \qquad \min f(x) + \mu\, \alpha(x)$$

Remarquons que si x est dans le domaine admissible (c'est-à-dire x vérifie les contraintes) nous minimisons la fonction de départ $f(x) = x$.

Nous n'ajoutons une pénalisation au critère que si la contrainte est violée ($x < 2$).

Nous avons un problème de minimisation sans contrainte (4.29) . Il peut être résolu par l'une des méthodes exposées au paragraphe 3.

Nous allons montrer sur l'exemple que si μ augmente indéfiniment, la solution du problème pénalisé converge vers la solution du problème initial.
Pour cela nous calculons la solution x_μ du problème (4.23).

Nous annulons la dérivée :

$$0 = (f(x) + \mu\, \alpha(x))' = (x + \mu\,(-x+2)^2)' = 1 - 2\mu(-x+2)$$

Donc

$$x_\mu = \frac{-1}{2\mu} + 2$$

La suite (x_μ) converge vers 2, solution optimale si μ tend vers l'infini.

L'algorithme de la méthode de pénalisation construira donc à chaque itération, μ_k de sorte que la suite (μ_k) indexée par k soit croissante et calculera x_{μ_k} solution du problème $\min f(x) + \mu_k\, \alpha(x)$.

La suite x_{μ_k} converge vers la solution optimale du problème.

Pour toutes les justifications théoriques de la méthode de pénalisation, nous renvoyons à Bazaraa - Shetty [9]

Idée de la méthode (optimisation de modèles macroéconomiques)

L'idée de la méthode consiste à considérer les équations du modèle comme des contraintes, les variables endogènes comme des contrôles supplémentaires et minimiser le critère $j(y,u)$ auquel nous ajoutons une pénalisation qui est un coût de violation des contraintes. Nous optimisons alors sans contrainte le critère pénalisé.

Lorsque le coût de violation des contraintes augmente indéfiniment, la solution du problème pénalisé converge vers la solution du problème initial (voir Bazaraa-Shetty [9] pour une démonstration.

Algorithme

1) Initialisation

Prendre . $\varepsilon > 0$ (critère d'arrêt)

. u° approximation initiale

. $\mu_1 > 0$ (paramètre de pénalisation)

. scalaire $\beta > 1$ (pour augmenter μ_1)

Poser k = 0 et aller à l'étape principale.

2) Etape principale

1°) - Calculer u^{k+1} comme solution optimale de $\min_{u \text{ dans } U} J(u) + \mu_k \alpha(u)$

- Aller en 2°)

2°) Si $\mu_k \alpha(u^{k+1}) < \varepsilon$ STOP

Sinon

$\mu_{k+1} = \beta \mu_k$

$k = k + 1$

aller en 1°)

6. SUGGESTION

Nous reprenons une méthode de pénalisation adaptée à la structure quasi-triangulaire du modèle proposé par Gabay et al [1].

En notant y_k^b, le vecteur des variables de bouclage et y_k^a, celui des autres variables endogènes à la période k, le modèle s'écrit :

$$(4.24) \qquad y_k^a = f_k^a (y_k, \ldots y_{k-p}, u_k \ldots u_{k-q})$$

$$(4.25) \quad L\ y_k^b = f_k^b (y_k \ldots y_{k-p}, u_k \ldots u_{k-q}) \qquad k = 1 \ldots T$$

où L est une sous-matrice de K correspondant aux variables de bouclage.
Les notations sont celles du chapitre 2.

Pour tout couple $v = (y^b, u)$ donné, la résolution du système récursif (4.24), de dimension n-s, permet de trouver y_k^a.
Ceci équivaut à la donnée d'une application $v \rightarrow \tilde{y}(v)$ qui est distincte par l'application $u \rightarrow y(u)$ construite au paragraphe 1 de ce chapitre, parce que en général, il y a une erreur sur les équations de bouclage.
L'erreur est donnée par :

$$(4.26) \qquad g_k(v) = L\ y_k^b - f_k^b (\tilde{y}_k(v), \ldots, \tilde{y}_{k-p}(v), u_k, \ldots u_{k-q}) \qquad k = 1 \ldots T$$

Il découle des définitions que :

$$(4.27) \qquad g_k(v) = 0 \quad \text{pour } k = 1, \ldots T$$

équivaut à

$$(4.28) \qquad y(u) = \tilde{y}(v)$$

La méthode décrite ici sera un mélange de la méthode de pénalisation et de la technique de construction du problème non contraint, min J(u), décrite au paragraphe 1.
$(J(u) = j(y(u), u))$.

La démarche décrite au paragraphe 1 consistait à résoudre le modèle, obtenir la solution $y(u)$ et écrire le problème (P) :

$$(P) \quad \min \; J(u) = j(y(u),u) \qquad \text{avec } u \text{ dans } U.$$

Ici, nous considérons le problème initial :

$$(P') \begin{cases} \min \; j(y,u) \\ \text{sous } K\, y_k = f_k\,(y_k, d_k)\,, \qquad k = 1 \ldots T \end{cases}$$

La résolution du système récursif (4.24) fournit une solution $\tilde{y}(v)$ qui est égale à $y(u)$ si $g_k(v) = 0$ pour tout k.

Dès lors, (P') équivaut à (P")

$$(P") \begin{cases} \min j(\tilde{y}(v),u) \\ \text{sous } g_k(v) = 0 \quad \text{pour } k = 1 \ldots T \end{cases}$$

Nous allons appliquer la méthode de pénalisation au problème (P").

Ceci revient à ne pénaliser que les équations de bouclage, les autres équations (contraintes) étant introduites dans le critère, à l'aide de $\tilde{y}(v)$ de la même manière que $y(u)$ était introduite dans $j(y,u)$ au paragraphe 1.

Nous obtenons le critère pénalisé (4.29) et le problème (P_ε) :

$$(4.29) \qquad J_\varepsilon(v) = j(\tilde{y}(v),u) + \underbrace{\frac{1}{\varepsilon} \sum_{k=1}^{T} \sum_{i=1}^{S} (g_k^i(v))^2}_{\substack{\text{coût de violation de la} \\ \text{contrainte } g_k(v)=0 \\ \text{pour } k=1 \ldots T}}$$

$$(P_\varepsilon) \begin{cases} \text{trouver } v_\varepsilon \in V \text{ tel que} \\ J_\varepsilon(v_\varepsilon) \leq J_\varepsilon(v) \text{ pour tout } v \in V \\ \text{où} \\ V \text{ est l'ensemble des couples } (y^b, u) \text{ "raisonnables"} \\ \\ \text{Par "raisonnables" nous entendons tels que les hypothèses, pour l'existence d'une solution, données au paragraphe 1 soient vérifiées.} \end{cases}$$

Comme V est aussi grand que l'on veut, pourvu qu'il soit borné, P_ε est un problème sans contrainte que nous pouvons résoudre par une des techniques proposées au paragraphe 3.

On peut démontrer que si $\varepsilon \longrightarrow 0$, la suite de solutions de (P_ε) converge vers la solution du problème (P).

Il nous reste à présenter les résultats numériques obtenus sur SANDY, c'est-à-dire à répondre à la question posée au chapitre 1. Ce sera l'objet du chapitre 5.

CHAPITRE 5 : RESULTATS

Dans ce chapitre, nous allons présenter les résultats des différentes simulations, effectuées sur le modèle SANDY.
Rappelons qu'une simulation est la résolution du modèle lorsque le vecteur des instruments est donné, pour toutes les périodes.
Les valeurs des instruments sont :

- soit des données (voir simulation de référence ci-dessous);
- soit des valeurs optimales, fournies par la résolution du problème d'optimisation suivant :

(P) Minimiser l'écart entre des objectifs (ici, variables endogènes) et leurs valeurs désirées, données, sous des contraintes, fournies par les équations du modèle macroéconomique.
(Ce problème (P) est traité en détail au chapitre 4.)

Nous avons effectué cinq simulations du modèle SANDY :

1°) Simulation de référence

Nous ne privilégions pas d'objectif. Les valeurs des instruments sont constantes pour chaque période de 1979 à 1985 et égales à celles de 1978 (données).
Il suffit donc de résoudre le modèle.

2°) Simulation avec 3 objectifs : Exportations nettes. Chômage. Déficit de l'Etat.
C'est la résolution du problème formulé au chapitre 1.

3°) Simulation : Objectif : Exportations nettes

4°) Simulation : Objectif : Chômage

5°) Simulation : Objectif : Déficit de l'Etat

Les résultats des différentes simulations sont reproduits dans les tableaux qui suivent.
Nous fournissons ensuite un tableau récapitulatif qui compare les résultats de ces cinq simulations.
Nos tableaux ne reprennent que les valeurs des variables économiques les plus importantes:

1. Produit intérieur brut et ses composantes
2. Commerce extérieur et balance des paiements
3. Production, emploi et salaires réels
4. Compte de l'Etat
5. Instruments de la politique économique.

Les remarques suivantes sont valables pour toutes les simulations :

1°) Les variables exogènes, autres que les instruments, sont considérées comme des données. Elles prennent les mêmes valeurs pour toutes les simulations.
2°) L'accroissement de la consommation publique (CG),(tableau 1.1) prend les mêmes valeurs pour toutes les simulations, puisque cette variable ne dépend que de l'emploi dans le secteur public (LG),qui est exogène.

La valeur élevée du taux de croissance de CG en 1979, égale à 9,6 %, provient de l'accroissement de LG entre 1978 et 1979. LG passe de 594.676 en 1978 à 626.52 en 1979; puis il se maintient constant à 628 jusqu'en 1985.

3°) L'observation du tableau 4, du Compte de l'Etat, nous permet de vérifier que la somme des dépenses équilibre la somme des recettes.
Les variations éventuelles de la trosième décimale résultent des erreurs d'arrondi.

1. REFERENCE INITIALE

L'observation des tableaux ci-dessous nous permet de voir que le déficit des exportations nettes (tableau 2) s'accroît. Ceci est dû à une croissance plus rapide des prix des importations par rapport aux prix des exportations, leurs volumes fluctuant de manière plus ou moins semblable.
Le chômage augmente (tableau 3), suite à l'augmentation de la population active et la diminution de l'emploi dans le secteur privé.
Le déficit de l'Etat continue à augmenter (tableau 4).

1. Le produit intérieur brut et ses composantes

1.1 *CROISSANCE EN PRIX DE* 1970

	1979	1980	1981	1982	1983	1984	1985
P.I.B	6.0	4.0	2.9	3.4	3.3	3.3	3.2
CONS. PR.	2.3	2.9	3.1	3.0	2.7	2.5	2.4
CONS. PUB.	9.6	.4	.0	.0	.0	.0	.0
INV. PRIVES	9.5	6.2	4.9	5.0	4.9	4.9	4.8
INV. PUB.	.0	.0	.0	.0	.0	.0	.0
EXPORTATIONS	6.6	5.0	3.7	5.6	5.6	5.7	5.7
IMPORTATIONS	4.3	5.0	4.6	5.4	5.3	5.3	5.2

1.2 *CROISSANCE DES PRIX*

	1979	1980	1981	1982	1983	1984	1985
PRIX P.I.B	5.8	6.6	6.0	5.7	5.6	5.4	5.3
PRIX CONS. PR.	6.7	7.1	6.2	5.9	6.0	6.0	6.0
PRIX CONS. PUB.	8.1	8.5	7.5	7.2	7.3	7.3	7.3
PRIX INV. PRIVES	8.6	8.9	7.4	7.1	7.2	7.2	7.2
PRIX INV. PUB.	10.1	10.2	8.6	8.3	8.4	8.4	8.4
PRIX EXPORTATIONS	11.3	11.2	7.8	7.8	7.6	7.4	7.3
PRIX IMPORTATIONS	14.5	13.0	8.7	8.7	8.7	8.7	8.7

2. Commerce extérieur et balance des paiements

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
VOL. COMM. MOND.(A)	6.20	4.00	3.00	5.00	5.00	5.00	5.00
PRIX COMM. MOND.(A)	15.20	14.00	10.00	10.00	10.00	10.00	10.00
TAUX CHA. (FB/US DOL)(B)	30.78	30.16	29.56	28.97	28.39	27.82	27.27

(A) en taux de croissance

(B) Chiffres en moyenne annuelle

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
PRIX COMM. MOND FB (A)	13.100	11.700	7.800	7.800	7.800	7.800	7.800
PRIX IMPORT.(A)	14.500	13.000	8.700	8.700	8.700	8.700	8.700
PRIX EXPORT. (A)	11.300	11.200	7.800	7.800	7.600	7.400	7.300
TERMES D'ECH.(A)	-3.200	-1.800	-.900	-.900	-1.100	-1.300	-1.400
VOL. EXPORT. (A)	6.600	5.000	3.700	5.600	5.600	5.700	5.700
VOL. IMPORT. (A)	4.300	5.000	4.600	5.400	5.300	5.300	5.200
EXPORT. NET. FB (B)	-59.772	-104.682	-160.488	-200.807	-253.640	-319.514	-400.053

(A) Taux de croissance

(B) En milliards de francs B.

3. Production, emploi et salaires réels

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
POP.ACTIVE	4102.100	4117.800	4144.700	4173.600	4200.100	4230.300	4253.600
EMPL. SEC. PUB.	626.520	628.000	628.000	628.000	628.000	628.000	628.000

(en milliers de personnes)

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
EMPL. SECT. PR.(A)	3142.762	3139.775	3136.479	3136.241	3136.349	3135.360	3132.674
CHOMAGE (A)	332.820	350.025	380.221	409.356	435.794	466.909	492.945
PRODUCTIV.(B)	5.400	4.000	3.000	3.400	3.300	3.300	3.300
SAL. REELS(B)	.000	.000	.000	.000	.000	.000	.000
TAUX UTIL. CAP. PROD.	.809	.814	.811	.812	.811	.810	.810

(A) en milliers de personnes

(B) en taux de croissance

4. Compte de l'Etat

DEPENSES

	1979	1980	1981	1982	1983	1984	1985
CONS. PUBLIQUE	647.692	706.000	759.196	814.123	873.640	937.594	1005.997
INV. PUBLICS	122.800	135.290	146.861	159.071	172.392	186.849	202.483
TRANSF. ETAT-MEN.	690.440	763.468	836.485	913.360	994.880	1081.620	1174.617
TOTAL	1460.932	1604.758	1742.542	1886.554	2040.912	2206.063	2383.097

RECETTES

	1979	1980	1981	1982	1983	1984	1985
IMPOTS DIRECTS	521.557	576.723	631.879	689.951	751.531	817.054	887.304
IMPOTS IND.	338.493	373.062	408.276	445.479	485.154	527.236	572.112
COTIS. SEC. SOC.	416.986	457.580	497.753	540.803	587.854	638.815	693.755
DEFIC. PUBLIC	183.897	197.394	204.633	210.322	216.374	222.958	229.927
TOTAL	1460.933	1604.759	1742.541	1886.555	2040.913	2206.063	2383.098

(en milliards de francs)B.)

5. *INSTRUMENTS DE POLITIQUE ECONOMIQUE*

	1979	1980	1981	1982	1983	1984	1985
INVEST. PUBL.(B)	54.279	54.279	54.279	54.279	54.279	54.279	54.279
SAL.REELS(A)	3.000	3.000	3.000	3.000	3.000	3.000	3.000
TAUX COT. SEC. SOC.	.176	.176	.176	.176	.176	.176	.176
TAUX TAX. DIR.	.190	.190	.190	.190	.190	.190	.190
TAUX TAX. INDIR.	.164	.164	.164	.164	.164	.164	.164

(A) en taux de croissance

(B) en milliards de francs 1970.

2. SIMULATION . 3 OBJECTIFS

Il faut remarquer les valeurs élevées et les fluctuations importantes des investissements publics (tableau 1.1.) Ceci provient du fait que c'est un instrument utilisé pour annuler le déficit des exportations nettes et le déficit de l'Etat et amener le chômage à une valeur de 50.000 personnes, en fin de période.

Au tableau 3, nous constatons que le taux de croissance de la productivité passe de 6.1 % en 1979 à -0.4 % en 1980.

L'accroissement de la productivité est proportionnel à celui du produit intérieur brut et inversément proportionnel à celui de l'emploi.
Entre 1979 et 1980, le taux de croissance de l'emploi varie peu.
Par contre, le tableau 1.1. montre que le taux de croissance du produit intérieur brut tombe de 8.6 % à 0.9 %. Ceci est dû notamment à la chute spectaculaire du taux d'accroissement des investissements publics qui passe de 57.5 % à -49.8 % (voir tableau 1.1).
Le produit intérieur brut dépend linéairement des investissements publics.

Au tableau 1.2., nous observons une ascension des valeurs des taux de croissance des prix du produit intérieur brut, de la consommation privée, de la consommation publique, entre 1979 et 1980. Le taux de croissance du prix de la consommation privée augmente en 1980 suite à la diminution du taux de croissance de la productivité à cette époque.
Le même effet se produit sur le taux de croissance du prix de la consommation publique puisqu'il est proportionnel à celui du prix de la consommation privée.
Les valeurs négatives de ces taux de croissance en 1979 proviennent, à notre avis, d'un problème de branchement :
Le taux de croissance d'une variable est,par définition, proportionnel au rapport entre les valeurs de la variable, prises à deux périodes consécutives.
Or, en 1978, les variables dépendent des instruments en 1978, lesquels sont donnés.
Par contre, en 1979, les valeurs des instruments sont calculées, comme solutions optimales d'un problème de minimisation.
La grande différence entre les valeurs données de 1978 et calculées de 1979, pour les instruments publics notamment, provoque ces valeurs négatives des taux de croissance des prix considérés.
Cet effet se répète pour l'objectif chômage, mais pas pour les objectifs, déficit des exportations nettes, ni déficit de l'Etat (voir plus loin , simulations suivantes).

Au tableau 3, nous constatons que le taux d'utilisation de la capacité de production diminue.
Or, le taux d'utilisation de la capapcité de production est le rapport entre le produit intérieur brut et l'"output potentiel".
Nous reprenons ci-dessous les taux de croissance de ces deux dernières variables.
Nous constatons que l'"output potentiel " augmente toujours beaucoup plus que le produit intérieur brut. Ceci provoque une diminution du taux d'utilisation de la capacité de production.

	1979	1980	1981	1982	1983	1984	1985
Produit intérieur brut (a)	8.6	0.9	2.6	2.7	2.4	2.7	3.1
Output potentiel (a)	6.5	5.1	5.5	5.6	5.5	5.6	5.4

(a) en taux de croissance.

L'output potentiel augmente suite à l'augmentation de l'emploi. Rappelons que nous minimisons le chômage.
Ce phénomène se répète aussi lors de la simulation où l'objectif privilégié est le chômage (voir plus loin).

1. Le produit intérieur brut et ses composantes

1.1 *CROISSANCE EN PRIX DE* 1970

	1979	1980	1981	1982	1983	1984	1985
P.I.B	8.6	.9	2.6	2.7	2.4	2.7	3.1
CONS. PR.	.0	1.7	.6	.4	1.1	1.9	3.6
CONS. PUB.	9.6	.4	.0	.0	.0	.0	.0
INV. PRIVES	10.6	4.2	5.0	4.8	4.7	4.9	4.9
INV. PUB.	57.5	49.8	23.4	14.4	16.7	34.6	55.3
EXPORTATIONS	6.5	5.0	4.1	5.6	5.7	5.7	5.7
IMPORTATIONS	2.9	1.5	2.8	4.0	4.1	4.1	4.0

1.2 *CROISSANCE DES PRIX*

	1979	1980	1981	1982	1983	1984	1985
PRIX P.I.B	5.6	10.2	3.8	5.2	6.7	5.4	3.4
PRIX CONS. PR.	5.8	12.7	3.6	5.8	7.2	5.7	4.6
PRIX CONS. PUB.	6.6	15.2	4.5	7.1	8.8	6.9	5.6
PRIX INV. PRIVES	.1	10.5	5.1	6.4	7.3	6.4	6.1
PRIX INV. PUB.	3.2	11.4	6.7	7.7	8.5	7.8	7.6
PRIX EXPORTATIONS	11.3	9.7	7.8	7.3	7.3	7.3	7.0
PRIX IMPORTATIONS	14.5	13.0	8.7	8.7	8.7	8.7	8.7

2. Commerce extérieur et balance des paiements

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
VOL. COMM. MOND.(A)	6.20	4.00	3.00	5.00	5.00	5.00	5.00
PRIX COMM. MOND.(A)	15.20	14.00	10.00	10.00	10.00	10.00	10.00
TAUX CHA. (FB/US DOL)(B)	30.78	30.16	29.56	28.97	28.39	27.82	27.27

(A) taux de croissance

(B) chiffres en moyenne annuelle

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
PRIX COMM. MOND FB (A)	13.100	11.700	7.800	7.800	7.800	7.800	7.800
PRIX IMPORT.(A)	14.500	13.000	8.700	8.700	8.700	8.700	8.700
PRIX EXPORT. (A)	11.300	9.700	7.800	7.300	7.300	7.300	7.000
TERMES D'ECH.(A)	3.200	3.300	.900	1.400	1.400	1.400	1.700
VOL. EXPORT. (A)	6.500	5.000	4.100	5.600	5.700	5.700	5.700
VOL. IMPORT. (A)	2.900	1.500	2.800	4.000	4.100	4.100	4.000
EXPORT. NET. FB (B)	34.920	29.170	23.240	17.390	11.570	5.860	.060

(A) Taux de croissance

(B) En milliards de francs B.

3. Production, emploi et salaires réels

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
POP.ACTIVE	4102.100	4117.800	4144.700	4173.600	4200.100	4230.300	4253.600
EMPL. SEC. PUB.	626.520	628.000	628.000	628.000	628.000	628.000	628.000

(en milliers de personnes)

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
EMPL. SECT. PR.(A)	3211.350	3261.150	3323.820	3388.630	3450.650	3516.480	3575.670
CHOMAGE (A)	264.230	228.650	192.880	156.960	121.490	85.780	49.950
PRODUCTIV.(B)	6.100	.400	1.000	1.100	.800	1.100	1.700
SAL. REELS(B)	3.900	3.800	1.600	.900	.600	.800	.400
TAUX UTIL. CAP. PROD.	.809	.777	.755	.735	.713	.694	.679

(A) En milliers de personnes

(B) En taux de croissance

4. Compte de l'Etat

DEPENSES

	1979	1980	1981	1982	1983	1984	1985
CONS. PUBLIQUE	559.634	647.370	676.222	723.905	787.334	841.945	889.563
INV. PUBLICS	181.387	101.577	133.772	123.428	156.186	226.507	109.025
TRANSF. ETAT-MEN.	603.336	667.742	702.839	754.411	817.544	874.180	932.011
TOTAL	1344.357	1416.689	1512.833	1601.744	1761.064	1942.632	1930.599

RECETTES

	1979	1980	1981	1982	1983	1984	1985
IMPOTS DIRECTS	491.048	559.349	621.272	679.394	792.299	919.274	890.034
IMPOTS IND.	369.513	346.198	364.756	376.279	374.086	375.263	410.872
COTIS. SEC. SOC.	379.355	424.091	457.201	493.792	559.837	630.692	629.639
DEFIC. PUBLIC	104.431	87.050	69.596	52.275	34.832	17.398	.019
TOTAL	1344.347	1416.688	1512.825	1601.740	1761.054	1942.627	1930.564

(en milliards de francs)

5. *INSTRUMENTS DE POLITIQUE ECONOMIQUE*

	1979	1980	1981	1982	1983	1984	1985
INVEST. PUBL.(B)	85.494	42.959	53.004	45.390	52.960	71.281	31.894
SAL.REELS(A)	3.900	3.800	1.600	.900	.600	.800	.400
TAUX COT. SEC. SOC.	.185	.189	.196	.199	.209	.220	.206
TAUX TAX. DIR.	.205	.211	.223	.227	.244	.265	.241
TAUX TAX. INDIR.	.207	.175	.179	.174	.164	.158	.160

(A) en taux de croissance

(B) en milliards de frands 1970.

3. SIMULATION OU L'OBJECTIF , EXPORTATIONS NETTES, EST PRIVILEGIE

Le déficit de la balance des paiements est minimisé par une diminution du volume des importations (tableau 2). Pour cela il faut diminuer les investissements publics (voir tableau 1.1). De même, la consommation privée diminue (voir tableau 1.1), suite à la diminution des revenus, due à l'augmentation des impôts. Cette augmentation des impôts réduit le déficit de l'Etat, qui devient finalement en boni.

1. Le produit intérieur brut et ses composantes

1.1 *CROISSANCE EN PRIX DE* 1970

	1979	1980	1981	1982	1983	1984	1985
P.I.B	4.9	2.4	1.8	2.9	2.7	2.4	3.9
CONS. PR.	.2	.8	.3	.9	.5	.1	1.7
CONS. PUB.	9.6	.4	.0	.0	.0	.0	.0
INV. PRIVES	9.0	5.7	4.7	5.1	4.9	4.7	5.4
INV. PUB.	2.4	4.9	4.2	.9	2.6	4.3	9.5
EXPORTATIONS	6.6	5.0	3.7	5.6	5.6	5.7	5.7
IMPORTATIONS	2.9	2.9	2.6	4.4	4.2	4.2	4.3

1.2 *CROISSANCE DES PRIX*

	1979	1980	1981	1982	1983	1984	1985
PRIX P.I.B	7.0	8.6	6.2	5.8	5.8	6.7	1.5
PRIX CONS. PR.	8.1	9.0	6.2	6.0	6.1	7.4	1.5
PRIX CONS. PUB.	9.7	10.9	7.6	7.3	7.4	8.9	1.9
PRIX INV. PRIVES	9.6	10.2	7.6	7.1	7.2	7.9	4.8
PRIX INV. PUB.	10.9	11.2	8.7	8.3	8.4	9.0	6.5
PRIX EXPORTATIONS	11.3	11.3	7.9	7.8	7.6	7.4	7.4
PRIX IMPORTATIONS	14.5	13.0	8.7	8.7	8.7	8.7	8.7

2. Commerce extérieur et balance des paiements

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
VOL. COMM. MOND.(A)	6.20	4.00	3.00	5.00	5.00	5.00	5.00
PRIX COMM. MOND.(A)	15.20	14.00	10.00	10.00	10.00	10.00	10.00
TAUX CHA. (FB/US DOL)(B)	30.78	30.16	29.56	28.97	28.39	27.82	27.27

(A) Taux de croissance

(B) Chiffres en moyenne annuelle

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
PRIX COMM. MOND FB (A)	13.100	11.700	7.800	7.800	7.800	7.800	7.800
PRIX IMPORT.(A)	14.500	13.000	8.700	8.700	8.700	8.700	8.700
PRIX EXPORT. (A)	11.300	11.300	7.900	7.800	7.600	7.400	7.400
TERMES D'ECH.(A)	3.200	1.700	.800	.900	1.100	1.300	1.300
VOL. EXPORT. (A)	6.600	5.000	3.700	5.600	5.600	5.700	5.700
VOL. IMPORT. (A)	2.900	2.900	2.600	4.400	4.200	4.200	4.300
EXPORT. NET. FB (B)	34.913	29.089	23.275	17.456	11.640	5.823	.004

(A) Taux de croissance

(B) En milliards de francs

3. Production, emploi et salaires réels

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
POP.ACTIVE	4102.100	4117.800	4144.700	4173.600	4200.100	4230.300	4253.600
EMPL. SEC. PUB.	626.520	628.000	628.000	628.000	628.000	628.000	628.000

(en milliers de personnes)

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
EMPL. SECT. PR.(A)	3134.600	3113.826	3090.780	3073.162	3057.277	3038.374	3032.167
CHOMAGE (A)	340.982	375.974	425.920	472.435	514.866	563.895	593.453
PRODUCTIV.(B)	4.600	2.900	2.500	3.400	3.200	2.900	4.100
SAL. REELS(B)	2.900	2.900	2.800	2.900	2.900	2.900	2.900
TAUX UTIL. CAP. PROD.	.802	.801	.796	.797	.797	.794	.800

(A) en milliers de personnes

(B) en taux de croissance

4. Compte de l'Etat

DEPENSES

	1979	1980	1981	1982	1983	1984	1985
CONS. PUBLIQUE	657.319	731.958	787.297	844.468	907.131	988.104	1007.245
INV. PUBLICS	120.673	127.656	132.887	142.547	150.574	157.055	151.266
TRANSF. ETAT-MEN.	694.414	773.350	840.798	912.396	988.814	1079.383	1129.621
TOTAL	1472.406	1632.964	1760.982	1899.411	2046.519	2224.542	2288.132

RECETTES

	1979	1980	1981	1982	1983	1984	1985
IMPOTS DIRECTS	574.052	700.735	811.519	901.985	1017.294	1157.076	1225.742
IMPOTS IND.	325.146	336.493	358.269	381.326	404.772	416.711	489.953
COTIS. SEC. SOC.	444.368	521.335	586.063	641.981	712.354	798.748	840.519
DEFIC. PUBLIC	128.840	74.400	5.131	-25.879	-87.899	-147.992	-268.109
TOTAL	1472.406	1632.964	1760.982	1899.413	2046.521	2224.543	2288.105

(en milliards de francs)

5. *INSTRUMENTS DE POLITIQUE ECONOMIQUE*

	1979	1980	1981	1982	1983	1984	1985
INVEST. PUBL.(B)	52.983	50.387	48.248	47.791	46.563	44.568	40.320
SAL.REELS(A)	2.900	2.900	2.800	2.900	2.900	2.900	2.900
TAUX COT. SEC. SOC.	.187	.198	.206	.210	.215	.221	.223
TAUX TAX. DIR.	.208	.228	.243	.249	.259	.270	.273
TAUX TAX. INDIR.	.159	.153	.153	.153	.152	.146	.166

(A) en taux de croissance

(B) en milliards de francs 1970

4. SIMULATION OU L'OBJECTIF, CHOMAGE EST PRIVILEGIE

Nous constatons une diminution de l'accroissement des salaires réels

1. Le produit intérieur brut et ses composantes

1.1 *CROISSANCE EN PRIX DE* 1970

	1979	1980	1981	1982	1983	1984	1985
P.I.B	12.4	3.2	4.4	4.0	3.9	4.9	4.3
CONS. PR.	13.2	2.4	5.0	3.8	3.3	4.4	2.8
CONS. PUB.	9.6	.4	.0	.0	.0	.0	.0
INV. PRIVES	12.2	-4.8	5.2	4.8	-4.8	5.3	4.8
INV. PUB.	21.7	-7.9	4.8	.2	-.0	3.9	3.9
EXPORTATIONS	6.6	5.0	4.0	5.6	5.7	5.7	5.8
IMPORTATIONS	9.5	5.5	5.2	5.9	5.6	6.1	5.7

1.2 *CROISSANCE DES PRIX*

	1979	1980	1981	1982	1983	1984	1985
PRIX P.I.B	-5.9	10.2	1.7	3.8	3.6	.5	1.4
PRIX CONS. PR.	-5.4	11.7	1.9	4.5	4.4	1.4	2.5
PRIX CONS. PUB.	-6.2	14.0	2.4	5.6	5.4	1.8	3.1
PRIX INV. PRIVES	.5	10.7	4.5	6.0	6.1	4.6	5.5
PRIX INV. PUB.	3.6	11.7	6.2	7.4	7.5	6.3	7.0
PRIX EXPORTATIONS	11.3	10.0	7.9	7.3	7.3	7.1	6.8
PRIX IMPORTATIONS	14.5	13.0	8.7	8.7	8.7	8.7	8.7

2. Commerce extérieur et balance des paiements

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
VOL. COMM. MOND.(A)	6.20	4.00	3.00	5.00	5.00	5.00	5.00
PRIX COMM. MOND.(A)	15.20	14.00	10.00	10.00	10.00	10.00	10.00
TAUX CHA. (FB/US DOL)(B)	30.78	30.16	29.56	28.97	28.39	27.82	27.27

(A) en taux de croissance

(B) chiffres en moyenne annuelle

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
PRIX COMM. MOND FB (A)	13.100	11.700	7.800	7.800	7.800	7.800	7.800
PRIX IMPORT.(A)	14.500	13.000	8.700	8.700	8.700	8.700	8.700
PRIX EXPORT. (A)	11.300	10.000	7.900	7.300	7.300	7.100	6.800
TERMES D'ECH.(A)	-3.200	-3.000	-.800	-1.400	-1.400	-1.600	-1.900
VOL. EXPORT. (A)	6.600	5.000	4.000	5.600	5.700	5.700	5.800
VOL. IMPORT. (A)	9.500	5.500	5.200	5.900	5.600	6.100	5.700
EXPORT. NET. FB (B)	-155.913	-254.498	-334.079	-428.401	-525.288	-668.083	-832.610

(A) en taux de croissance

(B) en milliards de francs

3. Production, emploi et salaires réels

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
POP.ACTIVE	4102.100	4117.800	4144.700	4173.600	4200.100	4230.300	4253.600
EMPL. SEC. PUB.	626.520	628.000	628.000	628.000	628.000	628.000	628.000

(en milliers de personnes)

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
EMPL. SECT. PR.(A)	3211.303	3261.218	3323.864	3388.448	3450.715	3516.558	3575.637
CHOMAGE (A)	264.279	228.581	192.836	157.149	121.428	85.711	49.983
PRODUCTIV.(B)	9.800	1.800	2.700	2.400	2.300	3.200	2.800
SAL. REELS(B)	-.200	.200	.400	.800	1.200	1.600	2.200
TAUX UTIL. CAP. PROD.	.836	.820	.810	.798	.785	.779	.770

(A) en milliers de personnes

(B) en taux de croissance

4. Compte de l'Etat

DEPENSES

	1979	1980	1981	1982	1983	1984	1985
CONS. PUBLIQUE	561.838	643.222	658.813	695.446	732.807	746.224	769.708
INV. PUBLICS	140.532	144.439	160.768	173.120	186.111	205.613	228.729
TRANSF. ETAT-MEN.	627.297	712.258	753.442	812.621	874.134	921.030	980.107
TOTAL	1329.667	1499.919	1573.023	1681.187	1793.052	1872.867	1978.544

RECETTES

	1979	1980	1981	1982	1983	1984	1985
IMPOTS DIRECTS	278.314	377.792	350.938	373.485	396.065	389.303	454.596
IMPOTS IND.	420.680	430.265	496.444	547.200	604.996	694.834	779.272
COTIS. SEC. SOC.	278.818	345.415	339.181	362.114	386.369	393.871	438.962
DEFIC. PUBLIC	351.853	346.450	386.460	398.388	405.620	394.861	305.683
TOTAL	1329.665	1499.922	1573.023	1681.187	1793.050	1872.869	1978.513

(en milliards de francs)

5. *INSTRUMENTS DE POLITIQUE ECONOMIQUE*

	1979	1980	1981	1982	1983	1984	1985
INVEST. PUBL.(B)	66.032	60.778	63.677	63.835	63.818	66.332	68.949
SAL.REELS(A)	.200	.200	.400	.800	1.200	1.600	2.200
TAUX COT. SEC. SOC.	.130	.143	.134	.134	.133	.128	.134
TAUX TAX. DIR.	.112	.134	.117	.116	.114	.106	.117
TAUX TAX. INDIR.	.208	.186	.201	.204	.209	.227	.241

(A) en taux de croissance

(B) en milliards de francs 1970.

5. SIMULATION OU L'OBJECTIF, DEFICIT DE L'ETAT, EST PRIVILEGIE

1. Le produit intérieur brut et ses composantes

1.1 *CROISSANCE EN PRIX DE* 1970

	1979	1980	1981	1982	1983	1984	1985
P.I.B	5.6	3.8	2.8	3.3	3.2	3.2	3.1
CONS. PR.	.6	1.9	2.4	2.5	2.3	2.1	2.0
CONS. PUB.	9.6	.4	.0	.0	.0	.0	.0
INV. PRIVES	9.3	6.1	4.9	5.0	4.9	4.9	4.8
INV. PUB.	3.5	1.2	.7	.5	.4	.3	.2
EXPORTATIONS	6.5	5.0	3.7	5.6	5.6	5.7	5.7
IMPORTATIONS	3.0	4.5	4.2	5.2	5.1	5.1	5.1

1.2 *CROISSANCE DES PRIX*

	1979	1980	1981	1982	1983	1984	1985
PRIX P.I.B	4.6	6.4	5.9	5.7	5.6	5.5	5.5
PRIX CONS. PR.	5.4	6.8	6.0	5.9	6.0	6.1	6.1
PRIX CONS. PUB.	6.5	8.2	7.3	7.2	7.3	7.4	7.5
PRIX INV. PRIVES	8.2	8.8	7.4	7.1	7.2	7.2	7.3
PRIX INV. PUB.	9.8	10.1	8.5	8.3	8.4	8.4	8.4
PRIX EXPORTATIONS	11.3	11.1	7.7	7.8	7.6	7.4	7.3
PRIX IMPORTATIONS	14.5	13.0	8.7	8.7	8.7	8.7	8.7

2. Commerce extérieur et balance des paiements

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
VOL. COMM. MOND.(A)	6.20	4.00	3.00	5.00	5.00	5.00	5.00
PRIX COMM. MOND.(A)	15.20	14.00	10.00	10.00	10.00	10.00	10.00
TAUX CHA. (FB/US DOL)(B)	30.78	30.16	29.56	28.97	28.39	27.82	27.27

(A) en taux de croissance

(B) chiffres en moyenne annuelle

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
PRIX COMM. MOND FB (A)	13.100	11.700	7.800	7.800	7.800	7.800	7.800
PRIX IMPORT.(A)	14.500	13.000	8.700	8.700	8.700	8.700	8.700
PRIX EXPORT. (A)	11.300	11.100	7.700	7.800	7.600	7.400	7.300
TERMES D'ECH.(A)	3.200	1.900	1.000	.900	1.100	1.300	1.400
VOL. EXPORT. (A)	6.500	5.000	3.700	5.600	5.600	5.700	5.700
VOL. IMPORT. (A)	3.000	4.500	4.200	5.200	5.100	5.100	5.100
EXPORT. NET. FB (B)	36.838	67.973	110.005	136.383	173.911	222.718	283.628

(A) en taux de croissance

(B) en milliards de francs

3. Production, emploi et salaires réels

EXOGENES

	1979	1980	1981	1982	1983	1984	1985
POP.ACTIVE	4102.100	4117.800	4144.700	4173.600	4200.100	4230.300	4253.600
EMPL. SEC. PUB.	626.520	628.000	628.000	628.000	628.000	628.000	628.000

(en milliers de personnes)

ENDOGENES

	1979	1980	1981	1982	1983	1984	1985
EMPL. SECT. PR.(A)	3140.170	3133.446	3126.221	3122.226	3118.883	3144.791	3109.207
CHOMAGE (A)	335.412	356.354	390.479	423.372	453.261	487.478	516.413
PRODUCTIV.(B)	5.200	3.900	3.000	3.400	3.300	3.300	3.300
SAL. REELS(B)	3.100	3.100	3.100	3.000	3.000	3.000	3.000
TAUX UTIL. CAP. PROD.	.807	.812	.809	.810	.809	.808	.808

(A) en milliers de personnes

(B) en taux de croissance

4. Compte de l'Etat

DEPENSES

	1979	1980	1981	1982	1983	1984	1985
CONS. PUBLIQUE	638.201	693.694	744.649	798.206	856.739	920.181	989.273
INV. PUBLICS	118.199	128.737	138.793	149.552	161.383	174.370	188.754
TRANSF. ETAT-MEN.	682.726	752.519	822.692	897.284	976.785	1061.808	1153.877
TOTAL	1439.126	1574.950	1706.134	1845.042	1994.907	2156.359	2331.904

RECETTES

	1979	1980	1981	1982	1983	1984	1985
IMPOTS DIRECTS	557.218	628.729	697.140	768.803	844.836	926.058	1014.641
IMPOTS IND.	344.305	378.374	413.037	449.015	486.988	526.692	567.673
COTIS. SEC. SOC.	433.162	480.782	526.322	574.992	628.256	686.190	749.578
DEFIC. PUBLIC	104.442	87.037	69.635	52.231	34.829	17.420	.020
TOTAL	1439.127	1574.922	1706.134	1845.041	1994.909	2156.360	2331.872

(en milliards de francs)

5. *INSTRUMENTS DE POLITIQUE ECONOMIQUE*

	1979	1980	1981	1982	1983	1984	1985
INVEST. PUBL.(B)	52.406	51.821	51.477	51.206	50.976	50.799	50.706
SAL.REELS(A)	3.100	3.100	3.100	3.000	3.000	3.000	3.000
TAUX COT. SEC. SOC.	.185	.188	.189	.191	.192	.193	.194
TAUX TAX. DIR.	.206	.210	.213	.216	.218	.220	.222
TAUX TAX. INDIR.	.172	.173	.174	.175	.175	.174	.174

(A) en taux de croissance

(B) en milliards de francs 1970

6. TABLEAUX RECAPITULATIFS

OBJECTIFS

	SIMUL. REFERENCE	SIMUL. OBJ. 1:EXPORT. NET.	SIMUL. OBJ. 2: CHOMAGE
EXPORT. NETTES (B)	-400.053	-.004	-832.611
CHOMAGE (A)	492.946	593.453	49.983
DEFICIT ETAT (B)	229.927	-268.109	305.683

OBJECTIFS (SUITE)

	SIMUL. OBJ. 3 : DEF. ETAT	SIMULATION : 3 OBJ.
EXPORT. NETTES (B)	-283.628	.060
CHOMAGE (A)	516.413	49.948
DEFICIT ETAT (B)	-.020	.019

INSTRUMENTS

	SIMUL. REFERENCE	SIMUL. OBJ. 1:EXPORT. NET.	SIMUL. OBJ. 2: CHOMAGE
INVEST. PUBLICS (B)	54.279	40.320	68.949
SALAIRES REELS (*)	3.000	2.900	.900
TAUX COTIS. SEC. SOC. (B)	.176	.223	.134
TAUX TAX. DIRECTE (B)	.190	.273	.117
TAUX TAX. INDIR.(B)	.164	.166	.241

INSTRUMENTS (SUITE)

	SIMUL. OBJ. 3 : DEF. ETAT	SIMULATION : 3 OBJ.
INVEST. PUBLICS (B)	50.706	31.894
SALAIRES REELS (*)	3.000	1.600
TAUX COTIS. SEC. SOC. (B)	.194	.206
TAUX TAX. DIRECTE (B)	.222	.241
TAUX TAX. INDIR.(B)	.174	.160

(*) Taux de croissance annuel moyen sur 7 ans

(A) En milliers de personnes, en fin de période (1985)

(B) En milliards de francs 1970, en fin de période (1985).

AUTRES VARIABLES (*)

	SIMUL. REFERENCE	SIMUL. OBJ. 1:EXPORT. NET.	SIMUL. OBJ. 2: CHOMAGE
CROISS. P.I.B.	3.7	3.0	5.2
CROISS PRIX CONS.	6.3	6.3	2.9
CROISS. EXPORT.	5.4	5.4	5.5
CROISS. IMPORT.	5.0	3.6	6.2
TERMES D'ECHANGES	2.0	1.4	1.9
PRODUCTIVITE	3.7	3.3	3.5

AUTRES VARIABLES (*) (SUITE)

	SIMUL. OBJ. 3 : DEF. ETAT	SIMULATION : 3 OBJ.
CROISS. P.I.B.	3.6	3.2
CROISS PRIX CONS.	6.1	4.7
CROISS. EXPORT.	5.4	5.5
CROISS. IMPORT.	4.6	3.3
TERMES D'ECHANGES	1.5	1.9
PRODUCTIVITE	3.6	1.6

(*) Taux de croissance annuel moyen, sur 7 ans.

REMARQUE

Dans la théorie de Tinbergen, les valeurs des instruments sont obtenues par la résolution du système linéaire suivant :

$$\Delta\, y = E\, z + \emptyset\, u$$

où

- . y est un vecteur de dimension n des valeurs des variables endogènes (objectifs)
- . z est un vecteur de dimension n' des instruments
- . u est un vecteur de dimension k des valeurs des variables exogènes données.
- . Δ, E, $\emptyset$ sont des matrices respectivement (n,n), (n,n'), (n,k).

Pour trouver z, Tinbergen se donne une valeur désirée des objectifs : y°. Il trouve z° par la résolution du système ci-dessus.

1°) Si n=n' (autant d'instruments que d'objectifs) et E de rang n'

$z^\circ = E^{-1} \Delta\, y^\circ - E^{-1} \emptyset\, u$ est l'unique solution du système. Elle est trivialement optimale.

2°) Si n > n' (plus d'objectifs)

On ajoute des instruments ou élimine des objectifs.

3°) Si n < n' (plus d'instruments).

Nous avons un système contenant plus d'inconnues (n') que d'équations (n). Nous fixons arbitraitement (n'-n) variables. Nous obtenons alors un système de n équations à n inconnues dont la solution est unique.

De multiples essais du choix des variables arbitrairement fixées, permettent d'analyser les réponses du modèle et de choisir celle qui paraît la meilleure.

Notre approche permet d'éviter cette question du choix des valeurs des instruments à fixer, lorsqu'ils sont en nombre plus élevé que les objectifs.
Les objectifs privilégiés sont introduits dans le critère. La résolution du problème de minimisation fournit une seule valeur optimale pour les instruments. Si la solution n'était pas unique, l'algorithme de minimisation donne celle qui est la plus proche de l'approximation initiale. Une seule simulation du modèle suffit alors pour trouver les variables endogènes.

ANNEXE 1

Dans cette annexe, nous fournissons :

1°- les équations du modèle SANDY

2°- le programme APL de la mise du modèle sous forme quasi-triangulaire.

1°) EQUATIONS DE SANDY

AESCG CG[I]←(¯200.5839469+0.7641975808×LG[I])

AESPM PM[I]←(PM[I-1]×¯0.1841718806+1.183164917×(PWX[I]×ER[I])÷PWX[I-1]×ER[I-1])

AISOP OP[I]←*(1.746686952×(0.2396438757×⊛KAP[I])+(1-0.2396438757)×⊛LAB[I])+(0.015×TIME[I])-7.794484847

AISDUC DUC[I]←GDP[I]÷OP[I]

AESINVF INVF[I]←(*(⊛KAP[I-1])+((1-0.3376681823)×⊛INVF[I-1]÷KAP[I-2])+(0.3376681823×⊛GDP[I]÷GDP[I-1])+(0.02724056311×(⊛WR[I-1]÷CK[I-1])-⊛WR[I-2]÷CK[I-2])+¯0.7582758908)

AISKAP KAP[I]←INVF[I]+(1-DEP[I])×KAP[I-1]

AESLP LP[I]←(*(0.1891205365×⊛GDP[I])+(0.07103032387×⊛DUC[I])+(¯0.1828175529×⊛WR[I]÷PGDP[I])+(0.7454803947×⊛LP[I-1])+1.70678507)

AISLAB LAB[I]←LP[I]+LG[I]

AISUN UN[I]←LF[I]-LAB[I]

AESWR WR[I]←(*(⊛PC[I])+((1-0.3386077875)×⊛WR[I-1]÷PC[I-1])+(0.3386077875×⊛PGDP[I]÷PC[I])+(1.156316102×0.3386077875×⊛GDP[I]÷LAB[I])+2.27808562)

AESPC PC[I]←(*¯2.50686469+(0.431123012×⊛WR[I]÷GDP[I]÷LAB[I])+(0.1618176685×⊛PM[I])+(0.1588761556×⊛DUC[I])+(0.1166723335×⊛TINDR[I])+0.3247361521×⊛PC[I-1])

AESPG PG[I]←(PG[I-1]×¯0.1763929836+1.179475383×PC[I]÷PC[I-1])

AESPI PI[I]←(PI[I-1]×0.2011633541+(0.5448439176×(WR[I]÷GDP[I]÷LAB[I])÷WR[I-1]÷GDP[I-1]÷LAB[I-1])+0.2661231938×PM[I]÷PM[I-1])

AESPIG PIG[I]←(PIG[I-1]×0.2847029772+(0.4432966136×(WR[I]÷GDP[I]÷LAB[I])÷WR[I-1]÷GDP[I-1]÷LAB[I-1])+0.2997124134×PM[I]÷PM[I-1])

AISICX ICX[I]←(PM[I]*.4)×((WR[I]÷211.2982958)*.3)×(CK[I]÷.1552545299)*.3

AESPX PX[I]←(PX[I-1]×¯0.3758851398+(0.9134341717×(PWX[I]×ER[I])÷PWX[I-1]×ER[I-1])+(0.1443039534×ICX[I-1]÷ICX[I-2])+0.2954106721×WT[I]÷WT[I-1])

AESXPORTS XPORTS[I]←(XPORTS[I-1]×0.2798466271+(1.013288742×WT[I]÷WT[I-1])+¯0.2887403529×(PX[I-1]÷PWX[I-1]×ER[I-1]×49.656)÷PX[I-2]÷PWX[I-2]×ER[I-2]×49.656)

```
AISLW      LW[I]←LAB[I]-LSE[I]

AISLSE     LSE[I]←(LSE[I]×79≥TIME[I])+(79<TIME[I])×LSE[I-1]-1.5

AISYW      YW[I]←.001×WR[I]×LW[I]

AESWSE     WSE[I]←(*¯3.282333906+(0.1804684116×⍟WR[I])+(0.832128249×⍟GDP[I])+0.3724558075×⍟WSE[I-1])

AISYSE     YSE[I]←.001×WSE[I]×LSE[I]

AESYK      YK[I]←(YK[I-1]×¯0.3006205386+(0.4119858359×GDP[I]÷GDP[I-1])+(0.916728737×PGDP[I]÷PGDP[I-1])+2.336102499×RR[I]-
       RR[I-1])

AISYDISP   YDISP[I]←(YW[I]+YSE[I]+YK[I])+((YSE[I]+YW[I]+YK[I])×TFTSR[I])+TFRM[I]-((YW[I]+YSE[I])×COTSSR[I])+(YW[I]+YSE[I]
       +YK[I])×DTAXR[I]

AESVARS    VARS[I]←(¯231.7847881+(129.4658287×GDP[I]÷GDP[I-1])+106.4474042×(PM[I]÷PC[I])÷PM[I-1]÷PC[I-1])

AESCP      CP[I]←(43.96162722+(0.5310432033×YDISP[I]÷PC[I])+0.3290852921×CP[I-1])

AISAD      AD[I]←CP[I]+CG[I]+INVF[I]+IG[I]+VARS[I]+XPORTS[I]

AISPID     PID[I]←((CP[I]×PC[I])+(CG[I]×PG[I])+((INVF[I]+VARS[I])×PI[I])+IG[I]×PIG[I])÷CP[I]+CGI+INF[I]I[I]VASI

AESMPORTS  MPORTS[I]←(*¯4.61227391+(1.4121×⍟AD[I])+¯0.1939899872×⍟×PM[I]÷PID[I])

AISGDP     GDP[I]←CP[I]+CG[I]+INVF[I]+VARS[I]+IG[I]+XPORTS[I]-MPORTSI

AISPGDP    PGDP[I]←((CP[I]×PC[I])+(CG[I]×PG[I])+((INVF[I]+VARS[I])×PI[I])+(IG[I]×PIG[I])+(XPORTS[I]×PX[I])-MPORTS[I]×PM[I
       ])÷GDP[I]



AESRR      RR[I]←((¯0.1231674088×(M1[I]÷GDP[I]×PGDP[I])-M1[I-1]÷GDP[I-1]×PGDP[I-1])+(0.5342410013×REX[I])+0.6567043735×RR
       [I-1])

AISCK      CK[I]←PI[I]×DEP[I]+RR[I]
```

APL	==>	FORTRAN
+	::=	+
-	::=	-
¯	::=	-
×	::=	*
÷	::=	/
⍟	::=	LOG
*	::=	EXP

2°) PROGRAMME APL DE LA MISE DU MODELE SOUS FORME QUASI-TRIANGULAIRE

```
      ∇RAVELLI[[]]∇
    ∇ RAVELLI E;NB;VV;EP1;A;VBA;P;S;VB;A
[1]   VVB←EP←CO←⍳0
[2]   PR←PROLOG E
[3]   →0×⍳0=NB←⍴VV←(~(⍳NB)∈PR)/⍳NB←1↑⍴E
[4]   E←E[VV;VV]
[5]   EP←VV[EP1←⌽PROLOG⌽E]
[6]   VV←(~VV∈VV[EP1])/VV
[7]   NB←1↑⍴E←E[A;A←(~(⍳NB)∈EP1)/⍳NB←1↑⍴E]
[8]   VBA←0⍴J←2
[9]  L0:→L0×⍳(J←J+1)≤NB,0⍴VBA←VBA,(∨/E[⍳J-1;J])/J
[10]  A←(~(⍳NB)∈VBA)/⍳NB
[11]  E←E[A,VBA;A,VBA]
[12]  ⎕←'VV ==> ',▼VV←VV[A,VBA],0⍴NVB←⍴VBA,0⍴⎕←'VBA ==> ',▼VV[VBA]
[13] L1:P←P[⍋P],0⍴P←PRE E[⍳VB;⍳VB←''⍴NB+1-NVB]
[14]  S←E[⍳VB;VB]/⍳VB
[15]  →L2×⍳0=(⍴S)×⍴P
[16]  →L3×⍳∨/P∈S
[17] L2:A←A,P,VB,S[⍋S←S,(~(⍳NB)∈(A←(0≠⍴P)/⍳¯1+L/P,S,NB),P,VB,S)/⍳NB]
[18]  →L4,VV←VV[A],0⍴E←E[A;A]
[19] L3:E←A/(A←VB≠⍳NB)≠E
[20]  VVB←VVB,VV[VB]
[21]  NB←⍴VV←A/VV
[22]  VBA←1+VBA-1
[23] L4:→L1×⍳0≠NVB←NVB-1
[24]  CO←VV
    ∇
```

```
      ∇PROLOG[[]]∇
    ∇ PR←PROLOG E;NB;NL
[1]   PR←⍳0,0⍴NB←⍳1↑⍴E
[2]  L1:→0×⍳∧/0≠NL←+/E
[3]   PR←PR,(NL=0)/NB
[4]   NB←(NL≠0)/NB
[5]   E←(NL≠0)/E
[6]   →L1×0≠1↑⍴E←(NL≠0)≠E
    ∇
```

```
      ∇PRE[[]]∇
    ∇ P←PRE E;J;S;S1;N;T
[1]   P←0⍴S←E[N;]/⍳N←''⍴1↑⍴E
[2]  L1:→0×⍳0=T←⍴S,0⍴J←1
[3]  L2:→L3×⍳0=⍴S1←(S[J]↑E[S[J];])/⍳S[J]
[4]   S←S,(~S1∈P,S)/S1
[5]  L3:→L2×⍳(J←J+1)≤T
[6]   →L1,S←(~S∈P)/S←T↑S,0⍴P←(T↑S),P
    ∇
```

ANNEXE 2

Nous fournissons une copie des programmes utilisés pour la résolution du modèle.

1°) RESOMO : Les arguments sont notés de la même façon qu'au chapitre 3.

```
      SUBROUTINE RESOMO (T,L,M,S,N,R,U)

      DECLARATIONS

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      INTEGER T,S,R
      DIMENSION U(35),YB(4),E(4),A(4,4),W(52)
      COMMON/CLFN/Y(7,37),Z(9,10),D(84),K
      COMMON/NS01BD/LP,LPD,IERR
      COMMON/CLFN2/L1,M1
      COMMON/GRD/D1(84)
      M1=M
      L1=L

      INITIALISATION DE D

      I1=(2*N)+R
      DO 7 I=1,I1
      D(I)=D1(I)
      CONTINUE
      DO 8 I=1,R
      D((2*N)+R+I)=U(I)
      CONTINUE

      INITIALISATION DES VARIABLES DE BOUCLAGE (POUR NS01AD)

      DO 10 I=1,S
      YB(I)=D(N+M+I)
      CONTINUE

      SCALING SUR LES VARIABLES DE BOUCLAGE

      YB(2)=YB(2)/1000
      YB(4)=YB(4)/2000

      DO 15 K=1,T
      RESOLUTION DU PROLOGUE (RECURSIF)

      CALL RESOPA (1,L,YB)

      RESOLUTION DES EQUATIONS DE BOUCLAGE ET DU COEUR

         1) INITIALISATION POUR NS01AD

            STEP = 1.D-7
            STPMAX = .500
            ACC = 4*1.D-14
            MAXFUN = 40
            IPRINT = 0
            LP = 3
            LPD = 3

         2) ANNULATION DE L'ERREUR

            CALL NS01AD (S,YB,E,A,STEP,STPMAX,ACC,MAXFUN,IPRINT,W)

      DEFINITION DE Y POUR LES EQUATIONS DE BOUCLAGE

      DO 20 I=1,S
      Y(K,M+I)=YB(I)
      CONTINUE
      Y(K,M+2)=Y(K,M+2)*1000
      Y(K,M+4)=Y(K,M+4)*2000
```

```
RESOLUTION DE L'EPILOGUE

CALL RESOPA (M+S+1,N,YB)
IF (K.EQ.T) RETURN

MISE A JOUR DE D

DO 30 I=1,N
D(I)=D(N+I)
D(N+I)=Y(K,I)
CONTINUE
DO 40 I=1,R
D((2*N)+I)=D((2*N)+R+I)
D((2*N)+R+I)=U((K*R)+I)
CONTINUE

CONTINUE

RETURN
END
```

2°) Arguments de la routine NS01AD

Subroutine NS01AD (N, X, F, A, STEP, STPMAX, ACC, MAXFUN, IPRINT, W)
Le programme qui appelle doit :

- établir N, X, STEP, STPMAX, ACC, MAXFUN, IPRINT.
- contenir dans ses "dimension statement" les déclarations des tableaux X, F, A, W.

N = nombre d'équations

X = vecteur de dimension N qui contient l'approximation initiale de la solution au moment de l'appel. Il est modifié à chaque itération et à la sortie normale de NS01AD, il contient la solution.

STEP = paramètre qui permet à la routine de calculer les dérivées partielles $\frac{\partial f_i}{\partial x_j}$ par différences finies.

$$\frac{\partial f_i}{\partial x_j} = (f_i(x_1 \dots x_{j-1}, x_j + \text{STEP}, \dots x_N) - f_i(x_1 \dots x_N))/\text{STEP}.$$

STPMAX = estimation généreuse de la distance entre l'approximation initiale et la solution.
Si NS01AD prédit des pas beaucoup plus grands que STPMAX, la routine effectue un "return" et envoie un message d'erreur. Il n'y a pas de solution au voisinage de l'approximation initiale.

ACC = précision demandée.
Le test d'arrêt est défini par :

$$(3.16) \qquad \sum_{i=1}^{N} [f_i(x_1 \dots x_N)]^2 \leq \text{ACC}$$

MAXFUN = nombre maximum d'appels à CALFUN autorisé. C'est une sécurité qui permet de sortir si la convergence n'a pas eu lieu en maxfun intérations

IPRINT = guide l'impression
Si IPRINT = 0, il n'y a pas d'impression par NS01AD.
Si IPRINT est non nul, les valeurs de $x_1 \ldots x_N$ sont imprimées toutes les / IPRINT / itérations. La solution finale est aussi imprimée.

NS01AD établit

. F : vecteur de dimension N prenant à chaque itération des valeurs des f_j pour le vecteur $x = (x_1 \ldots x_N)$

. A : matrice (N*N), estimation de l'inverse du jacobien $\left(\left(\frac{\partial f_i}{\partial x_j}\right)\right)_{i,j=1\ldots N}$

. W : espace de travail de dimension N(2N + 5)

NS01AD utilise le "common block", COMMON/NS01BD/LP,LPD,IERR.
LP et LPD désignent les numéros d'unité de sortie, pour les impressions faites par NS01AD.
IERR sert pour l'impression des messages d'erreur. Par exemple, IERR = 3 signifie que la solution n'a pas été obtenue en maxfun itérations.

La correspondance entre les paramètres s'établit comme suit :

NS01AD		RESOMO
. résout F(X) = 0		. résout E (YB) = 0
. X	est remplacé par	YB
. N	équivaut à	S (=4 pour SANDY)
. F	est remplacé par	E

Les autres paramètres gardent le même nom.
Nous déclarons dans RESOMO, les tableaux A(4,4), E(4), YB(4), W(52).

3°) CALFUN

Sous-routine utilisée par NS01AD pour définir la fonction à annuler. CALFUN fournit aussi la valeur des variables du coeur dans la matrice Y.

```
      SUBROUTINE CALFUN (S,YB,E)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      INTEGER S
      DIMENSION YB(4),E(4)
      COMMON/CLFN/Y(7,37),Z(9,10),D(84),K


      COMMON/CLFN2/L,M
      CALL RESOPA (L+1,M,YB)
      DO 10 I=1,S
      E(I) = YB(I)-F(K,M+I,Z,D,YB,Y)
   10 CONTINUE
      RETURN
      END
```

A N N E X E 3

Nous fournissons ici la copie des sous-programmes utilisés pour l'optimisation du modèle SANDY :

- FUNC (N4, U, C, G) utilisé par VA13AD
- PARSH (U, C, NA, G) qui calcule le gradient du critère au point U
- CNM (T, Y, YD) et CT (K, Y, YD) qui calculent le critère.

```
C
      SUBROUTINE PARSH (U,C,NA,G)
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      INTEGER T,S,R
      DIMENSION U(35),G(35)
      COMMON/RSM/T,L,M,S,N,R,YD(7,3)
      COMMON/CLEN/Y(7,37),Z(9,10),D(84),K
      COMMON/GDR/D1(84)
C
      DO 200 I=1,NA
      S1=U(I)
      DX=DABS(S1)*1.0D-5
      IF (DX.LT.1.0D-5) DX=1.0D-5
      U(I)=U(I)+DX
C
      CALL RESOMO (T,L,M,S,N,R,U)
      C1=CNM (T,Y,YD)
      G(I)=(C1-C)/DX
      U(I)=S1
200   CONTINUE
C
      RETURN
      END



C
      SUBROUTINE FUNC (N4,U,C,G)
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      INTEGER T,S,R
      DIMENSION G(35),U(35)
      COMMON/RSM/T,L,M,S,N,R,YD(7,3)
      COMMON/CLEN/Y(7,37),Z(9,10),D(84),K
      COMMON/GRD/D1(84)
C
      DO 20 I=1,R
      D((2*N)+R+I)=U(I)
20    CONTINUE
C
      CALL RESOMO (T,L,M,S,N,R,U)
      C = CNM (T,Y,YD)
C
      CALL PARSH (U,C,N4,G)
C
      RETURN
      END
C
C
C     DOUBLE PRECISION FUNCTION CT (K,Y,YD)
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION Y(7,37),YD(7,3)
C
      CT = (Y(K,37)-YD(K,2))**2
C
      RETURN
      END
C
C
C     DOUBLE PRECISION FUNCTION CNM (T,Y,YD)
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      INTEGER T
      DIMENSION Y(7,37),YD(7,3)
C
      S=0.
      DO 10 K=1,T
      S=S+CT(K,Y,YD)
10    CONTINUE
      CNM=S/2
C
      RETURN
      END
C
```

R E F E R E N C E S

[1] D. GABAY; P. NEPOMIASTCHY; M. RACHDI; A. RAVELLI : "Méthodes mathématiques de la modélisation macroéconomique" : Institut de recherche d'informatique et d'automatique. Rocquencourt 1979.

[2] D. GABAY; P. NEPOMIASTCHY; M. RACHDI; A. RAVELLI : "Etude, résolution et optimisation de modèles macroéconomiques".- Rapport Laboria, N° 312, juin 1978.

[3] P. NEPOMIASTCHY; A. RAVELLI; F. RECHENMANN : "An automatic method to get an econometric model in a quasi-triangular form". - Rapport Laboria, N° 313, juin 1978.

[4] Bibliothèque HARWELL, A.E.R.E. - Report Didcot , Oxfordshire (GB).

[5] P. CALLIER : "Théorie des Graphes". - Cours donné en 2ème licence mathématique aux Facultés Notre-Dame de la Paix - Namur.

[6] M. RACHDI : "Quelques algorithmes de résolution de modèles macroéconomiques". Thèse de 3ème cycle de mathématique de la décision. Université Paris IX-Dauphine; 1979.

[7] J.M. ORTEGA; W.C. RHEINBOLDT : "Iterative Solution for Nonlinear Equations in Several Variables ". Academic Press, New York, 1970.

[8] M.R. HESTENES : "Optimization Theory. The Finite dimensional Case". Wiley New York, 1975.

[9] BAZARAA ; SHETTY : "Nonlinear Programming : Theory and Algorithms". Wiley, New York, 1979.

[10] A.I. COHEN : "Conjugate Gradient Algorithms - Siam J. Numerical Analysis, 9 1972 - pp. 248-259.

[11] D.V. STEWARD : "Partitioning and tearing systems of equations". - Siam J. On Numerical Analysis, ser. B, Vol. 2 - 1965.

[12] L.S. LASDON; A.D. WAREN : "Survey of Nonlinear Programming Applications" - Operations Research; vol. 28, N°5, September-October 1980.

[13] F. THYS-CLEMENT : "Une génération dynamique de la théorie de Tinbergen . Application à un modèle à moyen terme pour la Belgique". Thèse de doctorat en sciences économiques - U.L.B. 1974-1975.

[14] R.S. PINDYCK : "Optimal Policies for economic Stabilization". Econometrica, Vol. 41, N° 3, May 1973.

[15] R.S. PINDYCK : "Optimal Planning for economic Stabilization. North Holland Publishing, Co, Amsterdam, 1973.

[16] Y. MURATA : "Mathematics for Stability and Optimization of economic Systems". Academic Press, New York, San Francisco, London, 1977.

[17] T. de BIOLLEY : "SANDY". Version August-September 1980 - Bureau du Plan.

[18] A. HERREMANS; T. de BIOLLEY : "Teaching trough Simulation. Centre de Recherche en Simulation des Systèmes socio-économiques (U.C.L.) Working Paper 79-12-3.

[19] T. de BIOLLEY; A. HERREMANS; D. HIRSCHBERG : "Man-machine complementarity in interactive Teaching" - Informatica 79.