



HAL
open science

Le traçage automatique de cartes d'isovaleurs. Adaptation du programme MAP (Davaud, 1975)

Frédéric Lardeux

► To cite this version:

Frédéric Lardeux. Le traçage automatique de cartes d'isovaleurs. Adaptation du programme MAP (Davaud, 1975). Rapport IRTMLM n° ITRM/11/90/ITRM/Doc-Ent., Institut Louis Malardé, Tahiti, French Polynesia. 1990, pp.1-32. hal-04753809

HAL Id: hal-04753809

<https://hal.science/hal-04753809v1>

Submitted on 25 Oct 2024

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

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



Distributed under a Creative Commons Attribution 4.0 International License

I.T.R.M.L.M.
B.P. 30
Papeete
TAHITI

O.R.S.T.O.M.
B.P. 529
Papeete
TAHITI

LE TRACAGE AUTOMATIQUE DE CARTES D'ISOVALEURS
ADAPTATION DU PROGRAMME MAP (DAVAUD, 1975)

F.LARDEUX

Février 1990

Ref. ITRMLM : n° 11 / 90 /ITRM / Doc-Ent.

SOMMAIRE

INTRODUCTION

1 - L'ALGORITHME

2 - EXEMPLES

3 - LE PROGRAMME INFORMATIQUE

2.1 - Description générale

2.2 - Mise en route

2.3 - Saisie des données - création de fichier

2.4 - Corrections dans un fichier

2.5 - Listage d'un fichier

2.6 - Traçage des cartes d'isovaleurs

2.7 - Le fichier des paramètres (MAP.PAR)

CONCLUSION

BIBLIOGRAPHIE

Annexe : Listing du programme MAP

INTRODUCTION

Le programme décrit ici est une adaptation de l'algorithme publié par Davaud (1975) et permet le traçage automatique de cartes d'isovaleurs.

Le traçage de cartes d'isovaleurs est couramment effectué en géologie ou en géographie: les cartes topographiques où apparaissent les lignes de niveau (altitude) sont des exemples les plus populaires.

Le traçage automatique (à l'aide d'un ordinateur), permet de s'affranchir de la part de subjectivité du traçage manuel, où l'interpolation entre deux points de mesures est parfois délicate. De plus, le gain de temps est non négligeable. En un mot, la rapidité et l'objectivité de la cartographie automatique rendent la méthode supérieure à la cartographie manuelle.

L'application de la méthode à d'autres domaines, comme celui de l'entomologie, est possible (et souhaitable), dès que l'on désire rendre compte de structures spatiales : cartographie d'un gîte larvaire de ceratopogonidae, dont les larves sont enfouies dans le sable, cartographie des densités de moustiques dans une zone donnée etc ...

Bien qu'actuellement des méthodes plus sophistiquées existent pour rendre compte de telles structures (kriegeage, par exemple (Matheron, 1965)), la méthode présentée ici a l'avantage de rester simple et de ne pas demander d'hypothèses de travail sur la structure des données. Le résultat reste descriptif (on obtient simplement une carte) et lié à la qualité des données disponibles.

Le programme présenté ici est adapté de l'algorithme de Davaud (1975) qui a publié une version succincte en FORTRAN IV. Notre version a été écrite en QUICK-BASIC (Microsoft). Une version compilée (programme .EXE) est disponible sur demande pour ceux qui ne disposent pas du compilateur Quick-Basic.

1 - L'ALGORITHME

Comme le souligne Davaud (1975), pour qu'un programme ait un caractère général, le traçage doit pouvoir s'effectuer à partir d'un nombre restreint de mesures, dont la répartition et la densité spatiale sont quelconques. La surface obtenue doit respecter tous les points de mesure et ne pas présenter de discontinuités.

De nombreux algorithmes satisfaisant à ces conditions ont été publiés (cf. la revue faite par Davaud, 1975). Davaud s'est d'ailleurs inspiré de l'ensemble de ces travaux pour mettre au point un algorithme qui repose sur la création successive de grilles de valeurs interpolées dont la maille devient de plus en plus fine. La méthode d'interpolation est fondée sur l'ajustement de polynômes locaux du premier degré avec sélection isotropique des points de mesure.

L'algorithme peut être décrit de la manière suivante (recopié de Davaud, 1975) :

A partir de mesures situées dans un système de coordonnées orthogonales (géographiques ou arbitraires), l'algorithme génère itérativement une succession de grilles de valeurs interpolées dont chaque maille rectangulaire diminue de moitié à chaque itération. Chaque nouvelle grille est calculée à l'aide des valeurs interpolées, obtenues lors des itérations précédentes, et des données initiales, qui ne sont jamais modifiées. Les points communs à plusieurs grilles sont recalculés à chaque itération, mais les points interpolés au

cours d'une itération ne viennent s'ajouter aux données existantes - initiales ou calculées - qu'au début de l'itération suivante. La procédure s'interrompt lorsque les dimensions de la maille équivalent aux dimensions des caractères typographiques de l'imprimante de l'ordinateur. La maille initiale doit donc être une puissance de deux de la maille finale. Cette succession de grilles d'interpolations permet de créer progressivement des mesures complémentaires dans les zones où la densité des points de mesure initiaux est faible ou nulle. La procédure d'interpolation proprement dite s'effectue en trois temps, pour chaque point d'intersection des grilles successives:

- On recherche tout d'abord les données - primitives ou calculées lors des itérations précédentes - situées dans un cercle centré sur le point à interpoler, dont le rayon est un multiple des dimensions de la maille.
- Les données ainsi sélectionnées sont alors pondérées en fonction de l'inverse du carré de la distance qui les sépare du point à interpoler. Cette pondération peut être sélectivement intensifiée de manière à accorder une importance plus grande aux mesures initiales.
- Finalement, on ajuste à ces données triées et transformées un polynôme du premier degré en x et y dont les paramètres permettent de calculer une valeur interpolée pour le point de la grille considéré. Lorsque le nombre de mesures rencontrées au voisinage de ce point n'est pas suffisamment élevé, l'interpolation est localement abandonnée.

Cette procédure se répète pour chaque point d'intersection des différentes grilles successives.

Une fois la dernière grille calculée, le traçage de la surface s'effectue sur l'imprimante de l'ordinateur en remplaçant les valeurs mémorisées par des symboles typographiques.

2 - EXEMPLES

Le programme a été testé sur les exemples fournis par Davaud (1975). A titre d'illustration, la figure 1 représente un de ceux-ci (exemple 2 de Davaud). Il s'agit de côtes de niveau repère de l'Ordovicien, à partir de données provenant de 41 sondages régulièrement espacés (programme : 6 itérations, pondération : 75).

La figure 2 est plus spécifiquement adaptée à l'entomologie. Il s'agit d'une carte de densité de moustiques *Aedes polynesiensis* sur un îlot d'un atoll. Les données sont le nombre de moustiques capturés sur appât vivant durant une période de 10 mn, à un endroit déterminé (coordonnées x,y). Les données sont ici obtenues sur une maille d'échantillonnage non régulière.

La répartition des insectes apparaît clairement en taches, que l'on peut relier à la densité de la végétation (par exemple, Lardeux et al., 1988)

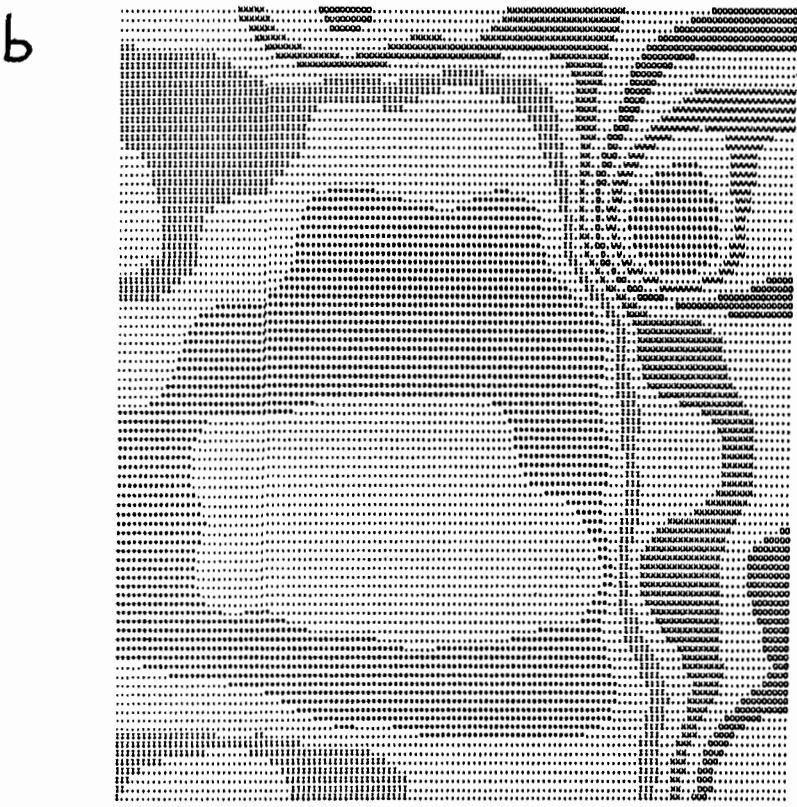
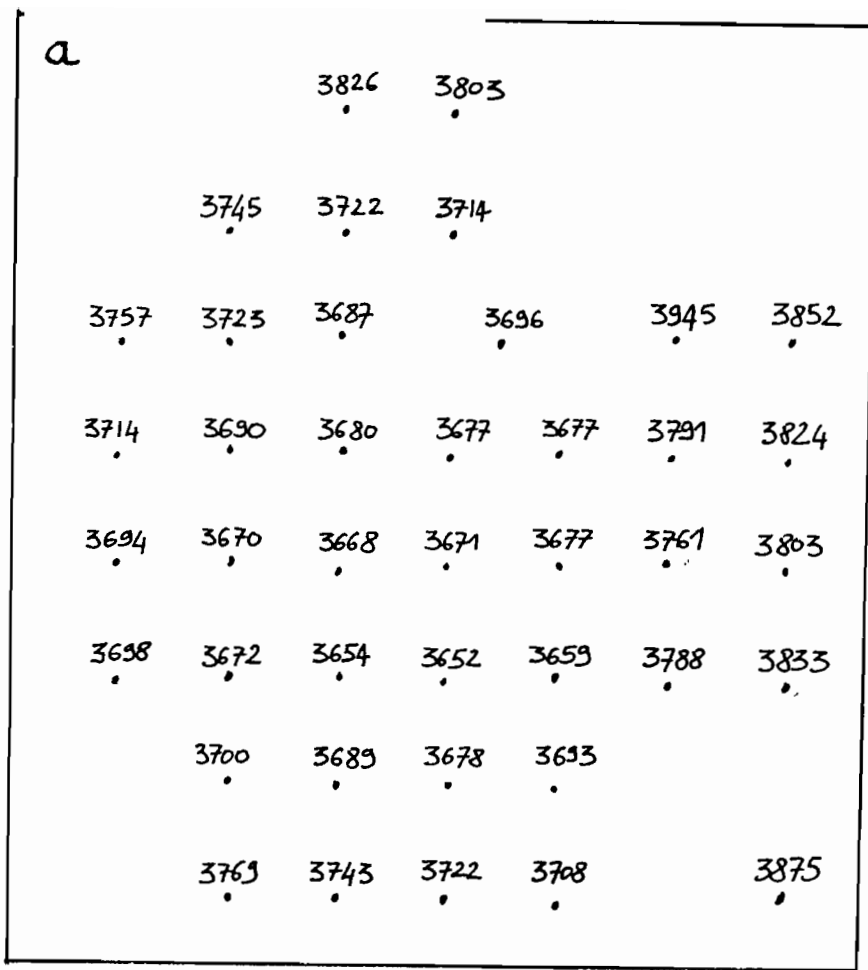
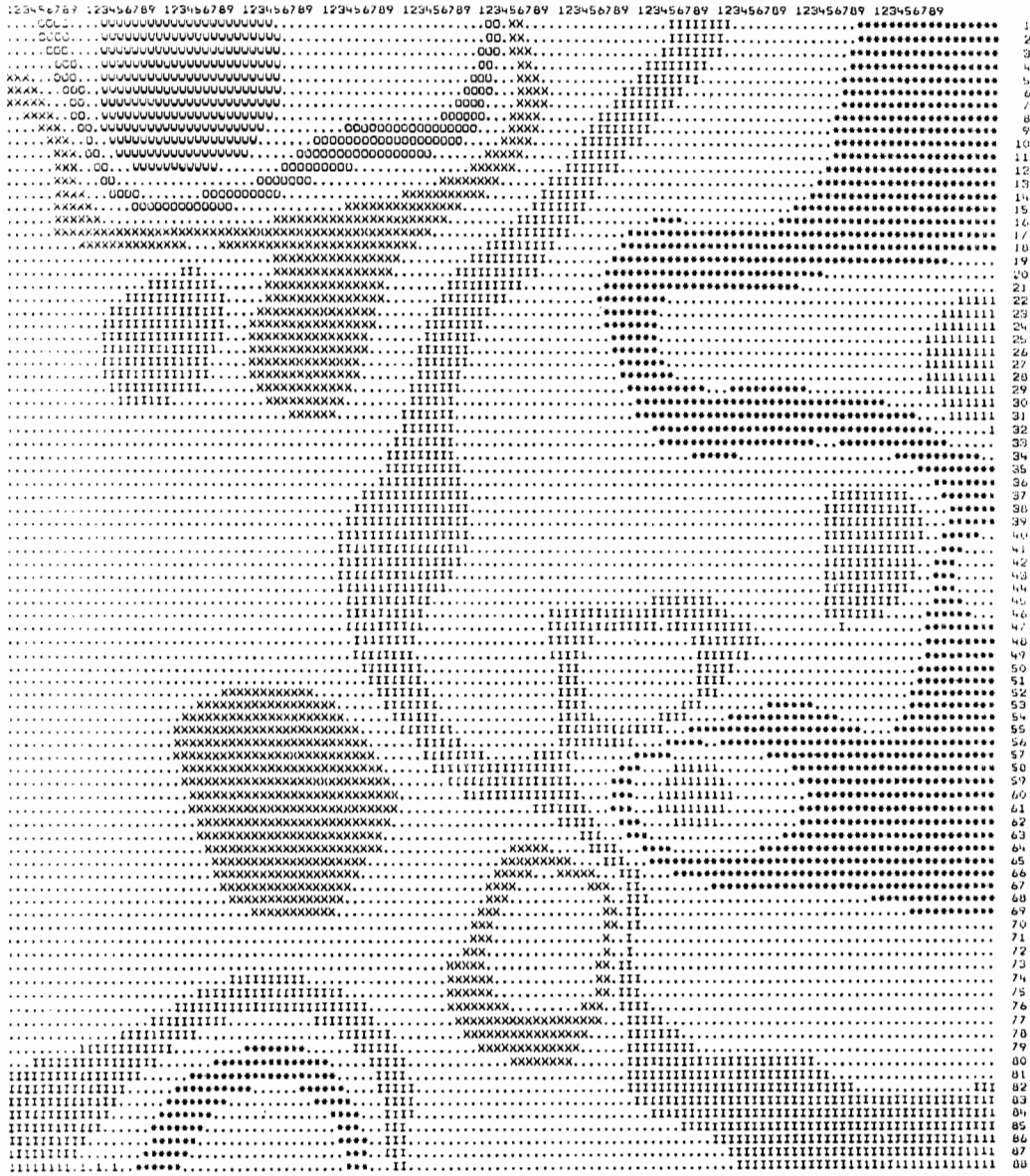


Figure 1 : Côtes de niveau repère de l'Ordovicien.
 a - Echantillons
 b - Carte d'isovaleurs : 6 itérations, pondération 75

Fichier titre : C:\Aqbasic\lmap5

Nombre de zones d'isovaleurs : 11
Nombre de points en entrée : 26
Médianisation des points de mesure : 80
Nombre d'iterations : 6



SYMBOL : I LIMITE SUPERIEURE : 10
 SYMBOL : O LIMITE SUPERIEURE : 20
 SYMBOL : X LIMITE SUPERIEURE : 40
 SYMBOL : K LIMITE SUPERIEURE : 60
 SYMBOL : I LIMITE SUPERIEURE : 80
 SYMBOL : O LIMITE SUPERIEURE : 100
 SYMBOL : X LIMITE SUPERIEURE : 120
 SYMBOL : K LIMITE SUPERIEURE : 140
 SYMBOL : O LIMITE SUPERIEURE : 160
 SYMBOL : X LIMITE SUPERIEURE : 180
 SYMBOL : K LIMITE SUPERIEURE : 200
 SYMBOL : I VALEURS SUP A : 200

Point	Abscisse	Ordonnée	Valeur	Colonne	Ligne
1	105	56	4	87	60
2	102	32	54	84	72
3	05	35	139	70	71
4	65	42	08	53	67
5	72	5	89	59	86
6	43	1	13	35	87
7	49	52	112	40	62
8	77	73	54	63	52
9	97	74	90	80	51
10	94	93	54	78	42
11	118	95	51	97	41
12	133	94	70	110	44
13	151	91	10	125	43
14	151	123	4	125	27
15	131	126	15	108	25
16	107	127	10	88	25
17	82	119	56	48	29
18	43	126	105	38	25
19	28	129	75	23	24
20	140	155	34	116	11
21	111	154	48	92	11
22	88	175	88	73	1
23	65	175	174	53	1
24	24	160	194	19	8
25	1	153	95	1	12
26	15	40	92	12	68

Figure 2 : Zones d'isovaleurs des captures de moustiques Aedes polynesiensis sur un îlot d'atoll.

3 - LE PROGRAMME INFORMATIQUE

2.1 - Description générale

Le programme proposé ici a été écrit en Quick-Basic (Microsoft) sur un compatible IBM.PC.

Il effectue les tâches suivantes :

- Création de fichiers de données spécifiques pour le programme (Saisie des données).
- Corrections d'erreurs de saisie dans un fichier de données.
- Listage (à l'écran ou à l'imprimante) des données.
- Modifications du fichier des paramètres (qui gère un certain nombre de valeurs : symboles typographiques, taille maximale du graphique etc...).
- Traçage de la carte.

Le programme a été conçu dans l'optique d'une utilisation la plus conviviale possible. L'utilisateur n'a pas besoin de connaissances en programmation : il doit simplement répondre aux questions posées en clair par le programme.

On peut déjà signaler qu'un certain nombre de paramètres sont laissés à l'appréciation des utilisateurs à chaque déroulement du programme. ce sont :

- Les dimensions du dessin
- Le nombre d'itérations (grilles d'interpolation)
- Le nombre de zones d'isovaleurs
- Les limites des zones d'isovaleurs
- L'intensité de la pondération initiale

On reviendra sur ces paramètres ultérieurement.

2.2 - Mise en route

Le programme porte le nom de MAP. Il fait appel à un fichier ASCII de paramètres nommé MAP.PAR, fournit avec MAP (cf. paragraphe 2.7). le programme MAP et son fichier de paramètres MAP.PAR doivent se trouver dans le même répertoire pour un bon fonctionnement.

le programme source est MAP.BAS, le programme compilé est MAP.EXE.

Si on désire faire fonctionner le programme source, il faut tout d'abord charger Quick-Basic, puis MAP.BAS et choisir l'option EXECUTION - DEMARRER.

Si on désire faire fonctionner la version compilée, il suffit de se positionner dans le répertoire où se trouve MAP.EXE, de taper MAP, puis de faire un 'Retour-chariot'

Une fois le programme lancé, il suffit de suivre les indications portées à l'écran.

Un premier écran apparaît (fig. 3), indiquant le titre du programme, la publication source, le nom et l'adresse du programmeur ainsi que la date de création (éventuellement la ou

TRACAGE AUTOMATIQUE DE CARTES D'ISOVALEURS

Source : DAVAUD E. - 1975 - Le traçage automatique de cartes d'isovaleurs : un algorithme simple en FORTRAN IV

Can. J. Earth Sci., 12 : 1069 - 1077

Adaptation F. Jardeux - Centre ORSTOM de Tahiti. B.P. 529 Papeete

Version Février 1990

Faites un 'retour-chariot pour continuer ...

Figure 3 : Premier écran : présentation du programme.

MENU GENERAL

- Modification du fichier des paramètres 1
- Saisie des données 2
- Correction dans un fichier de données 3
- Listage d'un fichier de données 4
- Traçage automatique de la carte 5
- Fin de la session 6

Votre choix (numéro) :

Figure 4 : Second écran : Menu général du programme MAP.

les dates de mise à jour). Il suffit de faire un 'Retour-chariot' pour passer à l'écran suivant.

Ce second écran (fig.4), est le MENU GENERAL du programme. Il permet de choisir entre les diverses grandes options (saisie des données, correction dans un fichier, listage des données, traçage, modification du fichier MAP.PAR) ou de terminer la session.

Les options proposées sont choisie en indiquant simplement le numéro voulu et en faisant un 'Retour-chariot'.

Les paragraphes suivants analysent en détail ces différentes options.

Remarque 1 : A la fin d'exécution de chaque option, le programme revient automatiquement au MENU GENERAL (sauf si l'option "Fin de la session" est choisie, bien évidemment). Lorsqu'une nouvelle option est choisie, le programme vous demande de confirmer que vous travaillez bien avec le même fichier (fig. 5), sauf, évidemment, pour l'option "Saisie des données" (= création de fichier). Si vous désirez changer de fichier, il suffit de l'indiquer au programme. Lors du premier choix, la question est, bien sur, "Nom du fichier des données " (fig .6). Il faut, dans chaque cas (premier "run" ou "run" ultérieurs), préciser le nom du fichier avec son chemin d'accès complet. Par exemple, si votre fichier est sur une disquette dans le lecteur A, il faut écrire A:\Nom-du-fichier. Si votre fichier est sur le disque dur C, dans un répertoire DATA par exemple, il faut écrire C:\DATA\Nom-du-fichier.

Remarque 2 : Lors de la création d'un fichier de données, le programme ajoute automatiquement l'extension .MAP au nom indiqué. Ceci permet de repérer plus facilement les fichiers créés et utilisables par ce programme. Le programme n'accepte pas des fichiers sans l'extension .MAP. Par contre, lorsque le programme demande le nom d'un fichier, il ne faut pas donner l'extension .MAP.

2.3 - Saisie des données - création de fichier

Cette option permet de créer des fichiers spécifiques à l'utilisation du programme MAP. Ces fichiers sont de format ASCII, séquentiels, dans lesquels les données sont stockées de la façon suivante :

Abcisse du point 1, Ordonnée du point 1, Valeur au point 1,
Abcisse du point 2, Ordonnée du point 2, etc...

La saisie des données se fait en entrant chaque valeur indiquée dans le 'tableau de saisie' (fig.7) et en faisant un 'retour-chariot' pour la valider.

Pour terminer le processus de saisie, il suffit de rentrer une valeur négative quelconque dans la colonne des abscisses et de faire des 'retour-chariot' pour les deux autres colonnes. Cette ligne ne sera pas prise en compte dans le fichier des valeurs.

Le programme demande alors de lui indiquer le nom du fichier pour ces données, en lui indiquant le chemin d'accès complet. Par exemple, C:\DATA\TYPO, si le fichier doit s'appeler TYPO et est stocké dans le disque dur C, dans un répertoire DATA.(fig.8)

Une fois la saisie terminée, le programme retourne au MENU GENERAL.

```
Le fichier de travail actuel est : c:\typo
Voulez vous le changer ? (o/n) :
```

Figure 5 : Demande de confirmation du fichier de travail en cours.

```
Nom du fichier des données (avec son chemin d'accès complet) :
```

Figure 6 : Demande du nom du fichier des données à traiter.

Pour finir la saisie, entrez une valeur négative en abscisse. Ce point ne sera pas pris en compte

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1			

Figure 7 : Tableau de saisie des données : couples (x,y) et valeur (z) aux coordonnées (x,y)

Pour finir la saisie, entrez une valeur négative en abscisse. Ce point ne sera pas pris en compte

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1	20	40	125
2	25	36	150
3	15	47	250
4	12	25	251
5	25	62	136
6	56	12	99
7	-6		

Nom du fichier avec son chemin d'accès complet : c:\data\typo

Figure 8 : Exemple de données fictives saisies et stockées dans le fichier TYPO (dans le disque dur C, sous le répertoire DATA)

Corrections dans le fichier : c:\typo

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1	20	40	125
2	25	36	150
3	15	47	250
4	12	25	251
5	25	62	136
6	56	12	99

Voulez vous corriger ces données (o/n) :

Figure 9 : Exemple d'écran permettant la correction des données.

Corrections dans le fichier : c:\typo

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1	20	40	125
2	25	36	150
3	15	47	250
4	12	25	251
5	25	62	136
6	56	12	99

Voulez vous corriger ces données (o/n) : o
 Numéro du point à corriger : 4
 Son abscisse est : 12 Nouvelle abscisse : 12
 Son ordonnée est : 25 Nouvelle ordonnée : 25
 Sa valeur est : 251 Nouvelle valeur : 287

Figure 10 : Exemple de correction de la donnée z (=valeur du point de coordonnées x, y)

Corrections dans le fichier : C:\TYPO

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1	20	40	125
2	25	36	150
3	15	47	250
4	12	25	287
5	25	62	136
6	56	12	99

Voulez vous corriger ces données (o/n) :

Figure 11 : Exemple de réaffichage de la partie du fichier corrigé.

FICHER : C:\TYPO

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1	20	40	125
2	25	36	150
3	15	47	250
4	12	25	287
5	25	62	136
6	56	12	99

Appuyez sur une touche pour continuer

Voulez vous l'impression sur imprimante (o/n) :

Figure 12 : Listage, à l'écran, du fichier des données.

FICHER : C:\TYPO

POINT N°	ABSCISSE (X)	ORDONNEE (Y)	VALEUR
1	20	40	125
2	25	36	150
3	15	47	250
4	12	25	287
5	25	62	136
6	56	12	99

Figure 13 : Listage, à l'imprimante, du fichier des données.

Dans cette rubrique, il n'y a pas de possibilités de corriger une valeur mal saisie. Au cas où une erreur dans la saisie se produirait, il faut tout de même finir de rentrer les autres données, comme si tout était correct et, lorsque le fichier est créé, passer à l'option de "CORRECTION" du MENU GENERAL (cf. paragraphe suivant).

2.4 - Corrections dans un fichier

L'option de correction permet de modifier des données déjà stockées dans un fichier.

Après avoir répondu à la question "Nom du fichier", le programme liste celui-ci à l'écran. Pour chaque écran affiché (c'est à dire pour chaque partie de fichier listée), le programme vous demande si des corrections sont à faire (fig.9).

Si la réponse est n (=non), le programme passe à un écran suivant en affichant la suite des données. Dans le cas contraire, le programme demande le numéro du point à corriger. Le programme donne alors la valeur de l'abscisse de ce point et demande la nouvelle valeur. Si l'abscisse n'est pas à corriger, on doit retaper tout de même la valeur. Le programme répète ce processus pour l'ordonnée et la "valeur" du point (fig.10). Une fois l'erreur (ou les erreurs) corrigée, le programme réaffiche l'écran corrigé et redemande si d'autres corrections sont à faire (fig.11). Le processus est répété jusqu'à ce qu'il n'y ait plus aucune correction à faire (répondre "n" à la question "Voulez vous corriger ces données (o/n) : ").

Le programme affiche alors la suite des données et réitère sa demande de correction. Ce processus est répété jusqu'à ce que tout le fichier ait été passé en revue.

Le programme sauvegarde alors les corrections dans un fichier de même nom (l'ancien fichier est détruit et remplacé par le fichier corrigé). Le programme retourne ensuite au MENU GENERAL.

2.5 - Listage d'un fichier

Cette option permet de lister un fichier à l'écran ou sur l'imprimante. Après avoir donné le nom du fichier à lister, le programme liste celui-ci à l'écran. En fin de fichier, le programme demande si le listage doit être reproduit sur l'imprimante (fig.12). En cas de réponse affirmative (o = oui), la sortie sur imprimante se fait comme à la figure 13.

2.6 - Traçage des cartes d'isovaleurs

Après avoir répondu à la question du "nom du fichier à traiter", le programme pose un certain nombre de questions préalables aux calculs (fig.14)

- 1 - "Nombre de zones d'isovaleurs : "

C'est le nombre de zones d'isovaleurs souhaitées, selon lesquelles sera dessinée la carte. Le maximum possible est de 19.

- 2 - "Dimension horizontale du dessin : "

Elle est exprimée par le nombre de colonnes requises. Les limites de Quick-Basic imposent un maximum de 125 (au delà, il y a un dépassement de la capacité mémoire en raison des

déclarations de variables qui dépendent de cette valeur). Ce nombre de colonnes est aussi limité par le fichier MAP.PAR (cf paragraphe 2.7 suivant), dans lequel les dimensions de l'imprimante sont demandées (80 colonnes, 132 colonnes ...). Si l'imprimante disponible est de 80 colonnes, la taille maximale pour la dimension horizontale du dessin tombe à 76 colonnes.

La précision de la carte est d'autant plus grande que sa dimension est élevée. La durée des calculs est cependant d'autant plus longue que cette dimension est élevée.

La qualité de la résolution est limitée par l'utilisation de l'imprimante. Elle pourrait être améliorée par l'utilisation d'un traceur de courbes ou en subdivisant le problème étudié en aires distinctes, juxtaposées, et en soumettant séparément chaque groupe de données.

- 3 - *"Nombre de grilles d'interpolation successives : "*

Ce paramètre permet de fixer indirectement les dimensions de la première grille d'interpolation. Le programme affiche d'ailleurs la taille de la maille initiale (en nombre de colonnes), compte tenu de la réponse fournie à la question. Si cette maille convient, il suffit de répondre o (= oui). Dans le cas contraire (n = non), le programme repose la question "Nombre de grilles d'interpolation ...".

Par exemple, si le nombre de grilles est de 6, la maille initiale aura $2^{(6-1)} = 32$ colonnes de côté.

- 4 - *"Intensité de la pondération des points de mesure : "*

Si la valeur attribuée à ce paramètre est élevée, la surface tend à passer par tous les points de mesure. Par contre, lorsque cette valeur décroît, la surface devient plus émoussée. L'émoussage est maximum pour une valeur de 1.

- 5 - *"Limites de classes de valeurs : "*

On doit entrer ces limites (supérieures), dans l'ordre croissant, de la plus petite à la plus grande.

Le programme procède alors aux calculs et affiche les numéros des grilles sur lesquelles il procède : "Je fais les calculs sur la grille n° ...".

A la fin des calculs, la carte est automatiquement reproduite sur l'imprimante. Donc, **NE PAS OUBLIER D'ALLUMER L'IMPRIMANTE !**

Comme le souligne Davaud, l'algorithme MAP peut, sans autre modification, servir soit à la cartographie automatique, soit à l'analyse de tendance. En faisant varier l'intensité de pondération, l'utilisateur peut contrôler le degré d'émoussage de la surface. En jouant sur le nombre d'itérations, il peut encore limiter l'interpolation aux zones où la densité de points de mesure est suffisamment élevée ou au contraire faire en sorte que la surface occupe entièrement l'espace délimité par les points de mesures extrêmes.

2.7 - Le fichier des paramètres (MAP.PAR)

Ce fichier contient un certain nombre de paramètres indispensables au bon déroulement du programme. Il sont fixés


```

Nombre de zones d'isovaleurs (max = 19) : 12

Dimension horizontale du dessin
Cette dimension ne peut pas être supérieure à 125 (nombre de colonnes) : 125

Nombre de grilles d'interpolation successives souhaitées : 6
La maille initiale aura 32 colonnes de côté
D'accord pour cette valeur (o/n) : 0

Intensité de la pondération des points de mesure : 75

Limites des classes de valeur :
De la plus petite à la plus grande, dans l'ordre croissant ....

Limite supérieure de la 1 ème classe : 50
Limite supérieure de la 2 ème classe : 75
Limite supérieure de la 3 ème classe : 80
Limite supérieure de la 4 ème classe : 150
Limite supérieure de la 5 ème classe : 160
Limite supérieure de la 6 ème classe : 170
Limite supérieure de la 7 ème classe : 180
Limite supérieure de la 8 ème classe : 200
Limite supérieure de la 9 ème classe : 220
Limite supérieure de la 10 ème classe : 250
Limite supérieure de la 11 ème classe : 300
Limite supérieure de la 12 ème classe : 350

Je fais les calculs sur la grille n° 1

```

Figure 14 : Questions posées par le programme à chaque lancement de MAP.

```

MODIFICATIONS DU FICHER DES PARAMETRES (MAP.PAR)

Symboles utilisés pour le traçage :

n°  Symbole      n°  Symbole      n°  Symbole      n°  Symbole
1    1            2    .            3    *            4    .
5    I           6    .            7    X           8    .
9    0           10   .           11   W           12   .
13   $           14   .           15   5           16   6
17   7           18   8           19   9           20   $

Voulez vous modifier ces symboles (o/n) :

```

Figure 15 : Exemple de symboles typographiques, avec la possibilité de modifications.

selon les désirs des utilisateurs, mais, pour un même type d'installation, ces paramètres n'ont pas à être changés à chaque démarrage de MAP... d'où leur présence sur un fichier à part.

Toutefois, ces paramètres peuvent être modifiés (option 1 du MENU GENERAL). Dans ce cas, le programme demande les questions suivantes :

- 1 - "Nombre de colonnes de l'imprimante : "
- 2 - "Definition des échelles horizontales et verticales"
- 3 - "Symboles graphiques"

A chaque question, le programme affiche la (ou les) valeur par défaut et demande si on désire la modifier. En cas de réponse négative, le programme passe à la question suivante. Si la valeur doit être modifiée, le programme demande la nouvelle valeur et réaffiche la correction.

Par exemple, dans le cas des symboles typographiques pour la cartographie, un premier écran apparaît (fig.15) où l'ensemble des valeurs par défaut est affiché, avec la question "Voulez vous modifier ces symboles (o/n) : ". Si la réponse est o (oui), le programme demande le "Numéro du symbole à corriger : ". Il affiche ensuite le symbole correspondant et demande le nouveau symbole désiré (fig. 16). Le programme réaffiche alors les symboles, en tenant compte de la correction (fig. 17). Les corrections prennent fin si on répond n (non) à la question "Voulez vous modifier ces symboles (o/n) : ".

Remarque : Dans le fichier MAP.PAR originel, la configuration de l'imprimante est fixée, par défaut, à 132 colonnes. De même, la définition des échelles horizontales et verticales suppose que 3 lignes représentent la même distance que 5 colonnes, en raison de la configuration des caractères de l'imprimante utilisée pour la mise au point du programme (ici, une Mannesman Tally 290). Pour la plupart des imprimantes matricielles du marché, ces paramètres sont en général identiques à ceux de MAP.PAR.

Les symboles peuvent être modifiés à souhait, en prenant par exemple ceux des caractères ASCII. Seul le caractère blanc n'est pas autorisé.

CONCLUSION

Le programme présenté ici est relativement souple d'emploi et modulable en fonction des besoins. Il peut répondre simplement à la plupart des problèmes de cartographie automatique de zones d'isovaleurs. Les représentations graphiques obtenues sont satisfaisantes, quelle que soit la complexité de la surface et quelles que soient la distribution et la densité des points de mesure.

Le programme peut être obtenu sur disquette, ainsi qu'une version compilée pour ceux qui ne possèdent pas Quick-Basic.

MODIFICATIONS DU FICHIER DES PARAMETRES (MAP.PAR)

Symboles utilisés pour le traçage :

n°	Symbole	n°	Symbole	n°	Symbole	n°	Symbole
1	1	2	.	3	*	4	.
5	I	6	.	7	X	8	.
9	0	10	.	11	W	12	.
13	\$	14	.	15	5	16	6
17	7	18	8	19	9	20	\$

Numéro du symbole à corriger : 15
Ce symbole est : 5
Nouveau symbole : !

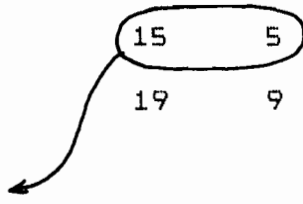


Figure 16 : Exemple de modification d'un symbole.

MODIFICATIONS DU FICHIER DES PARAMETRES (MAP.PAR)

Symboles utilisés pour le traçage :

n°	Symbole	n°	Symbole	n°	Symbole	n°	Symbole
1	1	2	.	3	*	4	.
5	I	6	.	7	X	8	.
9	0	10	.	11	W	12	.
13	\$	14	.	15	!	16	6
17	7	18	8	19	9	20	\$

Voulez vous modifier ces symboles (o/n) :

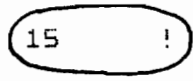


Figure 17 : Réaffichage des symboles modifiés.

BIBLIOGRAPHIE

- Davaud, E. - 1975 - Le traçage automatique de cartes d'isovaleurs: un algorithme simple en FORTRAN IV.
Can. J. Earth Sci., 12 : 1069 - 1077.
- Lardeux, F., Séchan, Y., Faaruaia, M., Tuhiti, P., - 1988 - Lutte biologique contre *Aedes polynesiensis* avec le copépode *Mesocyclops aspericornis*. Expérimentation à Rangiroa. III - Evaluation six mois après traitement.
Rapport ITRMLM 15/88/ITRM/Doc-Ent: 45 p.
- Matheron, - 1965 - *Les variables régionalisées et leur estimation*.
Masson et Cie (Eds), Paris.

Annexe : Listing du programme MAP

```

' programme MAP
' tracage automatique de cartes d'isovaleurs
' F. LARDEUX    février 1990
'
-----
clear
cls
print
print tab(20) "TRACAGE AUTOMATIQUE DE CARTES D'ISOVALEURS"
print : print : print
print "Source : DAVAUD E. - 1975 - Le traçage automatique de cartes "
print "                                d'isovaleurs : un algorithme simple "
print "                                en FORTRAN IV"
print
print "                                Can. J. Earth Sci., 12 : 1069 - 1077"
print
print
print "Adaptation F. lardeux - Centre ORSTOM de Tahiti. B.P. 529 Papeete"
print
print "Version Février 1990"
print : print : print : print:
input "Faites un 'retour-chariot pour continuer ...",r$
'++++++
defint i,n
defdbl f,f
dim LIM!(20)
dim U(200), V(200), W(200), lig(200), co(200)
dim F(200)

gosub lecdef : 'lecture des paramètres de dimensionnement par défaut

dim A(nbcol-6,nbcol-6)

'++++++programme principal    menu général    ++++++
arreter = 1
while arreter
cls
print : print : print
print tab(35) "MENU GENERAL"
print : print
print tab(15) "Modification du fichier des paramètres ..... 1"
print tab(15) "Saisie des données ..... 2"
print tab(15) "Correction dans un fichier de données ..... 3"
print tab(15) "Listage d'un fichier de données ..... 4"
print tab(15) "Traçage automatique de la carte ..... 5"
print
print tab(15) "Fin de la session ..... 6"
print : print : print
while r < 1 or r > 6
    print tab(25) : input "Votre choix (numéro) : ",r
wend
if r = 1 then gosub modif:ERASE A: gosub lecdef: DIM A(nbcol-6,nbcol-6)
if r = 2 then gosub saisie

```

```

    if r = 3 then gosub correc
    if r = 4 then gosub liste
    if r = 5 then gosub trace
    if r = 6 then end
    r = 0
wend
end
'+++++ fin du programme principal +++++
'+++++ lecture des paramètres de dimensionnement par défaut ++++
lecdef:
dim sym$(20) : '20 symboles pour le traçage des cartes
cls
open "i",#1,"map.par"
input #1, nbc, k1, kc, secteur
i = 1
while not eof(1)
    input #1, sym$(i)
    i = i + 1
wend
close #1
return
'+++++ fin de lecdef +++++
'+++++ saisie des paramètres utilisateur pour lancer le programme
trace:
call nomfich(fichier$) : 'demande le nom du fichier
cls
NCLAS = 0
while NCLAS < 1 or NCLAS > 19
    print
    input "Nombre de zones d'isovaleurs (max = 19) : ", NCLAS
wend

NC = 0
while NC < 1 or NC > NBCOL - 7
    print
    print "Dimension horizontale du dessin"
    print "Cette dimension ne peut pas être supérieure à ";nbc-7;
    input " (nombre de colonnes) : ",NC
wend

R$ = "N"
while R$ = "N" or R$ = "n"
    print
    input "Nombre de grilles d'interpolation successives souhaitées : ",ITERA
    print "La maille initiale aura ";2^(ITERA - 1); "colonnes de côté"
    Y$ = ""
    while Y$ <>"0" and Y$ <>"o" and Y$ <>"N" and Y$ <>"n"
        input "D'accord pour cette valeur (o/n) : ",Y$
        if Y$ = "0" or Y$ = "o" then R$ = "0"
    wend
wend

```

```

print
NWEIGHT = 0
while NWEIGHT < 1
  input "Intensité de la pondération des points de mesure : ",NWEIGHT
wend

PRINT
print "Limites des classes de valeur :"
print "De la plus petite à la plus grande, dans l'ordre croissant ...."
print
for K = 1 to NCLAS
  WHILE LIM!(k-1) >= LIM!(k)
    print "Limite supérieure de la ";K;" ème classe : ";
    input "",LIM!(K)
  wend
next K
print : print

'+++++ sous programme de calcul et traçage +++++
XMIN = 1E30
YMIN = XMIN
ZMIN = XMIN

XMAX = -1E30
YMAX = XMAX
ZMAX = XMAX

gosub lecdon

DZ = ZMAX - ZMIN
for I = 1 to nc+1
  for J = 1 to nc+1
    A(I,J) = 0
  next J
next I

'*****      CALCUL DES DIMENSIONS DE LA MAILLE D'INTERPOLATION      *****
DX = (XMAX - XMIN) / NC
DY = KC * DX / KL
NL = ((YMAX - YMIN) / DY + 1)
for I = 1 to NU
  IX = ((U(I) - XMIN) / DX)
  lig(i) = INT(ix)
  if lig(i) = 0 then lig(i) = 1
  IY = ((V(I) - YMIN) / DY)
  co(i) = INT(iy)
  if co(i) = 0 then co(i) = 1
  A(NL-IY,IX+1) = W(I) + DZ * 2
next I
NC = NC + 1

```



```

*****          CREATION SUCCESSIVES DE GRILLES D'INTERPOLATION          *****
for KPA = 1 to ITERA
  print "Je fais les calculs sur la grille n° ";kpa
  JMP = 2^(ITERA-KPA)
  DIST = (KL * DY * JMP) ^2
  NW = 0
  open "D",#1,"TEMPO.MAP"
  for ILIG = 1 to NL step JMP
    NDY = ILIG - KL * JMP
    NEY = ILIG + KL * JMP
    if NDY < 1 then NDY = 1
    if NEY > NL then NEY = NL
    for ICOL = 1 to NC step JMP
      if ((A(ILIG, ICOL) > ZMAX + DZ) and (KPA > 1)) then goto 75
      NDX = ICOL - KC * JMP
      NEX = ICOL + KC * JMP
      if NEX > NC then NEX = NC
      if NDX < 1 then NDX = 1
      NPOIN = 0
      for L = NDY to NEY
        for K = NDX to NEX
          if A(L,K) = 0 then goto 77
          D = ((ICOL - K) * DX)^2 + ((ILIG - L) * DY)^2
          if D > DIST then goto 77
          NPOIN = NPOIN + 1
          U(NPOIN) = K * DX
          V(NPOIN) = L * DY
          W(NPOIN) = A(L,K)
          ME = SECTEUR - (SECTEUR * D / DIST) + 1
          F(NPOIN) = ME * ME
          if A(L,K) <= ZMAX then goto 77
          W(NPOIN) = A(L,K) - 2 * DZ
          F(NPOIN) = F(NPOIN) * NWEIGHT
77      next K
        next L
      if NPOIN < 4 goto 75
      NW = NW + 1

*****          CALCUL DES COEFFICIFN1S DU POLYNOME          *****
SX = 0
SY = 0
SZ = 0
SXY= 0
SXZ= 0
SYZ= 0
SXX= 0
SYY= 0
NP# = 0
for K = 1 to NPOIN
  NP# = NP# + F(K)
  SX = SX + F(K) * U(K)
  SY = SY + F(K) * V(K)
  SZ = BZ + F(K) * W(K)

```

```

next K
XM = SX / NP#
YM = SY / NP#
ZM = SZ / NP#
for K = 1 to NPOIN
  SXX = SXX + (F(K) * (U(K) - XM))^2
  SYY = SYY + (F(K) * (V(K) - YM))^2
  SXY = SXY + F(K) * (U(K) - XM) * (V(K) - YM)
  SXZ = SXZ + F(K) * (U(K) - XM) * (W(K) - ZM)
  SYZ = SYZ + F(K) * (V(K) - YM) * (W(K) - ZM)
next K

SKK = SXX * SYY - SXY^2
if SKK = 0 then goto 87
B1 = (SXZ * SYY - SYZ * SXY) / SKK
B2 = (SYZ * SXX - SXZ * SXY) / SKK
B0 = ZM - B1 * XM - B2 * YM
X0 = ICOL * DX
Y0 = ILIG * DY
Z = B0 + B1 * X0 + B2 * Y0
print #1, ICOL, ILIG, Z
goto 75
87      print #1, ICOL, ILIG, ZM
75      next ICOL
        next ILIG
        close #1

'*****      ATTRIBUTION DES SYMBOLES ET IMPRESSION DE LA SURFACE      *****
open "I", #1, "TEMPO.MAP"
for I = 1 to NW
  input #1, K, L, B
  A(L, K) = B
next I
close #1
next KPA

WIDTH "LPT1:", nbc1
LPRINT
lprint "Fichier traité : "; fichier$
lprint
lprint "Nombre de zones d'isovaleurs      : "; nclas
lprint "Nombre de points en entrée          : "; NU
lprint "Pondération des points de mesure    : "; nweight
lprint "Nombre d'itérations                  : "; itera
lprint:lprint
Z$ = "123456789 "
FOR I = 1 TO int(nc/10)
  lprint Z$;
next i
lprint

for L = 1 to NL
  for K = 1 to NC
    if A(L, K) > ZMAX + DZ then A(L, K) = A(L, K) - 2 * DZ

```

```

        for J = 1 to NCLAS
            if A(L,K) > LIM!(J) then goto 86
            A(L,K) = ASC(sym$(J))
            goto 84
86      if j = nclas then A(1,k) = ASC(sym$(nclas+1))
        next J
84 next K
    for K = 1 to NC
        lprint CHR$(A(L,K));
    next K
    lprint USING "####"; L
next L

'*****          IMPRESSION DE LA LEGENDE          *****
lprint: lprint
for i = 1 to nclas
    lprint "SYMBOLE : ";SYM$(i);
    lprint " LIMITE SUPERIEURE : ";LIM!(i)
next i
lprint "SYMBOLE : ";sym$(nclas+1);
lprint " VALEURS SUP A      : ";LIM!(nclas)
lprint
lprint "-----"
lprint
lprint " Point          Abscisse          Ordonnée          Valeur          Colonne          Ligne"
gosub lecdon
for i = 1 to nu
    lprint tab(3) i;
    lprint tab(18) U(i);
    lprint tab(31) V(i);
    lprint tab(43) W(i);
    lprint tab(54) lig(i);
    lprint tab(65) NL - co(i)
next i

KILL "TEMPO.MAP"

end
'+++++++ fin de trace ++++++
'+++++ sous programme de lecture des données sur fichier +++++
'+++++
lecdon:
open "I",#1,fichier$ + ".map"
NU = 1
while not EOF(1)
    input #1,U(NU), V(NU), W(NU)
    if U(NU) > XMAX then XMAX = U(NU)
    if U(NU) < XMIN then XMIN = U(NU)
    if V(NU) > YMAX then YMAX = V(NU)
    if V(NU) < YMIN then YMIN = V(NU)
    if W(NU) > ZMAX then ZMAX = W(NU)
    if W(NU) < ZMIN then ZMIN = W(NU)

```

```

    NU = NU + 1
wend
NU = NU - 1
close #1
return
'+++++++ fin de lecdon ++++++

'+++++++
'+++++++ sous programme de saisie des données ++++++
'+++++++
saisie:
cls
print : print
PRINT tab(15) "Pour finir la saisie, entrez une valeur négative"
print tab(15) "en abscisse. Ce point ne sera pas pris en compte "
print
print TAB(10) " POINT N°          ABSCISSE          ORDONNEE          VALEUR"
print tab(10) "                  (X)                  (Y)"
print tab(10) "-----"
VIEW PRINT 9 TO 24
I = 1
arret = 1
while U(i-1) >= 0
  LOCATE ,13 : print i;
  LOCATE ,28 : input ;"" ,U(I)
  LOCATE ,47 : input ;"" ,V(I)
  LOCATE ,63 :input "" ,W(I)
  I = I + 1
wend
NBPOIN = I - 2
print : print
input "Nom du fichier avec son chemin d'accès complet : ",fichier$
VIEW PRINT
gosub ecrdon
return
'+++++++ fin de saisie ++++++

'+++++++
'+++++++ SOUS PROGRAMME D'ECRITURE DES DONNEES SUR FICHER ++++++
'+++++++
ecrdon:
OPEN "O",#1,FICHER$ + ".MAP"
FOR I = 1 TO NBPOIN
  PRINT #1,U(I), V(I), W(I)
NEXT I
CLOSE #1
RETURN
'+++++++ fin de ecrdon ++++++

'+++++++
'+++++++ sous programme de modification des paramètres ++++++
'+++++++
modif:
gosub lecdef : 'lecture des paramètres par défaut dans le fichier MAP.PAR

```

```

cls
print : print
print tab(15) "MODIFICATIONS DU FICHIER DES PARAMETRES (MAP.PAR)
print : print
view print 6 to 24

areter = 1
while areter
  cls
  print "Le nombre de colonnes de l'imprimante est : ";nbc01
  print
  y$ = ""
  while (y$ <>"0" and y$ <>"o" and y$ <>"N" and y$ <>"n")
    input "Voulez vous corriger cette valeur (o/n) : ",y$
  wend
  if y$="o" or y$ = "0" then
    print
    input "Nombre de colonnes de l'imprimante : ",nbc01
  else areter = 0
  end if
wend

areter = 1
while areter
  cls
  print
  print "Définition des échelles horizontales et verticales :"
  print
  print "Actuellement, ";KL;" lignes représentent la même distance que "
  print "          ";KC;" colonnes."
  print
  y$ = ""
  while (y$ <>"0" and y$ <>"o" and y$ <>"N" and y$ <>"n")
    input "Voulez vous corriger ces valeurs (o/n) : ",y$
  wend
  if y$="o" or y$ = "0" then
    print
    input "Nombre de lignes   : ",KL
    input "Nombre de colonnes : ",KC
    print
    print "Maintenant, ";KL;" lignes représentent la même distance que "
    print "          ";KC;" colonnes."
  else areter = 0
  end if
wend

areter = 1
while areter
  cls
  print "Valeur du paramètre 'secteur' : ";secteur
  print
  y$ = ""
  while (y$ <>"0" and y$ <>"o" and y$ <>"N" and y$ <>"n")
    input "Voulez vous modifier cette valeur (o/n) : ",y$
  wend
end while

```

```

wend
if y$="o" or y$ = "O" then
  print
  input "Secteur : ",secteur
  else areter = 0
end if
wend

areter = 1
while areter
  view print 6 to 24
  cls
  print "Symboles utilisés pour le traçage :"
  print
  print " n° Symbole          n° Symbole          n° Symbole          n° Symbole"
  print
  for i = 1 to 17 step 4
    locate , 1 : print i; : locate ,9 : print sym$(i);
    locate ,21 : print i+1; : locate , 29 : print sym$(i+1);
    locate ,41 : print i+2; : locate , 49 : print sym$(i+2);
    locate ,58 : print i+3; : locate , 66 : print sym$(i+3)
  print
  next i
  view print 20 to 24
  y$ = ""
  while (y$ <>"O" and y$ <>"o" and y$ <>"N" and y$ <>"n")
    input "Voulez vous modifier ces symboles (o/n) : ",y$
  wend
  if y$="o" or y$ = "O" then
    print
    ns = 0
    while ns<1 or ns>20
      cls
      input "Numéro du symbole à corriger : ",ns
      ancien$ = sym$(ns)
      print "Ce symbole est                : ";sym$(ns)
      input "Nouveau symbole              : ",sym$(ns)
      if sym$(ns) = "" then
        sym$(ns) = ancien$
        print "ATTENTION : ce symbole n'est pas valide !"
        input "Appuyez sur une touche pour continuer ...",a$
        ns = 0
      end if
    wend
  else areter=0
  end if
wend
'***** écriture des paramètres de dimensionnement par défaut
open "o",#1,"MAP.PAR"
print #1, nbc1, k1, kc, secteur
for i = 1 to 20

```

```

    print #1, sym$(i)
next i
close #1
view print
return
'+++++++ fin de modif ++++++'

'+++++++ sous programme de listage ++++++'
'+++++++ liste:
call nomfich(fichier$) : 'demande le nom du fichier à lister
gosub lecdon           : 'lecture des données du fichier

close #2
open "scrn:" for output as #2
ecr=0

impress:
cls
print #2, ""
print #2, "FICHER : "; fichier$
print #2, ""
print #2, tab(10) " POINT N°          ABSCISSE          ORDONNEE          VALEUR"
print #2, tab(10) "                  (X)                  (Y)"
print #2, tab(10) "-----"
VIEW PRINT 7 TO 24
debut = 1
fin = debut + 14

arret = 1
while arret
  cls
  if nu <= fin then fin = nu : arret = 0
  for i = debut to fin
    print #2, tab(13); : print #2, i;
    print #2, tab(28); : print #2, U(I);
    print #2, tab(47); : print #2, V(I);
    print #2, tab(63); : print #2, W(I)
  next i
  if ecr = 0 then
    print
    input "Appuyez sur une touche pour continuer .... ", a$
  end if
  debut = fin + 1
  fin = fin + 14
wend
close #2
print
print
if ecr = 1 then view print : return
y$=""
while (y$ <> "o" and y$ <> "0" and y$ <> "n" and y$ <> "N")

```

```
        input "Voulez vous l'impression sur imprimante (o/n) : ",y$
wend
if y$="N" or y$="n" then
    ecr = 0
    view print
    return
end if
open "lpt1:" for output as #2
ecr = 1
goto impress:
return
'+++++++ fin de liste ++++++
'+++++++ sous programme de demande de nom de fichier ++++++
'+++++++
sub nomfich(f$) static
cls
for i = 1 to 10 : print : next i
if f$ <> "" then
    print "Le fichier de travail actuel est : ";f$
    print
    y$ = ""
    while (y$ <>"0" and y$ <>"o" and y$ <>"n" and y$ <> "N")
        input "Voulez vous le changer ? (o/n) : ",y$
    wend
    if y$ = "o" or y$ = "0" then f$ = "" : call nomfich(f$)
    else
        while f$ = ""
            print "Nom du fichier des données (avec son chemin d'accès complet) : "
            print
            input "",f$
        wend
    end if
end sub
'+++++++ fin du sous programme de nom de fichier ++++++
'+++++++ sous programme de correction des données ++++++
correc:
call nomfich(fichier$) : 'demande le nom du fichier à corriger
gosub lecdon : 'lit les données sur fichier
cls
print:print
print tab(15) "Corrections dans le fichier : ";fichier$
print:print
print tab(10) " POINT N°          ABSCISSE          ORDONNEE          VALEUR"
print tab(10) "                  (X)                  (Y)"
print tab(10) "-----"
VIEW PRINT 9 TO 24

debut = 1
fin = debut + 9

arret = 1
```



```

while arret
  cls
  if nu <= fin then fin = nu : arret = 0
  for i = debut to fin
    print tab(13); : print i;
    print tab(28); : print U(I);
    print tab(47);: print V(I);
    print tab(63);: print W(I)
  next i
  y$=""
  while (y$<>"o" and y$<>"0" and y$ <> "n" and y$ <> "N")
    print
    input "Voulez vous corriger ces données (o/n) : ",y$
  wend
  if y$="N" or y$="n" then
    debut = fin + 1
    fin = fin + 9
    if nu <= fin then fin = nu
    goto 95
  else
    input "Numéro du point à corriger : ",cor
    print "Son abscisse est : ";u(cor);:input " Nouvelle abscisse : ",u(cor)
    print "Son ordonnée est : ";v(cor);:input " Nouvelle ordonnée : ",v(cor)
    print "Sa valeur est : ";w(cor);:input " Nouvelle valeur : ",w(cor)
  end if
95 wend
NBPOIN = NU
GOSUB ECRDON
view print
return
'+++++ fin de correc +++++

```