

L. LEBART

## **Programme d'agrégation avec contraintes**

*Les cahiers de l'analyse des données*, tome 3, n° 3 (1978),  
p. 275-287

[http://www.numdam.org/item?id=CAD\\_1978\\_\\_3\\_3\\_275\\_0](http://www.numdam.org/item?id=CAD_1978__3_3_275_0)

© Les cahiers de l'analyse des données, Dunod, 1978, tous droits réservés.

L'accès aux archives de la revue « Les cahiers de l'analyse des données » implique l'accord avec les conditions générales d'utilisation (<http://www.numdam.org/conditions>). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM

Article numérisé dans le cadre du programme  
Numérisation de documents anciens mathématiques

<http://www.numdam.org/>

## PROGRAMME D'AGRÉGATION AVEC CONTRAINTES

### [C.A.H. CONTIGUITÉ]

par L. Lebart (1)

#### 1 Finalité de la procédure

Le programme AGRAP doit fournir une aide dans l'établissement d'un zonage d'îlots : A l'intérieur d'une agglomération les îlots seront regroupés en classes connexes géographiquement, les regroupements étant effectués à partir des similitudes de caractéristiques socio-économiques

L'opération essentielle est une classification ascendante hiérarchique, avec contrainte de contiguïté géographique, et éventuellement contrainte sur les effectifs des classes.

Cette classification utilise certains intermédiaires de calcul qui peuvent être édités (en particulier une analyse des correspondances du tableau croisant les îlots et leurs caractéristiques socio-économiques). Elle nécessite une série de calculs annexes destinés à faciliter la compréhension et l'interprétation des résultats.

#### 2 Description des principales opérations

##### 2.1 La classification hiérarchique

Rappelons brièvement le principe des procédures classiques de classifications ascendantes hiérarchiques.

L'ensemble des  $n$  objets à classer est tout d'abord muni d'une mesure de distance (ou de similarité). Il est indispensable que toutes les distances soient calculées dès le début de la classification proprement dite (il s'agira ici de *distances socio-économiques*).

. La première étape consiste à chercher parmi les  $n$  objets quels sont les deux plus proches au sens de cette distance (donc à distance minimale ou à similarité maximale). Ces deux objets sont agrégés en un "nouvel objet".

Il est ensuite nécessaire de recalculer les distances entre le nouvel objet et tous les autres. (Les règles de recalcul varient selon la technique choisie).

. La seconde étape est alors analogue à la première, mais avec  $(n-1)$  objets. Le processus continue jusqu'à ce qu'il ne reste plus qu'un seul objet ; il nécessite  $(n-1)$  étapes.

On sait qu'il existe de nombreuses variantes de ces procédures selon le choix initial de la distance ou de l'indice de similarité, selon les formules de recalcul des distances après agrégation. L'une des méthodes les plus répandues dans la littérature statistique est la méthode du saut minimal, pour laquelle la distance entre un objet  $x$  quelconque et le nouvel objet  $t$  obtenu en agrégeant deux objets  $y$  et  $z$  est simplement la plus courte des deux distances  $d(x,y)$  et  $d(x,z)$ . (Cette formule de recalcul particulièrement simple conduit à une hiérarchie de  $p$ -types dite de l'ultramétrie inférieure maximale, étroitement liée à la recherche de l'arbre de longueur minimale). La méthode retenue ici est l'*Agrégation suivant la variance*. cf [C.A.H.] TI B n° 4 § 2.5.

---

(\*) Travail effectué dans le cadre d'une convention de recherche passée avec le Service des Programmes et Affaires Financières (D.0n des Télécommunications) - Groupement C1. Prévisions et Etudes Economiques. -

(1) Maître de recherches - C.N.R.S. -

## 2.2 La contrainte de contiguïté

La prise en compte d'une contrainte de contiguïté introduit quelques complications dans la procédure mais permet d'en accélérer l'exécution et de faire d'importantes économies de mémoire centrale.

L'algorithme fonctionnera comme précédemment, mais au lieu de chercher les deux objets les plus proches parmi tous les couples, il cherchera les deux objets les plus proches parmi tous les couples contigus, qui sont en nombre beaucoup plus restreint (1). Si l'on suppose que chacun des  $n$  îlots est en moyenne contigu à 8 îlots limitrophes, le nombre de couples d'îlots contigus est de l'ordre de  $4n$ . Pour 500 îlots, il y a ainsi environ 2000 couples contigus et près de 125.000 couples en tout.

Au lieu de tenir disponible en mémoire centrale l'ensemble des valeurs de toutes les distances, il suffira de se limiter aux distances entre objets contigus, ce qui permettra une grande économie de place-mémoire.

La relation de contiguïté est codée par une matrice donnant, pour chaque objet - (on dira dorénavant pour chaque *sommet* en adoptant la terminologie de la théorie des graphes) - les numéros des sommets contigus - (ou bien adjacents). Cette matrice doit être remise à jour après chaque agrégation, de même que la matrice des distances (dont les dimensions et le mode d'indexage sont identiques).

L'essentiel des difficultés réside dans ces procédures de remise à jour, qui seraient beaucoup plus aisées si la place mémoire nous permettait de travailler sur la matrice booléenne d'ordre  $(n \times n)$  associée au graphe de contiguïté.

## 2.3 La contrainte sur les effectifs des classes

L'existence de cette contrainte est une des options du programme. Elle permet de remédier aux déséquilibres de certaines classifications ascendantes hiérarchiques au cours desquelles une classe dominante "absorbe" tous les autres éléments un par un. Au-delà d'un certain effectif fixé par l'utilisateur du programme, la classe est considérée comme définitive. Elle est supprimée, et le graphe est remis à jour en conséquence - (on notera que cette opération peut détruire la connexité du graphe).

Cette option nous donne automatiquement une partition des îlots en classes connexes géographiquement. Il est cependant déconseillé de l'utiliser seule, car il est difficile d'apprécier la qualité de cette partition, sans référence à une classification sans contrainte sur les effectifs.

## 2.4 Mise en oeuvre des calculs

Un certain nombre d'opérations intermédiaires ou annexes sont également réalisées.

a) - Visualisation de la matrice associée au graphe - (option facultative).

Cette matrice booléenne est utilisée sans jamais figurer en mémoire centrale. Etant clairsemée (elle contient beaucoup de "0" et très peu de "1"), elle est stockée sous la forme de matrice de codage réduit (donnant les adresses des "1"). Un programme spécialement adapté de diagonalisation permet d'effectuer l'analyse des correspondances de la matrice booléenne à partir de la matrice de codage réduit. Des matrices "booléennes clairsemées" d'ordre  $(1000 \times 1000)$  peuvent ainsi être analysées en un temps très court (quelques secondes). La visualisation du graphe à partir des relations d'adjacence constitue une reconstitution grossière de la carte géographique des îlots. Sur cette carte approchée seront plus tard situés les différents "noeuds" de la hiérarchie.

(1) Le principe de cette méthode se trouve notamment dans la thèse de A. Thauront (Université Pierre et Marie Curie, 1975 - Paris)

b) - Analyse des correspondances du tableau croisant les flots et leurs caractéristiques socio-économiques - (obligatoire) -

Cette phase de calcul a deux fonctions :

- D'une part fournir une *carte socio-économique* des  $n$  flots, nous permettant de procéder à une analyse critique de la structure économique et démographique des flots sans prendre en compte d'information concernant leur position géographique.

- D'autre part de fournir de nouvelles coordonnées (sur les axes factoriels) pour les  $n$  flots, permettant un calcul économique des distances entre couples d'flots.

On retiendra en général 10 facteurs pour obtenir une approximation suffisante des distances. En mémoire centrale ne figurera que le tableau ( $n \times 10$ ) des coordonnées sur les axes, au lieu du tableau ( $n(n-1)/2$ ) des distances.

c) - Classification proprement dite :

Elle est effectuée suivant la procédure décrite en 2.2.

On calcule toutes les distances entre les flots *adjacents*.

Les deux flots adjacents les plus proches sont alors agrégés en un seul nouvel flot, ainsi que leurs coordonnées. La matrice associée au graphe est remise à jour, ainsi que les distances. Puis le processus continue par la recherche des deux flots les plus proches, ... etc.

La contrainte sur les effectifs des classes interviendra éventuellement comme indiquée en 2.3.

d) - Aides à l'interprétation :

Les paramètres et les représentations graphiques édités par le programme sont présentés et commentés à propos d'un exemple: cf [Zones Aix], ce cahier, pp 289-305.

### 3. Description générale du programme

#### 3.1 Fonctions de la version actuelle du programme :

(Les sous-programmes figurent entre parenthèses)

a) - Edition des principaux paramètres, du dictionnaire des codes en 3 et 4 caractères (3 caractères = numérotation séquentielle ; 4 caractères = numéros INSEE).

(sous-programme DONE5)

b) - Edition et apurements divers de la matrice de contiguïté.

b.1. Test de dépassement des numéros de sommet du graphe.

b.2. Test de symétrie de la relation binaire. (Les arêtes du graphe qui ne sont pas mentionnées deux fois sont détectées, et supprimées).

b.3. Classement, pour chaque sommet, des sommets adjacents par numéro croissant.

(sous-programme CONTI et SUPPR)

b.4. Détection et élimination éventuelles des boucles (sommets adjacents à eux-mêmes).

c) - Reconstitution graphique de l'ensemble des relations d'adjacence par analyse factorielle des correspondances de la matrice associée au graphe.

Cette diagonalisation se fait directement à partir de la matrice de codage réduit (ne contenant que des adresses de sommets adjacents) sans nécessiter le calcul de la véritable matrice associée, dont les dimensions seraient excessives, même pour un gros ordinateur.

Il y a deux graphiques, où les points sont repérés par les deux types d'identificateurs à 3 et 4 caractères.

(sous-programmes = VUGRA, PROD, GSMOD, VPROP, TRIDI, FPLAN)

d) - Analyse des correspondances du tableau des caractéristiques socio-économiques des zones.

Editions traditionnelles = valeurs propres, taux d'inertie, coordonnées, contributions absolues et relatives, graphique du plan (1,2).

Calcul du tableau de coordonnées factorielles sur NVDIM axes, en vue de la classification.

(sous-programme = BAS2, AFCOR, VPROP, TRIDI, VPOUR, CONTC, FPLAN)

e) - Agrégation avec contrainte de contiguïté sur les NOBS sommets. [Avec, de plus, contrainte sur les effectifs des classes si KMAX < NOBS. Si KMAX = 30, par exemple, une classe est figée dès que son effectif égale ou dépasse 30. Dans la situation la plus extrême, elle peut provenir de l'agrégation de deux classes d'effectif 29, et donc contenir finalement 58 (2(KMAX - 1)) éléments].

Listage des intermédiaires classiques = Noeuds, aînés, benjamins.

(sous-programmes appelés =

MIZAJ = Mise à jour de la matrice de contiguïté à partir du codage réduit.

AGREG = Agrégation des coordonnées des sommets contigus les plus proches.

MIZD = Mise à jour des distances après agrégation).

f) - Editions des descriptions des noeuds. Liste, pour chaque noeud, des identificateurs des sommets (à 3, puis 4 caractères).

(sous-programme DCLAS)

g) - Description de certains noeuds spécifiés sur une carte paramètre : le numéro du noeud définit un tronquage de l'arbre, dont les noeuds non-triviaux sont décrits (code en 4 caractères).

Ces noeuds sont ensuite positionnés sur le graphique généré au paragraphe c) (le graphique correspondant se fait sur 4 pages s'il y a plus de 100 noeuds).

(sous-programmes DNEU, NESUP, DCLAS, FPLAN)

h) - Si KMAX < NOBS, description de la partition finale (position des noeuds dans le plan factoriel géographique du § c).

(sous-programmes DNEU, NESUP, DCLAS, FPLAN)

i) - Dans tous les cas = positionnement de tous les noeuds sur le plan factoriel du § d. (Graphique en 3 exemplaires).

(sous-programmes NESUP, DCLAS, FPLAN)

### 3.2 Utilisation du programme :

Il y a deux cartes FORTRAN à changer éventuellement :

- la carte DIMENSION Q (...) [Q(MOTS)]

- la carte MOTS = ...

La valeur minimale de MOTS se calcule à partir des paramètres de dimension qui seront lus sur la carte paramètre 2.

1) *Carte paramètre 1* :(20A4) (lue dans le programme principal)

- Titre du travail en format libre

2) *Carte paramètre 2* :( 5I5) (lue dans SAS)

On lit : NIND, NODIM, NWDIM, NVDIM, ND

NIND Majore le nombre d'ilots à classer (NOBS ou CARDJ dans le listage FORTRAN)

NODIM Majore (NOBS + NVAR + 1) (voir ci-dessous)

NWDIM Majore NVAR (NVAR = nombre de variables décrivant une zone)

NVDIM Majore la dimension du sous-espace d'approximation (KFAC)

(NVDIM ≤ NVAR - On prendra i.e. : NVDIM = 8 ou 10)

- ND Majore NOBS x NDEG (degré Max. du graphe) et NWDIM x NWDIM  
 On peut alors savoir que MOTS doit être supérieur à :  
 $M = 2ND + (NVDIM + 16) \times NODIM$  (Majorant approché).  
 Si MOTS est inférieur à cette quantité, le programme est interrompu
- 3) *Carte paramètre 3* : (lue dans SAS)  
 Lecture de KFAC, KMAX, LVU, (3I5)  
 KFAC = Dimension du sous-espace (i.e. = KFAC = 10)  
 KMAX = Borne pour l'effectif des classes  
       (si KMAX = 10 : pas de classes d'effectif  $\geq 20$ )  
 Si LVU = 0 pas de visualisation  
 Si LVU = 1 visualisation et agrégation  
 Si LVU = 2 visualisation seule
- 4) *Carte paramètre 4* : (lue dans DONE5)  
 Cette carte contient en 5° colonne le numéro de la bande magnétique (en principe : 1) sur laquelle sont lues les données. Si cette valeur est nulle (carte vierge), les données sont lues sur carte.
- 5) *Carte paramètre 5*  
 Lecture de NOBS, NVAR, NDEG (3I5)  
 NOBS = Nombre d'ilots  
 NVAR = Nombre de variables décrivant les flots  
 NDEG = Degré Maximum (de l'ordre de 10 en général) du graphe  
       (nombre maximum d'ilots contigus à un ilot)
- 6) *Carte de pondération* : (en 16F5.3) - il y a 2 cartes 6 -  
 Ces cartes contiennent NVAR coefficients de pondération, éventuellement égaux à 1.  
 (Ces coefficients de pondération sont fournis par un programme utilisé en amont de la classification proprement dite).
- 7) *Carte donnant les identificateurs des variables* (Format 25A3)  
 Cette carte contient les NVAR identificateurs nécessaires pour les graphiques d'analyse des correspondances et les listages de résultats (coordonnées, contributions) correspondants.
- 8) *Format des données* : (en 20A4)  
 Il s'agit du Format (FORTRAN) de lecture, ilot par ilot, des NVAR variables qui les caractérisent. Cette lecture, rappelons-le, se fait sur la bande "1" ou sur carte selon que la carte 4 contient un 1 en colonne 5 ou est vierge.
- 9) *Cartes données* : Elles sont lues selon le format de la carte 8. En tête de chaque ilot figurent un identificateur à 4 caractères et un identificateur à 3 caractères de l'ilot. Il n'y a pas de cartes données si la lecture se fait sur bande.
- 10) *Identificateurs des noeuds* : (Format 25A3)  
 Il y a NOBS ilots. Les noeuds seront identifiés depuis NOBS + 1 jusqu'à 2NOBS.  
 Exemple : pour 704 ilots, le premier noeud sera le noeud 705 - au-delà de 999, on fera intervenir les lettres de l'alphabet.ex :  
 1024 = A24, 1099 = A99, 1118 = B18, 1200 = COO, 1234 = C34 etc...
- Il y a (NOBS - 1)/25 cartes 10  
 (Si (NOBS - 1) n'est pas divisible par 25, et si la division donne par exemple 18,3, il y aura évidemment 19 cartes 10).

11) *Matrice de contiguïté* : NOBS cartes en (2014). Le Format figure dans le sous-programme CONTI, et peut être facilement modifié.

12) Carte de commande des noeuds à décrire, en 1515

(on portera les numéros des noeuds à la suite des numéros des sommets : s'il y a 500 flots, le premier noeud a pour numéro 501).

- Eventuellement d'autres cartes pourront être lues à partir du sous-programme DONE5, qui sert d'antichambre pour le prétraitement des données.

N.B. Les 10 sous-programmes dont les noms suivent ne sont pas listés :

AFCOR, CONTC (Analyse des correspondances et éditions)

GSMOD (Orthonormalisation)

VPROP, TRIDI, VPOUR, (Diagonalisation)

SHELK (Tri)

FPLAN, EPURE, BORNS (Graphiques)

Les listages correspondants sont publiés dans l'ouvrage "Techniques de la description Statistique" (L. Lebart, A. Morineau, N. Tabard, DUNOD 1977).

*N. B. La version publiée ci-dessus comporte quelques modifications par rapport à celle utilisée actuellement dans les services concernés des directions régionales des Télécommunications.*

*Bien loin d'introduire des complications supplémentaires, l'intervention de la contrainte de contiguïté a pour effet de diminuer l'encombrement et d'accélérer l'exécution du programme sans contrainte, car l'ensemble des couples examinés par l'algorithme est considérablement restreint.*

*Des techniques de classification hiérarchique peuvent introduire des contraintes analogues en faisant intervenir des seuils de distances, selon la suggestion ancienne de P.H. Sneath (Computer in taxonomy, J. Gen. Microbiol, 17, pp 201-226 ; 1957).*

*C'est ainsi que fonctionne l'algorithme de M. Bruynooghe (Cahiers de l'Analyse des Données, n° 1, pp 7-33 ; 1978) permettant de traiter de vastes tableaux.*

*Addendum :*

*Dans le listage du programme, il convient de modifier le sous-programme "CONTI".*

*Après l'instruction :*

*1 CONTINUE*

*ajouter :*

*LX(NOBS + 1) = NOBS \* NDEG*

*Cette rectification a été proposée par A. et G. Thauront.*

```

C..... PROGRAMME AGRAF.
C..... CLASSIFICATION AVEC CONTRAINTE DE CONTIGUITE ET D EFFECTIFS
C..... AGRÉGATION ASSURANT UNE PERTE D INERTIE MINIMALE A CHAQUE ETAPI
C.....
C..... DIMENSION NOM(20)
C..... *****
COMMON/BBB/NB, NC, NCUIS, NBEL
COMMON/ENSOR/LEC, IMP
C..... DEUX CARTES FORTRAN A CHANGER....
C..... DIMENSION Q(7000)
C..... MOTS = 7000
C..... *****
C..... *****
NCUS = 2
NB=3
NBEL = 4
IMP=6
LEC=5
C..... LECTURE ET IMPRESSION DU TITRE EN 20 A4
102 FORMAT(20A4)
101 WRITE(IMP, 101)(NOM(K), K=1, 20)
C..... PARAMETRES DE DIMENSION . NIND, NODIM, NVDIM, ND
C..... NIND MAJORE NOBS ( NIND = NOBS EST POSSIBLE)
C..... NODIM MAJORE... NOBS + NVAR + 1
C..... NODIM MAJORE LE NOMBRE DE VARIABLES INITIALES (DONC NVAR)
C..... NVDIM MAJORE LE NOMBRE DE DIMENSIONS RETENUES, DONC KEAC
C..... ND MAJORE NOBS*DEGRE MAXIMUM
C..... NVDIM EST PLUS PETIT QUE NVDIM...
553 FORMAT(13I5)
WRITE(IMP, 554) NIND, NODIM, NVDIM, NVDIM, ND
554 FORMAT(7H NIND= ,16,8H, NODIM= ,16,8H, NVDIM= ,16,
1 8H, NVDIM= ,16,8H, ND= ,16)
ND1 = 1
ND2 = ND1 + NODIM*NVDIM
ND3 = ND2 + ND
ND4 = ND3 + ND
NZ = NODIM
ND5 = ND4 + NZ
ND6 = ND5 + NZ
ND7 = ND6 + NZ
ND8 = ND7 + NZ
ND9=ND8
ND10 = ND9 + NZ
ND11=ND10+2*NIND
ND12 = ND11 + NZ
ND13 = ND12 + NZ
ND14 = ND13 + NZ
ND15=ND14 + 2*NIND
ND16 = ND15 + NODIM
ND17 = ND16 + NODIM
ND18 = ND17 + NODIM
NF IN = ND18 + NODIM
WRITE(IMP, 229)
229 FORMAT(//, 20H CONTROLE . MOTS=
,16,6H NF IN= ,16, //)
IF(NF IN, GT, MOTS) GO TO 100
CALL SAS
10Z, NODIM, NVDIM, ND,
1 Q(ND1), Q(ND2), Q(ND3), Q(ND4), Q(ND5), Q(ND6), Q(ND7), Q(ND8), Q(ND9), Q(ND10),
1 Q(ND11), Q(ND12), Q(ND13), Q(ND14), Q(ND15), Q(ND16), Q(ND17),
1 Q(ND18)
)
100 CONTINUE
END
SUBROUTINE DOTES
1(X, NOBS, NVAR, NODIM, NVDIM, NF, IDOBS, IDVAR, P, ID4 , KA, NDEG, MD, ND, LD)
C..... LECTURE DES DONNEES CONFORMEMENT A LA NOTICE
C..... PEUT ETRE ADAPTE... PENSER A MODIFIER LA NOTICE.....
DIMENSION COEF(40), FMT(40), ID4(NODIM), X(NVDIM)
DIMENSION KA(ND), NDEG(NODIM), MD(NODIM), LD(NODIM), LK(NODIM)
DIMENSION IDVAR(NODIM), F(NODIM), IDOBS(NF)
COMMON/ENSOR / LEC, IMP
COMMON/BBB/NB, NC, NCUIS, NBEL
READ(LEC, 553) NBZ
IF(NBZ, EQ, 0) NBZ = LEC
C..... PARAMETRES FONDAMENTAUX . NOBS, NVAR, NDEG
C..... NOBS = NOMBRE DE ZONES A CLASSER, NVAR = NOMBRE DE VARIABLES
C..... NDEG = DEGRE MAXIMAL DU CHAPPE
553 FORMAT(//, 7H NOBS= ,16,8H, NVAR= ,16,8H, NDEG= ,16, //)
READ(LEC, 553) NOBS, NVAR, NDEG
553 FORMAT(15I5)
WRITE(IMP, 556) NOBS, NVAR, NDEG
556 FORMAT(//, 7H NOBS= ,16,8H, NVAR= ,16,8H, NDEG= ,16, //)
C..... LECTURE DES Ponderations
56 FORMAT(16F6, 3)
READ(LEC, 56) (COEF(K), K=1, NVAR)
C..... LECTURE DES IDENTIFICATEURS DES VARIABLES
8888 FORMAT(25A3)
READ(LEC, 8888) (IDVAR(K), K=1, NVAR)
REWRITE(ND)
C..... .NFOR = 20 OU 40 S IL Y A UNE OU DEUX CARTES FORMAT.
READ(LEC, 500) (FMT(LL), LL=1, NFOR)
500 FORMAT(20A4)
DO 300 I=1, NOBS
DO 310 J = 1, NVAR
READ(NBZ, FMT) ID4(I), IDOBS(I), (X(J), J=1, NVAR)
DO 310 J = 1, NVAR
X(J) = X(J) *COEF(J)
WRITE(NB) (X(J), J=1, NVAR)
300 CONTINUE
REWRITE(NB)
C..... LECTURE DES IDENTIFICATEURS DES NOEUDS...
NO1 = NOBS + 1
NO2 = 2*NOBS
READ(LEC, 8888) (IDOBS(I), I=NO1, NO2)
C..... ICI, LES OBSERVATIONS ONT LE NETE POIDS
DO 400 I=1, NOBS
400 P(I)=1
C..... LECTURE DE LA MATRICE DE CONTIGUITE DANS LE S/P CONTI
CALL CONTI(NOBS, NDEG, NODIM, ND, KA, NDEG, MD, LK)
C..... ICI, LECTURE SEPARÉE DES IDENTIFICATEURS DES OBSERVATIONS.
READ(LEC, 8888) (IDOBS(I), I=1, NOBS)
READ(LEC, 500) (ID4(I), I=1, NOBS)
C..... FIN DE LA MODIFICATION EXCEPTIONNELLE...
DO 10 I=1, NOBS
10 IDVAR(I)NVARO = IDOBS(I)
IDVAR(I)NVAR = IDOBS(I)
WRITE(IMP, 42)
WRITE(IMP, 1001)
1001 FORMAT(//, 40H DICTIONNAIRE DES CODES EN A3 ET A4.
DO 301 I = 1, NOBS, 6
DO 301 I = 1, NOBS, 6
NJ = I+5
IF(NJ, GT, NOBS) NJ = NOBS
301 WRITE(IMP, 1000) (( IDOBS(L), ID4(L), L= NJ, NJ)
1000 FORMAT(5X, 6(5X, A3, 5H --- , A4))
42 WRITE(IMP, 42)
42 FORMAT(1X, 120(1E-))
RETURN
END

```



C..... INITIALISATION DU TABLEAU DIS

```

DO 50 J = 1, NOBS
KDEC = MDEC(J)
KD = MD(J)
IF ( KD .GT. KDEC ) GO TO 50
JZ = LK(J)
DO 51 LD = 1, KDEC
J1 = KAJZ+LD)
51 DIS(JZ+LD) = DISK2(PIJ, J1, J, NVAR, NODIM, NVDIM, L)
C..... FIN DE L INITIALISATION
NOBS1 = NOBS - 1
KLAS = LC - 1
J = LC - 1
1000 K = J + 1
      K = J - LC + 1
DO 116 JJ = 1, NOBS
IF(JJ.EQ.NOBS) GO TO 12
IF(MDEG(JJ).EQ.0) GO TO 116
SP = JJ
GO TO 117
116 CONTINUE
117 CONTINUE
S=KA(JY+1)
INF = DIS(JY+1)
KLAS = KLAS + 1
DO 115 JZ=1, NOBS1
JZ = LK(JZ)
KD=MD(JZ)
KDEC=MDEC(JZ)
IF(KD.GT.KDEC) GO TO 115
DO 30 LD=KD, KDEC
J1 = KAJZ + LD)
DJ1J2 = DIS(JZ+LD)
IF(DJ1J2-INF).55, 30, 30
35 INF=DJ1J2
S=J1
SP=J2
30 CONTINUE
115 CONTINUE
A(K) = KING(SOM(S), SOM(SP))
B(K) = MAX0(SOM(S), SOM(SP))
D(K) = INF
IA = A(K)
IB = B(K)
P(J) = P(IA) + P(1B)
60 SOM(S)=J
SOM(SP)=J
WRITE(IMP, 9) J, D(K), A(K), B(K), P(J), S, SP
9 FORMAT(2X, I4, F8.5, 4X, I4, 6X, I8, 5X, I3, 10X, I8, (3, 1E., 13, 1E) )
IF(J.SQ.KC) GO TO 11
C..... MISE A JOUR DE LA MATRICE DE CONTIGUITE KA
CALL MIZAJ(KA, S, SP, NOBS, NODIM, NO, MDEC, MD, LK, DIS)
C..... AGRÉGATION DES ZONES CONTIGUES A DISTANCE MINIMALE
CALL AGRÉG2(PIJ, L, NODIM, NVDIM, NVAR, S, SP)
IF(P(J) .GE. KMAX) CALL SUPPR(KA, MDEC, MD, LK, NOBS, MD, SP)
C..... MISE A JOUR DU TABLEAU DES DISTANCES DIS
CALL MIZO
1 (PIJ, DIS, KA, S, SP, NOBS, NVAR, NODIM, NO, NVDIM, MDEC, MD, L, LK)
IF(P(J) .LT. KMAX) GO TO 11
MDEC(SP) = 0
ND(SP) = 1
KC = KC - 1
11 CONTINUE
12 CONTINUE
IF(J .LT. KC) GO TO 1000
12 CONTINUE
KC = KLAS

```

SUBROUTINE SAS

```

1(CVZ, NODIM, NVDIM, NVDIM, MD,
1, PIJ, T, DIS, NDM, IV, KA, N, IV, D, L, NA, MDEC, ITM, MD, LIP, IP, LK, NOM,
IA, B, SOM, P, KNOM, ID4, PRI, PRJ)
C.....
DIMENSION PIJ(NODIM, NVDIM), TV(NODIM, NVDIM, K(ND), DIS(ND))
DIMENSION ITM(NODIM, LIP(NODIM), IP(NODIM), K(NODIM), RUM(NODIM))
DIMENSION DI(NODIM, L(NODIM), LV(NODIM), IV(NODIM), NA(NODIM))
DIMENSION A(NODIM, B(NODIM), SOM(NODIM), NOM(NZ), P(NZ), L(LSTV(30))
DIMENSION KNOM(NODIM), ID4(NODIM), MDEC(NODIM), MD(NODIM)
DIMENSION PRI(NODIM), PRJ(NODIM), LK(NODIM)
INTEGER A, B, SOM, P, S, SP
REAL INF, P
C..... COMMON/BBB/IB, NC, NCIS, NBEEL
COMMON/ENSOB/ LEG, IMP
553 FORMAT(15I5)
C..... PARAMETRES SERVANT AU CALCUL.. KFAK, KMAX, LVU
C..... KFAK = DIMENSION DU SOUS-ESPACE.. I.E. KFAK = 10
C..... KMAX = EFFECTIF MAXIMAL D UNE CLASSE
C..... LVU=0...PAS DE RECONSTITUTION DU GRAPHE
C..... LVU=1...RECONSTITUTION ET AGRÉGATION
C..... LVU=2...RECONSTITUTION SEULE
READ(LEG, 553) KFAK, KMAX, LVU
WRITE(IMP, 559) KFAK, KMAX, LVU
555 FORMAT(//, 7I8, KFAK= , 16, 8H, KMAX= , 16, 8H, LVU= , 16, //)
JMAX = NODIM
C..... APPEL DES DONNEES, ICI SUR LE SUPPORT EB
C..... NOBS OBJETS A CLASSER, DECRITS PAR NVAR VARIABLES
CALL BONES, NVAR, NODIM, NVDIM, NZ, NOM, KNOM, PRJ, ID4, KA, MDEC, MD, ND, LK)
1(PRI, NOBS, NVAR, NODIM, NVDIM, NZ, NOM, KNOM, PRJ, ID4, KA, MDEC, MD, ND, LK)
IF(LVU.EQ.0) GO TO 1515
ITER = 100
KFAK = 100
C..... APPEL DE LA VISUALISATION DU GRAPHE DE CONTIGUITE
CALL UNVAR
1(KA, NODIM, ND, NVDIM, MDEC, PIJ, NOBS, NITER, KF, PRI, LK, L, PRJ, NOM, ID4)
1515 CONTINUE
IF(LVU.EQ.2) GO TO 888
WRITE(IMP, 100) KFAK
C..... CALCUL DE LA BASE, PIJ PERMETTANT DE CALCULER LES DISTANCES
CALL BAS2(PIJ, NOBS, NVAR, NODIM, NVDIM, PRJ, L, T, PRI, D, JMAX, KFAK, KNOM)
NVAR = KFAK
C..... ATTENTION , AVEC BAS2, NVDIM CONCERNE LE NOMBRE DE FACTEURS
C
WRITE(IMP, 100)
WRITE(IMP, 42)
WRITE(IMP, 31) KMAX
31 FORMAT(//, 23H CLASSIFICATION (KMAX= , 14, 2H) , //)
WRITE(IMP, 42)
PRINT 6
6 FORMAT(//1X, BENEUD, 2X, 6INDICE, 2X, 5H A1VE, 2X, 5BENJAMIN, 2X, 5BPOI
1DS)
LC = NOBS + 1
KC = 2*NOBS - 1
C..... NOYAU DE LA PROCEDURE D AGRÉGATION
DO 10 J = 1, NOBS
SOM(J)=J
D(J)=0
A(J)=0
B(J)=0
P(J)=1
10 CONTINUE
DO 49 LND = 1, ND
49 DIS(LND) = 0.

```

```

C..... DESCRIPTION DES CLASSES AVEC IDENTIFICATEURS EN 3 PUIS 4 CARACT.
DO 411 KKK=3,4
WRITE(IMP,100)
WRITE(IMP,42)
WRITE(IMP,36) KKK
38 FORMAT(9H CODE EN ,12,12H CARACTERES ,/)
WRITE(IMP,42)
WRITE(IMP,600)
600 FORMAT(20H I J I I (J) I A(J) I B(J) I P(J) I ,5X,
140DESCRIPTION DES CLASSES DE LA HIERARCHIE ,49X,1H1)
DO 419 J = 1C,KC
IF(KKK.EQ.3)CALL DCLAS(J,NOBS,N,NUM,NOM,D,A,B,P,KC,L,Y,1)
IF(KKK.EQ.4)CALL DCLAS(J,NOBS,N,NUM,IDA,D,A,B,P,KC,L,Y,1)
410 CONTINUE
411 CONTINUE
C..... DESCRIPTION ET VISUALISATION GRAPHIQUE DE CERTAINS NOEUDS
C..... DANS L'ESPACE GEOGRAPHIQUE (PLAN ISSU DE VUCRA...)
READ(LEC,553)(LIST(KNEU),KNEU = 1,15)
DO 370 KNEU = 1,15
IF(LIST(KNEU).EQ.0) GO TO 370
NEU = LIST(KNEU)
CALL DNEU(NEU,NOBS,N,NUM,IDA,D,A,B,P,KC )
CALL NESUP
1(NZ,L,D,PRI,PRJ,NOM,ITM,NEU,NOBS ,N,NUM,NOM,D,A,B,P,KC ,1)
370 CONTINUE
IF(KKAK.GE.NOBS) GO TO 710
WRITE(IMP,42)
WRITE(IMP,43)
43 FORMAT(//,34H PARTITION FINALE AVEC CONTRAINTES ,//)
NEU = KLAS
CALL DNEU(NEU,NOBS,N,NUM,IDA,D,A,B,P,KC )
CALL NESUP
1(NZ,L,D,PRI,PRJ,NOM,ITM,NEU,NOBS ,N,NUM,NOM,D,A,B,P,KC ,1)
710 CONTINUE
C..... VISUALISATION DES NOEUDS DANS LE PLAN FACTORIEL (ISSU DE AFG0R)
WRITE(IMP,42)
WRITE(IMP,687)
687 FORMAT(//,45H POSITION DES NOEUDS DANS LE PLAN FACTORIEL ,//)
NEU = KLAS
CALL NESUP
1(NZ,L,D,PRI,PRJ,NOM,ITM,NEU,NOBS ,N,NUM,NOM,D,A,B,P,KC ,2)
688 CONTINUE
100 FORMAT(1H1)
42 FORMAT(1X,120(1H-))
STOP
END
C..... INDICE DE DISTANCE CALCULE SUR LES COORDONNEES FACTORIELLES
DIMENSION X(NODIM,NVDIM),P(NODIM)
DIS=0
DO 1 J=1,NVAR
A=X(N1,J)-X(N2,J)
1 DIS=DIS+A*A
DISK2=DIS*(N1)*P(N2)/(P(N1)+P(N2))
RETURN
END

```

```

SUBROUTINE VUCRA
1(KA,NODIM,ND,NVDIM,MDEC,PIJ,NOBS,NITER,KF,PRI,LX,L1,PRJ,NOM,IDA)
C..... VISUALISATION DU GRAPHE DE CONTIGUITE PAR ANALYSE DES CORRESP.
C..... DE LA MATRIE A PARTIR DU CODAGE REDUIT(SANS MISE EN MEMOIRE)
DIMENSION KA(ND),MDEC(NODIM),PIJ(NODIM,NVDIM),PRI(NODIM),LK(NODIM)
DIMENSION LA(ND),MDEC(NODIM),PIJ(NODIM,NVDIM),PRI(NODIM),LK(NODIM)
DIMENSION AA(10,10),VAP(10)
COMMON/ENSOR/LEC,IMP
COMMON/BBB/NB,NC,NGUS,NBEL
REAL L
SSS = 0.
DO 1 I=1,NOBS
L(I) = MDEC(I)
SSS = SSS + L(I)
DO 1 J=1,KF
1 PIJ(I,J) = SENS(A(BBB))
2 L(I) = L(I)/SSS
C..... RECHERCHE D'UN SUPPORT PAR PUISSANCE ITEREE, PUIS ORTHONORM.
DO 555 KB = 1,NITER
CALL PROD(KA,NODIM,ND,NVDIM,MDEC,PIJ,NOBS,KF,PRI,LX,L)
CALL GSMOD ( NODIM,NOBS,KF,L,PIJ,KR,PRI,PRJ)
555 CONTINUE
REWRITE(NGUS)
DO 10 K = 1,KF
10 WRITE(NGUS)(PIJ(I,K),I=1,NOBS)
CALL PROD(KA,NODIM,ND,NVDIM,MDEC,PIJ,NOBS,KF,PRI,LX,L)
REWRITE(NGUS)
DO 9 K = 1,KF
DO 9 KP = 1,KF
9 AA(K,KP) = 0.
DO 11 KP = 1,KF
READ(NGUS)(PRI(I),I=1,NOBS)
DO 13 K = 1,KF
DO 13 I = 1,NOBS
13 AA(K,KP) = AA(K,KP) + PIJ(I,K)*PRI(I)*L(I)
11 CONTINUE
C..... DIAGONALISATION DU NUAGE PROJETE SUR LE SUPPORT PRECEDENT
CALL VPROP(10,KF,AA,VAP,PRJ,MODE)
REWRITE(NGUS)
DO 14 K = 1,KF
14 READ(NGUS)(PIJ(I,K),I=1,NOBS)
DO 16 I = 1,NOBS
PRI(I) = 0
PRJ(I) = 0
DO 15 K = 1,KF
PRI(I) = PRI(I) + PIJ(I,K)*AA(K,1)
15 PRJ(I) = PRJ(I) + PIJ(I,K)*AA(K,2)
16 CONTINUE
REWRITE(NGUS)
WRITE(NGUS) (PRI(I),I=1,NOBS),(PRJ(I),I=1,NOBS)
WRITE(IMP,42)
WRITE(IMP,39)
39 FORMAT(//,26H RECONSTITUTION DU GRAPHE
CALL PLAN(NODIM,NOBS,1,2,PRI,PRJ,NOM,1,123,2,2,5,1)
42 FORMAT(1X,120(1H-))
RETURN
END

```

C..... OPTION ....SUPPRESSION DES ARETES LITIGIEUSES....  
 IZER = NOBS + 1  
 WRITE(IMP,100)  
 100 FORMAT(1H1)  
 CALL SUPPR(KA,MDEC,ND,LX,NOBS,ND,IZER)  
 RETURN  
 LND

C..... SUBROUTINE SUPPR(KA,MDEC,ND,LX,NOBS,ND,IZER)  
 DIMENSION KA(WD),MDEC(NOBS),MD(NOBS),LK(NOBS) KA,  
 COMMON/ENSOR/LEC,IMP  
 IF(IZER.LE.NOBS) GO TO 100  
 WRITE(IMP,42)  
 WRITE(IMP,10)  
 WRITE(IMP,42)  
 100 CONTINUE ,//)

DO 30 I = 1,NOBS  
 JI = LK(I)  
 KDA = MDEC(I)  
 IF(KDA.EQ.0) GO TO 30  
 K = 0  
 DO 31 L = 1,KDA  
 KIL = KA(JI + L)  
 IF(KIL.EQ.IZER) GO TO 31  
 K = K + 1  
 IF(KIL .LE. I) KD = KD + 1  
 KA(JI + K) = KIL  
 MD ( I ) = KD + 1  
 MDEC ( I ) = K  
 JDEB = JI + 1  
 JFIN = JI + K  
 IF(IZER.LE.NOBS) GO TO 30  
 IF(JDEB.GT.JFIN) GO TO 30  
 WRITE(IMP,3) I, (KA(KZ),KZ = JDEB,JFIN)  
 3 FORMAT(3H \*,14,3H \*,1615)  
 30 CONTINUE  
 42 FORMAT(1X,120(1H-))  
 RETURN  
 END

31 CONTINUE  
 MD ( I ) = KD + 1  
 MDEC ( I ) = K  
 JDEB = JI + 1  
 JFIN = JI + K  
 IF(IZER.LE.NOBS) GO TO 30  
 IF(JDEB.GT.JFIN) GO TO 30  
 WRITE(IMP,3) I, (KA(KZ),KZ = JDEB,JFIN)  
 3 FORMAT(3H \*,14,3H \*,1615)  
 30 CONTINUE  
 42 FORMAT(1X,120(1H-))  
 RETURN  
 END

C..... SUBROUTINE DREIC(NEU,JJJ,N,NUM,NOM,D,A,B,P,KCARD)  
 DESCRIPTION JUSQU AU NOEUD...NEU...DE LA HIERARCHIE  
 DIMENSION A(JJJ),B(JJJ),N(JJJ),D(JJJ),NUM(JJJ)  
 DIMENSION NOM(JJJ),P(KCARD)  
 INTEGER A,B,P  
 COMMON/ENSOR/LEC,IMP  
 LCARD= JJJ + 1  
 WRITE(IMP,42)  
 WRITE(IMP,20) NEU  
 20 FORMAT(///,40H DESCRIPTION DES GROUPES JUSQU AU NOEUD= ,15,///)  
 42 FORMAT(1X,120(1H-))  
 DO 1 KV = LCARD,NEU  
 DO 10 K = KV,NEU  
 L=K-JJJ  
 IF(KV.EQ.A(L).OR.KV.EQ.B(L)) GO TO 1  
 CALL DCLAS(KV,JJJ,N,NUM,NOM,D,A,B,P,KCARD,LX,1)  
 1 CONTINUE  
 1 RETURN  
 END

31 CONTINUE  
 MD ( I ) = KD + 1  
 MDEC ( I ) = K  
 JDEB = JI + 1  
 JFIN = JI + K  
 IF(IZER.LE.NOBS) GO TO 30  
 IF(JDEB.GT.JFIN) GO TO 30  
 WRITE(IMP,3) I, (KA(KZ),KZ = JDEB,JFIN)  
 3 FORMAT(3H \*,14,3H \*,1615)  
 30 CONTINUE  
 42 FORMAT(1X,120(1H-))  
 RETURN  
 END

C..... SUBROUTINE CONTI(NOBS,MDEC,NODIN,ND,KA,MDEC,ND,LX)  
 LECTURE,CORRECTIONS DE LA MATRICE DE CONTIGUITE  
 DIMENSION KA(ND),MDEC(NODIN),LK(NODIN),MD(NODIN),V(50),KV(5)  
 COMMON/ ENSOR/ LEC,IMP  
 NY = NOBS\*MDEC  
 DO 500 I=1,NY  
 500 KA(I) = 0  
 DO 1 I=1,NOBS  
 DO 10 I=1,NOBS  
 LK(I) = (I-1)\*MDEC  
 C..... LECTURE DE LA MATRICE DE LA MATRICE DE CONTIGUITE  
 JID = LK(I) + 1  
 JIF = JID + 19  
 READ(LEC,2) (KA(KL),KL=JID,JIF)  
 KD=0  
 KDEC=0  
 KDEV = 0  
 JI = LK(I)  
 DO 4 LL=1,MDEC  
 IF(KA(JI+LL).EQ.1) KA(JI+LL)=NOBS+1  
 KKK = KA(JI+LL)  
 IF(KKK.GT.NOBS) WRITE(IMP,5) I  
 5 FORMAT(27H ATTENTION ERREUR SOMMET ,15)  
 KDEC=KDEC+1  
 IF(KKK.LE.NOBS) KDEV = KDEV + 1  
 IF(KKK.LE.1) KD = KD + 1  
 4 CONTINUE  
 MD(I)=KD+1  
 C..... CLASSEMENT DES ADJACENTS PAR NUMEROS CROISSANTS  
 DO 165 JI=1,KDEV  
 CALL SHELK(KDEC,V,KV)  
 KDEV = KDEV  
 MDEC(I) = KDEC  
 IF(KDEC.EQ.0) GO TO 1  
 DO 166 JL = 1,KDEC  
 KA(JI+JL) = V(JL)  
 JFIN = JID + KDEC - 1  
 WRITE(IMP,3) I,(KA(KZ),KZ=JID,JFIN)  
 3 FORMAT(3H \*,14,3H \*,1615)  
 1 CONTINUE  
 C..... FORMAT POUR LE SP CONTI  
 2 FORMAT(2014)

C..... CONTROLE INITIAL DE LA MATRICE DE CODAGE CONDENSE  
 DO 11 I=1,NOBS  
 JI = LK(I)  
 KDA = MDEC(I)  
 IF(KDA.EQ.0) GO TO 11  
 DO 10 L = 1,KDA  
 KKK = KA(JI + L)  
 JKK = LK(KKK)  
 KKD = MDEC(KKK)  
 KKB = 1  
 DO 12 LL = 1,KKD  
 KKB = 1 -KA(JKK+LL)  
 12 KKB = KKB\*KBB  
 IF(KKB .EQ. 0 ) GO TO 10  
 WRITE(IMP,20) KKK,I,I,KKK  
 KA(JI + L) = NOBS + 1  
 10 CONTINUE  
 11 CONTINUE  
 20 FORMAT(1H,10H ATTENTION ,14,10H EST SUR LA LIGNE ,14,  
 1 6H MAIS ,14,54H N EST PAS SUR LA LIGNE ,14,///)

C..... LECTURE DE LA MATRICE DE LA MATRICE DE CONTIGUITE  
 JID = LK(I) + 1  
 JIF = JID + 19  
 READ(LEC,2) (KA(KL),KL=JID,JIF)  
 KD=0  
 KDEC=0  
 KDEV = 0  
 JI = LK(I)  
 DO 4 LL=1,MDEC  
 IF(KA(JI+LL).EQ.1) KA(JI+LL)=NOBS+1  
 KKK = KA(JI+LL)  
 IF(KKK.GT.NOBS) WRITE(IMP,5) I  
 5 FORMAT(27H ATTENTION ERREUR SOMMET ,15)  
 KDEC=KDEC+1  
 IF(KKK.LE.NOBS) KDEV = KDEV + 1  
 IF(KKK.LE.1) KD = KD + 1  
 4 CONTINUE  
 MD(I)=KD+1  
 C..... CLASSEMENT DES ADJACENTS PAR NUMEROS CROISSANTS  
 DO 165 JI=1,KDEV  
 CALL SHELK(KDEC,V,KV)  
 KDEV = KDEV  
 MDEC(I) = KDEC  
 IF(KDEC.EQ.0) GO TO 1  
 DO 166 JL = 1,KDEC  
 KA(JI+JL) = V(JL)  
 JFIN = JID + KDEC - 1  
 WRITE(IMP,3) I,(KA(KZ),KZ=JID,JFIN)  
 3 FORMAT(3H \*,14,3H \*,1615)  
 1 CONTINUE  
 C..... FORMAT POUR LE SP CONTI  
 2 FORMAT(2014)

C..... CONTROLE INITIAL DE LA MATRICE DE CODAGE CONDENSE  
 DO 11 I=1,NOBS  
 JI = LK(I)  
 KDA = MDEC(I)  
 IF(KDA.EQ.0) GO TO 11  
 DO 10 L = 1,KDA  
 KKK = KA(JI + L)  
 JKK = LK(KKK)  
 KKD = MDEC(KKK)  
 KKB = 1  
 DO 12 LL = 1,KKD  
 KKB = 1 -KA(JKK+LL)  
 12 KKB = KKB\*KBB  
 IF(KKB .EQ. 0 ) GO TO 10  
 WRITE(IMP,20) KKK,I,I,KKK  
 KA(JI + L) = NOBS + 1  
 10 CONTINUE  
 11 CONTINUE  
 20 FORMAT(1H,10H ATTENTION ,14,10H EST SUR LA LIGNE ,14,  
 1 6H MAIS ,14,54H N EST PAS SUR LA LIGNE ,14,///)

C..... LECTURE DE LA MATRICE DE LA MATRICE DE CONTIGUITE  
 JID = LK(I) + 1  
 JIF = JID + 19  
 READ(LEC,2) (KA(KL),KL=JID,JIF)  
 KD=0  
 KDEC=0  
 KDEV = 0  
 JI = LK(I)  
 DO 4 LL=1,MDEC  
 IF(KA(JI+LL).EQ.1) KA(JI+LL)=NOBS+1  
 KKK = KA(JI+LL)  
 IF(KKK.GT.NOBS) WRITE(IMP,5) I  
 5 FORMAT(27H ATTENTION ERREUR SOMMET ,15)  
 KDEC=KDEC+1  
 IF(KKK.LE.NOBS) KDEV = KDEV + 1  
 IF(KKK.LE.1) KD = KD + 1  
 4 CONTINUE  
 MD(I)=KD+1  
 C..... CLASSEMENT DES ADJACENTS PAR NUMEROS CROISSANTS  
 DO 165 JI=1,KDEV  
 CALL SHELK(KDEC,V,KV)  
 KDEV = KDEV  
 MDEC(I) = KDEC  
 IF(KDEC.EQ.0) GO TO 1  
 DO 166 JL = 1,KDEC  
 KA(JI+JL) = V(JL)  
 JFIN = JID + KDEC - 1  
 WRITE(IMP,3) I,(KA(KZ),KZ=JID,JFIN)  
 3 FORMAT(3H \*,14,3H \*,1615)  
 1 CONTINUE  
 C..... FORMAT POUR LE SP CONTI  
 2 FORMAT(2014)

C..... CONTROLE INITIAL DE LA MATRICE DE CODAGE CONDENSE  
 DO 11 I=1,NOBS  
 JI = LK(I)  
 KDA = MDEC(I)  
 IF(KDA.EQ.0) GO TO 11  
 DO 10 L = 1,KDA  
 KKK = KA(JI + L)  
 JKK = LK(KKK)  
 KKD = MDEC(KKK)  
 KKB = 1  
 DO 12 LL = 1,KKD  
 KKB = 1 -KA(JKK+LL)  
 12 KKB = KKB\*KBB  
 IF(KKB .EQ. 0 ) GO TO 10  
 WRITE(IMP,20) KKK,I,I,KKK  
 KA(JI + L) = NOBS + 1  
 10 CONTINUE  
 11 CONTINUE  
 20 FORMAT(1H,10H ATTENTION ,14,10H EST SUR LA LIGNE ,14,  
 1 6H MAIS ,14,54H N EST PAS SUR LA LIGNE ,14,///)

```

SUBROUTINE BAS2
1(P1J,NOBS,NVAR,NODIM,NVDIM,PRJ,L,S,PRI,D,JMAX,KFAC,KWORD)
C..... RECHERCHE D'UNE BASE DE TRAVAIL PAR ANALYSE DES CORRESPONDANCES
C..... DIMENSIONS DES RESULTATS ET GRAPHIQUES INTERMEDIAIRES
REAL L
DIMENSION P1J(NODIM,NVDIM),S(JMAX,JMAX),PRJ(NODIM),PRI(NODIM)
DIMENSION L(NODIM),D(NODIM)
COMMON/BBB/NB,NC,NCUS,NBEL
COMMON/ENSOR/LEC,IMP
WRITE(IMP,42)
30 FORMAT(//,30H ANALYSE DES CORRESPONDANCES ,//)
42 FORMAT('X,120(1H-))
REWIND,NBEL
KF = 6
NROT = NOBS + NVAR
NF = NROT + 1
C..... ANALYSE DES CORRESPONDANCES, EDITIONS CLASSIQUES
CALL AFOR(NODIM,JMAX,KFAC,NOBS,NVAR,NVAR,S,PRI,PRJ,SS,
1 P1J, L, D )
TRACE = 0.
DO 10 J=2,NVAR
PRI(J) = ABS(PRI(J))
10 TRACE = TRACE + PRI(J)
CALL VPOUR(2,NVAR,10,PRI,TRACE)
WRITE(IMP,100)
100 FORMAT(1H1)
CALL CONTIC
1(NODIM,JMAX,KF ,NOBS,NOBS,NVAR,NVAR,PRI,PRJ,P1J,L,KWORD)
DO 4 K1 = 1,3,2
K2 = K1 + 1
DO 3 I=1,NROT
PRI(I) = P1J(I,K1)
3 D(I) = P1J(I,K2)
WRITE(IMP,100)
4 CALL FPLAN(NT,NTOT,K1,K2,PRI,D,KKOM,1,118,2,2,5,1)
C..... MISE EN MEMOIRE DES COORDONNEES ET DES POIDS
DO 1 I=1,NOBS
DO 2 J=1,KFAC
2 P1J(I,J) = P1J(I+NVAR,J)
L(I) = PRJ(I + NVAR)
1 CONTINUE
WRITE(NBEL)((P1J(I,J),I=1,NOBS),J=1,2)
RETURN
END
SUBROUTINE MESUP
1(NZ,F1,F2,PRI,PRJ,1DOBS,IDX,NEU,JJJ,N,NUM,NOM,D,A,B,P,KCARD,LC)
C..... REPRESENTATION DE LA POSITION DES NOEUDS DANS UN PLAN FACTORIEL
DIMENSION AJJJ,B(JJJ),N(JJJ),D(JJJ),NUM(JJJ),NOM(JJJ),PKCARD
DIMENSION IDN(JJJ),F1(JJJ),F2(JJJ),PRI(JJJ),PRJ(JJJ),1DOBS(NZ)
INTEGER A,B,P
COMMON/BBB/NB,NC,NCUS,NBEL
COMMON/ENSOR/LEC,IMP
LCARD= JJJ + 1
NPAGE = 1
NLCGN = 58
NOBS = JJJ
IPOIN = 0
IF(LC.EQ.1) NBAN = NCUS
IF(LC.EQ.2) NBAN = NBEL
REWIND,NBAN
READ(NBAN) (PRI(I),I=1,NOBS),(PRJ(I),I=1,NOBS)
WRITE(IMP,42)
WRITE(IMP,42)
20 FORMAT(///,40H VISUALISATION JUSQU'AU NOEUD..... 15,///)
C..... LES NOEUDS SONT POSITIONNES EN ELEMENTS SUPPLEMENTAIRES
DO 1 KV = LCARD,NEU
IF(LC.EQ.2) CO TO 30
DO 10 K = KV,NEU
L-K-JJJ
IF(KV.EQ.A(L).OR.KV.EQ.B(L)) CO TO 1
10 CONTINUE
30 CONTINUE
IPOIN = IPOIN + 1
CALL DCLAS(KV,JJJ,N,NUM,NOM,D,A,B,P,KCARD,L,Y,2)
K = IPOIN
F1(K) = 0
F2(K) = 0
IDN(K) = 1DOBS(KV)
RLY = LY
DO 3 LIG = 1,LY
LLL = NOK(LIG)
F1(K) = F1(K) + PRI(LLL)
3 F2(K) = F2(K) + PRJ(LLL)
F1(K) = F1(K) / RLY
F2(K) = F2(K) / RLY
1 CONTINUE
IF(IPOIN.GT.100) NLCGN = 116
IF(IPOIN.GT.100) NPAGE = 2
CALL FPLAN(NOBS,IPOIN,1,2,F1,F2,IDX,1,NLCGN,NPAGE,2,5,1)
42 FORMAT('X,120(1H-))
RETURN
END

```

C..... MISE A JOUR DU CODAGE REDUIT DE LA MATRICE DE CONTIGUITE  
 DIMENSION KA(ND),NDEG(NODIM),MD(NODIM)  
 DIMENSION LK(NODIM),DIS(ND),MA(100),MAUK(100)  
 COMMON/ ENSOR/ LEC,IMP  
 C..... I1 EST FUSIONNE AVEC I2, DANS I2.....  
 IND = 0  
 KIND = 1  
 I1I = 0  
 LL = 0  
 K1=NDEG(I1) + 1  
 K2= NDEG(I2)  
 K3= NDEG(I2)  
 DO 10 I, K2  
 IF(KA(J12+L).EQ.I1) GO TO 10  
 I1= I1 + 1  
 KA(J12 + L) = KA( J12 + L)  
 10 CONTINUE LL = 1  
 C..... FUSION DES LIGNES I1 ET I2  
 KP1= KA(J11 + K1)  
 KA(J11 + K1) = NOBS + 1  
 KA(J12 + K2) = NOBS + 1  
 DO 1 K = 1, K1  
 KK = KA(J11 + K)  
 IF(KK.EQ.I2) GO TO 1  
 M1 = KIND  
 DO 2 L=M1, K2  
 KL = KA(J12 + L)  
 IF(KK - KL) 3,1,4  
 4 IND = IND + 1  
 MAUK(IND)=KL  
 KIND=KIND + 1  
 GO TO 2  
 3 CONTINUE  
 IND = IND + 1  
 MAUK(IND) = KK  
 GO TO 1  
 2 CONTINUE  
 1 CONTINUE  
 KA(J11 + K1 ) = KP1

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

C..... MISE A JOUR DES VOISINS DE I1  
 K3 = NDEG(I1)  
 DO 60 L=1,K3  
 IL = KA(J11 + L)  
 J1L = LK(IL)  
 IF(IL .EQ. I2) GO TO 60  
 KP = NDEG(IL)  
 KD = 0  
 LT = 0  
 NPM=0  
 IF(KF.EQ.0) WRITE(IMP,200) I1,IL  
 200 FORMAT(//5SH ATTENTION ERREUR SUR KF...I1,I2  
 ,2110)  
 DO 61 K = 1,KP  
 KAK = KA(J11 + K)  
 IF(KAK.EQ.I2) GO TO 65  
 IF(KAK .EQ. I1) GO TO 61  
 LT = LT + 1  
 MAULT = KAK  
 IF(MA(LT).GT.I2) GO TO 70  
 GO TO 61  
 70 NHT = NPM + 1  
 IF(NPM .GT. 1) GO TO 61  
 LT = LT + 1  
 MA(LT)=MA(LT-1)  
 MA(LT-1)=12

GO TO 61  
 65 MDEG(IL) = KF - 1  
 61 CONTINUE  
 IF(MM.NE. 0) GO TO 62  
 LT = LT + 1  
 MA ( LT ) = 12  
 62 CONTINUE  
 DO 71 LLL=1,LT  
 IF(MA(LLL).LE.IL) KD = KD + 1  
 71 KA(J1L + LLL)=MA(LLL)  
 MD(IL) = KD + 1  
 60 CONTINUE

C  
 MDEG(I1) = 0  
 MD(I1) = 1  
 J12=LK(I2)  
 MDEG(I2) = IND  
 C..... TRANSLATION DE KA ET DE DIS SI LE DEGRE D UN NOUVEAU SOMMET  
 C..... EST TROP ELEVE...  
 IF((J12+IND) .LE. LK(I2+1))GO TO 300  
 IF(I1 .EQ. (I2+1)) GO TO 300  
 LTRA = LK(I1 + 1) -LK(I1)  
 LFF = LK(I1 + 1)  
 LDD = LK(I2+1)+LTRA  
 DO 290 IZ = LDD, LFF  
 IZ1 = LFF - IZ + LDD  
 DIS(IZ1) = DIS(IZ1-LTRA)  
 290 KACIZ1) = KA(IZ1-LTRA)  
 292 CONTINUE  
 IZ1 = IZ + 1  
 DO 291 IZ = IZ1, I1  
 LK(LZ) = LK(LZ) + LTRA  
 300 CONTINUE  
 IF(IND .EQ. 0) GO TO 400  
 DO 50 L=1,IND  
 IF(MA(L).LE.I2) I1I = I1I + 1  
 KA(J12 + L) = MAUK(L)  
 50 CONTINUE  
 400 CONTINUE  
 MD(I2) = I1I+1

C  
 DO 53 I=1,NOBS  
 NJK=NDEG(I)  
 IF(NJK.EQ.0) GO TO 53  
 J1I = LK(I) + 1  
 NKZ = MD(I)  
 J12 = LK(I)+NJK  
 53 CONTINUE  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

SUBROUTINE ADRG(X,P,NODIM,NVDIM,NVAR,I1,I2)  
 DIMENSION DE I1 ET I2 DANS I2  
 DIMENSION X(NODIM,NVDIM),P(NODIM)  
 A= P(I1)+P(I2)  
 DO 1 J=1,NVAR  
 1 X(I2,J)=(P(I2)\*X(I2,J) + P(I1)\*X(I1,J))/A  
 P(I2)=A  
 RETURN  
 END

```

SUBROUTINE DCIAS(J, JJJ, K, NUM, NOM, D, A, B, P, KCARD, LY, MODE)
DESCRIPTION DU NOEUD J. CF. CAH. N. JAMBU.
DIMENSION N(JJJ, D, C, JJJ), A(JJJ), B(JJJ), NUM(JJJ), NOM(JJJ), P(KCARD)
INTEGER A, B, P
COMMON/ENSOR/ LEC, IMP
KI=1
N(1)=J
LY=0
538 IF(N(KI)-JJJ) 535, 535, 540
535 NPI=N(KI)
LY=LY+1
NUM(LY)=NOM(NPI)
IF(MODE.EQ.2) NUM(LY) = NPI
KI=KI+1
GO TO 534
540 KJ=KI+1
NPI=N(KI)
N(KJ) = A(NPI - JJJ)
N(KI) = B(NPI - JJJ)
KI=KI+1
534 IF(KI.NE.0) GO TO 538
IF(MODE.EQ.2) GO TO 100
PRINT 42
42 FORMAT(IX, 120(1H-))
K = J - JJJ
PRINT400, J, D(K), A(K), B(K), P(J), (NUM(IL), IL=1, LY)
FORMAT(2H 1, 14, 1H 16, 3, 3(1H, 14), 1H, 18(1X, A4), 1H, /,
150(2H 1, 4X, 1H, 16X, 1H, 18(1X, A4), 1H, /, /)
90 CONTINUE
RETURN
END
    
```

```

C.....
SUBROUTINE MIZD
1(P1J, DIS, KA, S, SP, NOBS, NVAR, NODIM, ND, NVDIM, MDEC, MD, P, LK)
MISE A JOUR DU TABLEAU DES DISTANCES
DIMENSION DIS(ND), KA(ND), MDEC(NODIM)
DIMENSION MD(NODIM), P(NODIM), P1J(NODIM, NVDIM)
DIMENSION LK(NODIM)
INTEGER S, SP
KSP = MDEC(SP)
IF (KSP.EQ.0) GO TO 4
JSP = LK(SP)
DO 1 L = 1, KSP
KMOD = KA(JSP + L)
IF(KMOD.LE. SP) GO TO 3
DIS(JSP + L) = DISK2(P1J, SP, KMOD, NVAR, NODIM, NVDIM, P)
3 CONTINUE
KS2 = MDEC(KMOD)
IF(KS2.EQ.0) GO TO 1
DO 2 LL = 1, KS2
JKMOD = LK(KMOD)
KJ2 KA(JKMOD + LL)
IF(KJ2.LE. KMOD) GO TO 2
DIS(JKMOD + LL) = DISK2(P1J, KMOD, KJ2, NVAR, NODIM, NVDIM, P)
2 CONTINUE
1 CONTINUE
4 CONTINUE
RETURN
END
    
```

```

SUBROUTINE PROD(KA, NODIM, NDDIM, NVDIM, MDEC, P1J, NOBS, KT, X, LK, P)
PUISSANCE ITEREE APPLIQUEE AU CODAGE REDUIT
DIMENSION LK(NODIM), P(NODIM)
DIMENSION P1J(NODIM, NVDIM), KA(NDDIM)
DIMENSION X(NODIM), MDEC(NODIM)
DO 3 J=1, KT
TOT = 0
AMOV = 0
DO 1 I=1, NOBS
X(I) = 0
KDEC = MDEC(I)
TOT = TOT + KDEC
TERM = 0.
DO 2 LL = 1, KDEC
K = KA(LL + LL)
2 TERM = TERM + P1J(K, J)/DEC
1 X(I) = TERM
4 P1J(I, J) = X(I) - AMOV / TOT
3 CONTINUE
RETURN
END
    
```

```

C.....
SUBROUTINE PROD(KA, NODIM, NDDIM, NVDIM, MDEC, P1J, NOBS, KT, X, LK, P)
PUISSANCE ITEREE APPLIQUEE AU CODAGE REDUIT
DIMENSION LK(NODIM), P(NODIM)
DIMENSION P1J(NODIM, NVDIM), KA(NDDIM)
DIMENSION X(NODIM), MDEC(NODIM)
DO 3 J=1, KT
TOT = 0
AMOV = 0
DO 1 I=1, NOBS
X(I) = 0
KDEC = MDEC(I)
TOT = TOT + KDEC
TERM = 0.
DO 2 LL = 1, KDEC
K = KA(LL + LL)
2 TERM = TERM + P1J(K, J)/DEC
1 X(I) = TERM
4 P1J(I, J) = X(I) - AMOV / TOT
3 CONTINUE
RETURN
END
    
```