

UNIVERSITÉ DE NANCY

N° 388

Sc. N. 71
5A

**RECHERCHE DE RÈGLES D'ORDONNANCEMENT
PAR TRAITEMENT SUR ORDINATEUR**



THÈSE

présentée à la

DE L'UNIVERSITÉ DE NANCY

pour obtenir le grade de

DOCTEUR DE SPÉCIALITÉ (MATHÉMATIQUES)

par

Jean-Marie PROTH

soutenue en Janvier 1971
devant la Commission d'Examen

25/1/71

JURY :

J. LEGRAS, *Président*
M. DEPAIX, *Examineur*
M. PAIR, *Examineur*

RECHERCHE DE RÈGLES D'ORDONNANCEMENT PAR TRAITEMENT SUR ORDINATEUR



THÈSE

présentée à la

FACULTÉ DES SCIENCES
DE L'UNIVERSITÉ DE NANCY

pour obtenir le grade de

DOCTEUR DE SPÉCIALITÉ (MATHÉMATIQUES)

par

Jean-Marie PROTH

soutenue en Janvier 1971
devant la Commission d'Examen

JURY :

J. LEGRAS, *Président*
M. DEPAIX, *Examineur*
M. PAIR, *Examineur*

Ce travail a été réalisé sous la direction du Monsieur le Professeur LEGRAS, Directeur du Centre Universitaire de Calcul Automatique ; qu'il soit remercié ici pour le temps qu'il a bien voulu me consacrer ainsi que pour les conseils éclairés qu'il m'a prodigués. Que soient également remerciés Messieurs PAIR et DEPAIX qui ont accepté de participer au jury.

Enfin, j'exprime ma profonde reconnaissance aux Chercheurs du Centre de Calcul responsables du projet SIVA et dont les travaux ont facilité l'exposé qui suit, ainsi qu'aux Ingénieurs des Aciéries de Pompey, qui m'ont fait profiter de leur expérience.

Ma gratitude va aussi au Personnel Technique et à Mademoiselle Gigoux qui se sont aimablement chargés de la réalisation technique de ce fascicule.

CHAPITRE I

POSITION DU PROBLEME ET DEFINITIONS

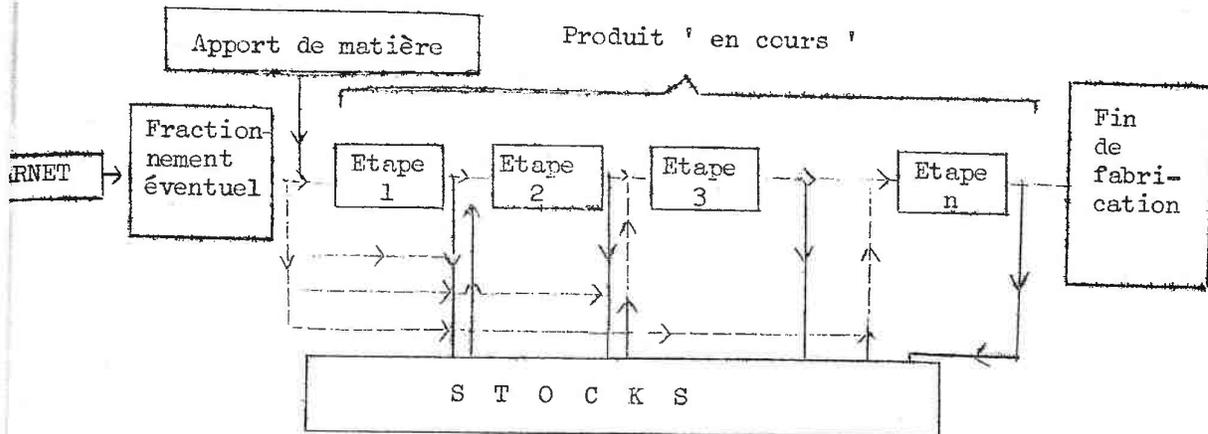
I - INTRODUCTION

Le problème qui suit trouve son origine dans la fabrication en aciéries. Partant d'une certaine quantité de matière, on lui fait subir une suite de transformations jusqu'à obtenir la quantité de produit fini demandée. La quantité de matière nécessaire peut être introduite dans le système (elle provient des hauts fourneaux et, dans ce cas, le poids de matière introduite sera multiple d'un poids donné) ou, chaque fois que cela est possible, prélevée dans la matière en stocks (excédent de matière lors de fabrications antérieures). Chaque transformation à subir par la matière pourra éventuellement être effectuée sur différentes machines. Il se pose donc, pour l'exécution d'une transformation donnée, un problème de choix de machines. A chaque stade de fabrication, on pourra stocker un certain nombre d'éléments de matière correspondant à l'excédent de matière traitée par rapport au produit demandé.

Dès qu'une commande est passée, et avant qu'elle ait subi un commencement d'exécution, on dira qu'elle figure sur carnet. Entre l'instant de début d'exécution et l'instant de fin d'exécution, la commande sera dite ' en cours '.

La décision d'exécution d'une commande étant prise, on recherche si elle figure entièrement ou en partie dans les stocks disponibles. Si oui, la commande est fractionnée de façon à utiliser au maximum les stocks disponibles, et ceci en commençant par la matière stockée la plus élaborée.

Si bien que l'exécution d'une commande peut se schématiser comme suit :



(Les flèches verticales descendantes indiquent le stockage d'excédent de matière aux différents niveaux).

Remarquons qu'un tel problème est particulier : il ne s'agit pas de fabriquer un certain nombre d'éléments qui viendront s'assembler pour former un produit fini, mais de transformer une certaine quantité de matière première qui, au fur et à mesure qu'elle s'élabore, peut perdre une fraction de son ensemble qui sera stockée et servira à la fabrication éventuelle d'autres produits. En ce sens, notre problème pourrait être qualifié de semi divergent, contrairement aux problèmes courants qui pourraient être dits convergents.

Nous pouvons, dès à présent, donner une idée de la complexité du problème. Il est facile de comprendre qu'il est souhaitable de décider l'exécution d'une commande le plus tard possible, de façon à diminuer les ' en cours ' et de réduire au maximum la quantité de matière figurant dans le système. Cependant, une telle politique risque d'entraîner un dépassement systématique des délais.

De plus, elle réduit considérablement le choix des produits susceptibles d'être lancés, à un instant donné, sur une machine libre. Elle risque donc d'augmenter le temps d'immobilisation des machines qui peuvent exécuter des travaux très différents, mais qui doivent être immobilisées afin d'y être préparées. Toutes ces notions seront

précisées par la suite, mais nous voyons dès à présent que les méthodes classiques de la recherche opérationnelle peuvent difficilement être utilisées dans un tel problème.

Tout au plus peut-on simuler une fabrication et observer son évolution. Nous reviendrons plus loin sur ce point particulier mais avant de préciser et de formaliser le problème, il convient de donner un certain nombre de définitions.

II - DEFINITIONS

1) Etat de la matière

On dira que la matière change d'état lorsque le poids, la forme, l'aspect ou la température d'un de ses éléments varie, ou lorsqu'il est transporté d'un point à un autre de l'usine.

La fabrication d'un produit sera donc caractérisée par une suite d'états (e_0, e_1, \dots, e_n) de la matière qui lui est attribuée. Une telle suite sera appelée chemin de fabrication.

e_0 correspond à la matière introduite ou prélevée dans les stocks (état initial) et e_n désigne l'état final (produit fini).

2) Machine

On appellera " machine " tout facteur permettant de faire passer un élément quelconque de matière d'un état e_i à un état e_j . Ainsi, une machine pourra tout aussi bien être une équipe de finition qu'un train de laminage, un four ou un appareil de transport.

3) Travail d'une machine

On dira qu'une machine effectue un travail lorsqu'elle fait passer un ou plusieurs éléments de matière d'un état e_i à un état e_j .

4) Opérations

On appellera opérations les classes de travaux définies de la manière suivante :

deux travaux appartiennent à la même classe s'ils peuvent être effectués par les mêmes machines, si le passage d'un travail à l'autre ne nécessite aucune préparation de la machine et si les temps de préparation nécessaires pour passer de l'exécution d'un de ces travaux à un travail quelconque est le même pour chacune de ces machines.

La définition de ces opérations dépend donc des machines, des temps de préparation et des travaux que peuvent effectuer les machines considérées.

Plus précisément, si ζ_i et ζ_j sont deux travaux, on dira qu'ils appartiennent à la même opération θ_k si et seulement si :

- 1) Toute machine pouvant effectuer ζ_i peut effectuer ζ_j et réciproquement.
- 2) Sur toute machine pouvant exécuter ζ_i et ζ_j , il est possible d'exécuter ζ_i et ζ_j (ou ζ_j et ζ_i) à la suite l'un de l'autre sans qu'il soit nécessaire d'immobiliser la machine en vue de la modifier.
- 3) Quel que soit $l \neq i, j$, et quelle que soit la machine M_s pouvant effectuer les travaux ζ_i, ζ_j , et ζ_l :

$$\theta_{il}^s = \theta_{jl}^s$$

$$\theta_{li}^s = \theta_{lj}^s$$

où $\theta_{m n}^k$ représente le temps de préparation nécessaire à la machine M_k pour exécuter le travail ζ_n sachant qu'elle vient d'exécuter le travail ζ_m .

5) Systeme

On appellera système l'ensemble constitué

- * des machines
- * des commandes figurant sur carnet
- * de la matière en cours de transformation
- * des stocks, c'est à dire de la matière affectée en attente à différents niveaux d'élaboration.

III - EXPOSE DU PROBLEME

Le moment est venu de présenter, à l'aide des termes définis précédemment, le système dont nous nous occuperons et de dégager les questions qu'il nous pose.

Désignons ce système par (S).

Il se compose :

- 1 - D'un ensemble $\{M_i\}_{i=1, \dots, n}$ de machines, la machine M_i pouvant effectuer les opérations $\theta_{i_1}, \dots, \theta_{i_{p_i}}$
- 2 - D'un ensemble E_1 d'en cours, c'est à dire de quantités de matière en cours de fabrication et affectées à des commandes.
- 3 - D'un ensemble E_2 constituant le carnet de commandes.
- 4 - D'un ensemble de stocks désigné par ST.

Chacun de ces ensembles est caractérisé par un certain nombre de paramètres et son état est décrit à chaque instant, par des " fonctions ". Ces paramètres et " fonctions " sont détaillés au cours du chapitre suivant. Ici nous donnerons au terme " fonction " un sens plus large que le sens mathématique : une fonction sera un ensemble de nombres ou de quantités alphanumériques dépendant du temps.

A l'instant t_0 , nous connaissons l'état X_0 de (S). L'évolution de l'état de (S) au cours du temps dépendra d'une part des commandes qui vont se présenter et d'autre part des décisions que nous allons prendre.

A l'instant t_0 par exemple, ces décisions consisteront :

- 1 - A choisir, parmi les machines libres, celles que nous allons utiliser.
- 2 - A décider, pour chacune de ces machines, de l'opération qu'elle effectuera.
- 3 - A choisir la commande dont nous voulons poursuivre l'élaboration.

4 - Si la commande choisie est en début de fabrication, à déterminer le chemin de fabrication qu'elle suivra.

Ces décisions étant prises, le système (S) évoluera jusqu'à l'instant t_1 , instant où la première des machines en fonctionnement à l'instant t_0 , ou que l'on a décidé d'utiliser à cet instant, se trouvera libérée. L'état du système sera alors désigné par \mathcal{E}_1 .

Plus généralement, supposons connu l'état \mathcal{E}_i du système à l'instant t_i .

\mathcal{E}_{i+1} , état du système à l'instant t_{i+1} , est alors fonction de :

- * \mathcal{E}_i , état du système à l'instant t_i
- * D_i , ensemble des décisions prises à l'instant t_i
- * $C_{i, i+1}$, ensemble des nouvelles commandes qui se présentent entre t_i et t_{i+1} .

Il nous est donc possible d'écrire

$$\mathcal{E}_{i+1} = \mathcal{E}_{i+1} (\mathcal{E}_i, D_i, C_{i, i+1})$$

et, de proche en proche :

$$\mathcal{E}_{i+1} = \mathcal{E}_{i+1} (\mathcal{E}_0, D_0, D_1, \dots, D_i, C_{0,1}, C_{1,2}, \dots, C_{i, i+1})$$

\mathcal{E}_0 est connu. La suite $C_{0,1}, C_{0,2}, \dots, C_{i, i+1}, \dots$ des commandes introduites l'est également. Si bien que la suite $\mathcal{E}_0, \mathcal{E}_1, \dots$ des états du système est fonction de la suite (D_0, D_1, \dots) des décisions prises. Cette situation est résumée dans le schéma suivant :



Le problème consiste, à première vue, à déterminer la suite $(D_0, D_1, \dots, D_n \dots)$ des décisions qui conduit à la suite $(\mathcal{E}_0, \mathcal{E}_1, \dots, \mathcal{E}_n)$ des états " la plus favorable ". En fait, on s'aperçoit bien vite qu'il est impossible de déterminer si une suite d'état est la plus favorable de toutes les suites possibles. Il est généralement difficile de comparer deux suites d'états données. Les raisons essentielles en sont simples :

a) Un état \mathcal{E}_i est caractérisé par un ensemble de fonctions que nous choisissons (c'est l'objet du paragraphe suivant). Or ces fonctions ne nous donnent que certaines caractéristiques de l'état de (S) et nous sommes incapables d'en donner une liste exhaustive.

b) Les valeurs prises par ces fonctions au cours du temps ne sont pas indépendantes et, comme nous le verrons, une évolution favorable des valeurs prises dans le temps par l'une d'elles est souvent associée à une évolution défavorable des valeurs prises par une autre fonction. Par exemple, une diminution du temps moyen d'immobilisation des machines, souhaitable en période de haute conjoncture, peut entraîner des dépassements de délais importants pour certaines commandes.

c) Supposons que (S) soit passé par la suite $(\mathcal{E}_0, \mathcal{E}_1, \dots, \mathcal{E}_n)$ d'états. Nous sommes incapables de dire si cette évolution est favorable pour la suite ou, au contraire, si l'état \mathcal{E}_n va rendre plus difficiles les décisions à prendre au temps t_n . Si par exemple on s'est attaché essentiellement à respecter les délais, entre t_0 et t_n , il se peut que par la suite beaucoup soient dépassés. Nous serons donc plus modestes dans nos buts.

Après avoir déterminé des fonctions caractérisant les états, nous construirons d'abord une maquette, puis ce que nous avons appelé un " programme de test de règles de décisions ".

La maquette est un programme qui permet à l'ordinateur de calculer les valeurs prises par les fonctions définissant l'état du système à partir des décisions prises. Elle permet à l'utilisateur de prendre, à chaque instant t_j , les décisions qu'il juge opportunes en fonction de l'état du système. Il peut ainsi tester ses propres décisions et recommencer le test autant de fois qu'il le souhaitera

jusqu'à arriver à une suite $(\mathcal{E}_0, \mathcal{E}_1, \dots, \mathcal{E}_n, \dots)$ d'états qui lui semble favorable.

Le ' programme de test de règles de décisions ' nous a permis, des règles simples étant choisies à priori, d'observer le résultat de leur application sur la suite des états.

Il ne faudra pas perdre de vue, que, dans un cas comme dans l'autre, la suite des états observés dépend de l'état initial et des commandes introduites.

IV - CARACTERISATION DES ETATS

J'ai choisi, pour caractériser l'état \mathcal{E}_i du système (S) à l'instant t_i , un certain nombre de fonctions qui peuvent être divisées en deux catégories :

1) Les fonctions qui définissent ce que l'on pourrait appeler l'état du système à l'instant t_i , à savoir :

- a) les délais actualisés de toutes les commandes. Les valeurs de ces fonctions sont données si l'utilisateur le désire.
- b) le poids et le nombre des en - cours.
- c) pour chaque commande arrivée en fin d'exécution entre t_{i-1} et t_i , l'écart et l'écart pondéré par le poids entre le temps de fin de fabrication et le délai moyen.
- d) le nombre d'ouvriers employés entre t_{i-1} et t_i .
Le nombre d'ouvriers employés pour préparer une machine donnée M_j est généralement différent du nombre d'ouvriers nécessaires pour faire fonctionner M_j . La période $[t_{i-1}, t_i]$ se trouve donc souvent divisée en un certain nombre d'intervalles définis par les temps de fin de préparation de certaines machines.
- e) les poids des stocks.

f) dans le cas de la maquette de calcul :

- α) Le temps de fabrication minimum indispensable pour terminer l'exécution des commandes susceptibles d'être prises en charge par une machine libre à l'instant t_i .
- β) L'opération que vient d'effectuer chaque machine libre à l'instant t_i .

2) Les fonctions que définissent la suite ($\xi_0, \xi_1, \dots, \xi_i$) des états du système. Ces fonctions caractérisent en partie la qualité de l'état ξ_i .

- a) le temps moyen d'immobilisation des machines entre t_0 et t_i . Ce temps moyen T_i est défini de la manière suivante :

$$T_i = \frac{\text{Somme des temps d'immobilisation des machines entre } t_0 \text{ et } t_i}{t_i - t_0}$$

- b) l'écart moyen pondéré pour toutes les commandes terminées entre t_0 et t_i

Si C_1, C_2, \dots, C_p sont les commandes terminées entre t_0 et t_i ,
 P_1, P_2, \dots, P_p leurs poids respectifs,
 $\mu_1, \mu_2, \dots, \mu_p$ leurs délais respectifs (délais demandés par le client)
et S_1, S_2, \dots, S_p leur temps de fin de fabrication respectifs,

l'écart moyen G_i pondéré par leur poids est défini comme suit :

$$G_i = \frac{\sum_{j=1}^p (S_j - \mu_j) P_j}{\sum_{j=1}^p P_j}$$

c) la dispersion pour toutes les commandes terminées entre t_0 et t_i . Avec les notations précédentes, cette dispersion F_i est définie comme suit :

$$F_i = \frac{\sum_{j=1}^p (S_j - \mu_j)^2 P_j}{\sum_{j=1}^p P_j}$$

CHAPITRE II

F I C H I E R S

Le système et les différents états qu'il prend dans le temps sont définis par des paramètres et des fonctions. Ces paramètres et les valeurs de ces fonctions sont rangés dans des fichiers que nous présentons maintenant.

I - FICHIERS CONCERNANT LES COMMANDES

1) Fichier ' P I A '

Ce fichier contient les paramètres caractérisant les commandes (nuance d'acier, poids)

2) Fichier ' P I B '

Il contient les valeurs des fonctions caractérisant les commandes à l'instant considéré : code de la commande (variable), nombre de ' chemins de fabrication ' possibles, codes indiquant si la commande est ' en cours ' ou sur carnet, en attente ou en cours de passage sur une machine, un entier indiquant le nombre de commandes issues de la commande initiale, les délais actualisés, le poids de matière attribuée à cette commande.

3) Fichier ' P II A '

Contient, pour chaque ' chemin de fabrication ' : le nombre d'étapes, la forme et le poids actuels de l'élément de matière attribués à la commande.

4) Fichier ' P II B '

Contient les caractéristiques des étapes.

Pour chaque étape de chaque chemin de fabrication, on y trouve :

l'opération à effectuer à cette étape, la mise au mille, la forme de l'élément de matière à la fin de l'étape, le temps de passage de 1000 unités de poids de matière par cette étape, le nombre de fractionnements de l'élément de matière au cours de cette étape.

II - FICHIER CONCERNANT LES STOCKS : Fichier ST

Il contient le nombre d'unités de matière stockées, rangées par nuance, forme et poids unitaire.

III - FICHIER DE CORRESPONDANCE OPERATIONS - MACHINES : Fichier OM

Ce fichier contient, pour chaque machine, les codes des opérations qu'elle peut effectuer.

IV - FICHIERS ' MACHINES '

1) Fichier ' MA I '

Contient, pour chaque machine, et pour tous les couples (θ_i, θ_j) d'opérations que cette machine peut effectuer, le temps de préparation et le nombre d'ouvriers nécessaires pour préparer la machine à effectuer θ_j sachant qu'elle vient d'effectuer θ_i (et réciproquement).

2) Fichier ' MA II '

Il contient, pour chaque machine et pour chaque opération qu'elle peut effectuer, le nombre d'ouvriers nécessaires à son fonctionnement et le temps de passage de 1000 unités de poids de matière pour cette opération.

3) Fichier ' MA III '

Il contient les valeurs des fonctions caractérisant l'état de chaque machine à l'instant considéré : nombre d'ouvriers employés, code indiquant si la machine est en préparation ou en fonctionnement, temps de préparation et temps de fonctionnement restant, numéro de l'opération en cours (ou précédemment effectuées).

CHAPITRE III

MAQUETTE

I - UTILISATION DE LA MAQUETTE

A l'instant t_i , nous avons demandé à la maquette d'indiquer :

1) Pour chaque machine occupée, l'opération qu'elle est en train d'effectuer

2) Pour chaque machine libre :

α) l'opération précédemment effectuée par la machine.

β) les commandes dont elle est susceptible de faire avancer l'élaboration avec, pour chacune de ces commandes :

$\beta 1$) Un message indiquant si la commande est déjà lancée ou non (produit en fabrication ou sur carnet)

$\beta 2$) Le code de la commande.

$\beta 3$) L'opération que la matière attachée à cette commande doit subir à cet instant.

$\beta 4$) Le chemin de fabrication intéressé (égal à 1 si la commande est déjà lancée)

$\beta 5$) Le temps de fabrication minimum attaché à cette commande, à partir du niveau d'élaboration auquel elle se trouve.

$\beta 6$) Les délais actualisés.

$\beta 7$) Si la commande a déjà été lancée, le poids actuel et le poids final, sinon, les poids actuels minimum et maximum ainsi que les poids finals associés.

La maquette donne alors la possibilité à l'utilisateur de choisir les machines qu'il va lancer parmi les machines libres (si l'utilisateur choisit par erreur une machine occupée, un message est prévu et le choix est repris). Pour chaque machine choisie, l'utilisateur indique l'opération qu'il veut lui voir effectuer ainsi que la (ou les) commande (s) dont il veut poursuivre l'élaboration. Lorsqu'une de ces commandes est à lancer, il sera demandé à l'utilisateur de préciser le poids de matière associée. Enfin, l'utilisateur précisera le temps au bout duquel il veut connaître la situation du système. Si ce temps ζ est inférieur au temps ζ' au bout duquel la première machine lancée à l'instant t_i ou en fonctionnement à l'instant t_i sera libérée, alors la maquette donnera la situation du système au temps $t_{i+1} = t_i + \zeta$. Sinon, $t_{i+1} = t_i + \zeta'$. La situation à l'instant t_{i+1} étant calculée, la maquette permettra d'introduire de nouvelles commandes à cet instant et le cycle recommencera.

II - MODE DE DESCRIPTION DE LA MAQUETTE

Dès qu'un programme atteint un certain degré de complexité, il est vain d'essayer de le comprendre à l'aide de l'organigramme qui lui est associé. J'ai essayé dans ce qui suit d'adopter une présentation qui, partant de grandes lignes du programmes, permet de pénétrer progressivement dans les détails. Cette présentation par niveaux (de complexité) peut conduire jusqu'aux niveaux les plus élémentaires du programme, ceux des instructions.

Cette manière de présenter un programme obéit à un certain nombre de règles que je vais énoncer :

- 1) Le niveau I est constitué par l'énoncé des traitements à exécuter au cours du programme. Ces traitements sont tous exécutés et dans l'ordre dans lequel ils sont énoncés.

Exemple

Si les énoncés des n traitements à exécuter au cours du programme sont désignés par a_1, a_2, \dots, a_n , le niveau I se présentera de la manière suivante :

NIVEAU I Page 1

- 1) a_1
- 2) a_2
- ⋮
- n) a_n

Si une page ne suffit pas aux énoncés, on en utilisera autant qu'il en faudra et leur ordre de lecture sera donné par les références : Page 1, Page 2,, Page n.

2) Le niveau II reprend éventuellement les énoncés a_1, a_2, \dots, a_n pour les préciser, et dans l'ordre où ils apparaissent au niveau I.

Les précisions concernant a_i doivent être données sous forme d'énoncés de traitements $a_{i1}, a_{i2}, \dots, a_{ik}$ qui doivent être tous exécutés et dans l'ordre dans lequel ils apparaissent. Ces énoncés sont numérotés. a_i est rappelé en début de page, avec le numéro qu'il porte au niveau I, et souligné.

Exemple

Le niveau I, donné en exemple ci-dessus, pourrait se prolonger de la manière suivante :

NIVEAU I Page 1

- 1) a_1
- 1) a_{11}
- 2) a_{12}
- 3) a_{13}

NIVEAU II Page 2

- 2) a_2
- 1) a_{21}

2) a_{22}

NIVEAU II Page 3

3) a_3

1) a_{31}

2) a_{32}

3) a_{33}

4) a_{34}

etc ...

Notons que les énoncés des traitements précisant a_i peut se prolonger sur plusieurs pages et que tout énoncé a_i n'a pas nécessairement besoin d'être précisé. Par ailleurs, les pages du niveau II sont numérotées dans leur ordre de lecture.

- 3) Le niveau III précise de la même manière les énoncés a_{ij} (ou certains de ces énoncés seulement)
- 4) On peut prolonger ces niveaux jusqu'à l'écriture du programme
- 5) Un énoncé peut représenter un traitement à exécuter plusieurs fois (notion de boucle). Alors la précision de ce traitement au niveau suivant se présentera sous la forme :

1) Pour [précision des différentes applications
du traitement]

- | | | |
|--------------|---|--|
| 2) a_{ij1} | } | Enoncés des différents modules
rencontrés au cours de ce traitement |
| 3) a_{ij2} | | |
| 4) a_{ij3} | | |
| 5) a_{ij4} | | |

Exemple

NIVEAU 4 Page p

1 - 3 - 4) Recherche de l'état des différentes machines au
temps t_i

1) Pour chaque machine

- | |
|---|
| 2) Recherche du temps d'occupation à l'instant t_i |
| 3) Recherche du temps total d'immobilisation à
l'instant t_i |
| 4) |

6) Si un traitement nécessite un ou plusieurs tests, sa pré-
cision au niveau suivant comportera une table de décision.

Exemple

NIVEAU III Page q

2 - 5) a_{25}

1) a_{251}

2) a_{252}

3)

C1 (condition 1)	O	N	O	N
C2 (condition 2)	0	0	N	N
4) a_{253}	1			3
5) a_{254}	2	1	1	
6) a_{255}			2	
7) a_{256}			3	1

8) a_{257}

.....

Si C1 et C2 sont vrais, la séquence exécutée sera, dans l'ordre :

a_{251} , a_{252} , a_{253} , a_{254} , a_{257}

Si C1 est vrai et si C2 n'est pas réalisée, la séquence exécutée sera :

a_{251} , a_{252} , a_{254} , a_{255} , a_{256} , a_{257}

Si C1 et C2 ne sont pas réalisés, la séquence sera, dans l'ordre :

a_{251} , a_{252} , a_{256} , a_{253} , a_{257}

Indiquons cependant que, lorsque le test est simple, on préférera à une table de décision

1	C	O	N
2)	a_1	1	
3)	a_2		1

Une phrase telle que :

Si (C) alors 1) (a_1)

sinon 2) (a_2)

Je vais donner un exemple simple d'utilisation de cette manière de présenter un programme. Considérons le programme de résolution d'une équation du second degré. Nous pourrions le présenter comme suit (à un même niveau, plusieurs pages seront représentées sur la même feuille).

NIVEAU I Page 1

- 1) Lecture des coefficients a , b , c
- 2) Calcul de Δ
- 3) Calcul des racines éventuelles et sortie des résultats

NIVEAU II Page 1

1) Lecture des coefficients a , b , c

- 1) Lecture de a
- 2) Lecture de b
- 3) Lecture de c

NIVEAU II Page 2

2) Calcul de Δ

1) $\Delta = b^2 - 4ac$

NIVEAU II Page 3

3) Calcul des racines éventuelles et calcul des résultats

1) C1 : Δ est positif	0	N	N
C2 : Δ est négatif	N	N	0
2) Calcul des deux racines	1		
3) Calcul de la racine double		1	
4) Sortie du message ' Pas de racine réelle '			1

Remarquons que cette table de décision aurait pu être remplacée par la séquence :

1) Si (Δ est positif) alors 2) (calcul des deux racines)
sinon si (Δ est négatif) alors 4) (sortie de message ' pas de racine réelle)
sinon 3) (calcul de la racine double)

3 - 2) Calcul des deux racines

1) $x_1 = (-b + \sqrt{\Delta}) / 2a$

2) $x_2 = (-b - \sqrt{\Delta}) / 2a$

4) Pour chaque racine

5) Impression de sa valeur

3 - 3) Calcul de la racine double

1) $x = -b / 2a$

2) Impression de x.

Cet exemple simple a permis d'illustrer notre mode de présentation. Nous allons l'appliquer à la maquette, en détaillant tout d'abord les fichiers utilisés. Précisons de suite que nous n'avons pas poussé l'exposé du programme jusqu'aux niveaux les plus élémentaires, mais suffisamment loin cependant pour permettre de le reconstituer rapidement.

F I C H I E R S

1) Fichiers concernant les commandes

1 - 1) Fichier ' Produit I A '

Ce fichier contient les paramètres attachés aux commandes.
Pour chaque commande, on y trouvera :

- ① Un entier désignant la nuance d'acier à utiliser
- ② Un réel désignant le poids de la commande en kilogrammes
- ③ Un réel désignant le poids maximum que l'usine peut se permettre de livrer.

1 - 2) Fichier ' Produit I B '

Ce fichier contient les valeurs des fonctions caractérisant la commande à l'instant considéré. Il contient, pour chaque produit :

- ① Un entier de quatre chiffres, qui est le code de la commande en question. En entrée, le premier de ces quatre chiffres sera un 0. Nous savons qu'une commande peut, au moment du lancement, être fractionnée et se voir attribuer des éléments de matière diversement élaborés.

Le premier des quatre chiffres servira à distinguer ces différentes fractions de la commande initiale.

- ② Un entier donnant le nombre de ' chemins de fabrication ' (c'est à dire de suites d'états de la matière) possibles pour l'exécution de la commande. Dès que la décision de fabrication est prise, une

de ces suites d'états est choisie et cet entier devient égal à 1.

- ③ Un entier égal
 - * à un si la commande figure encore sur carnet
 - * à 2 si elle est ' en cours '

- ④ Un entier
 - * égal a 0 si la matière attribuée à cette commande est en attente ou si la commande figure encore sur carnet

 - * égal au numéro de la machine sur laquelle la matière qui lui est attribuée est en cours de passage sinon

- ⑤ Un entier désignant le nombre de commandes issues de la commande initiale (donc cet entier est égal à 1 pour les commandes figurant encore sur carnet)

- ⑥ Un réel désignant le délai (actualisé) souhaité par le client

- ⑦ Un réel désignant le délai à ne pas dépasser (délai actualisé). Ce délai est donc supérieur (ou égal) au précédent.

- ⑧ Un réel, nul si la commande figure encore sur carnet, égal au poids de matière qui lui est attribuée sinon.

1 - 3) Fichier ' Produit II A '

Ce fichier contient les valeurs des fonctions caractérisant chaque ' chemin de fabrication ' attaché à chaque commande.

On y trouvera :

- ① Un entier donnant le nombre d'étapes restant à effectuer dans le chemin de fabrication considéré.
- ② Un entier désignant la forme de l'élément de matière attribuée à la commande (forme fonction du temps)
- ③ Un réel donnant, à l'instant considéré, le poids de l'élément de matière attribuée à la commande.

Si la commande figure encore sur carnet, ② et ③ désigneront respectivement, pour chaque ' chemin de fabrication ' possible, la forme et le poids de l'élément de matière qui sera attribuée à la commande lors de la décision de lancement.

1 - 4) Fichier ' Produit II B '

Ce fichier contiendra, pour chaque étape de chaque chemin de fabrication :

- ① Un entier désignant l'opération à effectuer à cette étape
- ② Un réel désignant le poids de matière nécessaire avant passage par cette opération pour obtenir 1000 unités de poids de matière après passage par cette opération
- ③ Un entier donnant la forme de l'élément de matière après passage par cette opération
- ④ Un réel représentant le temps de passage de mille unités de poids de matière par cette opération
- ⑤ Un entier indiquant en combien d'éléments est fractionné chaque élément de matière au cours de cette opération.

2) Fichier concernant les stocks

Ce fichier concerne la matière qui n'est affectée à aucune commande.

Pour chaque quantité de matière dont l'élément est de forme j et constitué d'acier de nuance i , on y trouvera :

- * le nombre de stocks de poids élémentaires différents
- * les poids élémentaires, rangés par ordre croissant
- * pour chacun de ces poids, le nombre d'éléments figurant en stocks.

3) Fichier de liaison entre opérations et machines

Ce fichier contient une matrice à termes entiers, $IA (m, n)$ (m : nombre d'opérations différentes que l'on rencontre ; n : nombre de machines du système) avec :

$$IA (i, j) = \begin{cases} 1 & \text{si l'opération } i \text{ peut être effectuée par la machine } j \\ 0 & \text{sinon} \end{cases}$$

4) Fichiers caractérisant les machines

4 - 1) Fichier ' Machine I '

Pour chaque machine, ce fichier contient une suite de quadruplets (i, j, n, t) indiquant que, pour préparer la machine à effectuer l'opération j sachant qu'elle vient d'effectuer l'opération i , il faut n ouvriers durant un temps t .

4 - 2) Fichier ' Machine II '

Pour chaque machine, ce fichier contient une suite de doublets (i, n_1) indiquant qu'il est nécessaire d'affecter à cette machine n_1 ouvriers lorsqu'elle effectue l'opération i .

4 - 3) Fichier ' Machine III '

Ce fichier contient les fonctions définissant, à chaque instant, l'état de chaque machine. On y trouve, pour chaque machine :

- ① Un entier désignant le nombre d'ouvriers employés si la machine est en fonctionnement (0 sinon)
- ② Un réel représentant le coût d'immobilisation de la machine par unité de temps
- ③ Un réel représentant le temps durant lequel la machine restera occupée (à partir de l'instant considéré).
Si la machine est libre, ce réel sera nul
- ④ Un entier désignant l'opération en cours si la machine est occupée, et l'opération précédemment effectuée sinon.
- ⑤ Un réel représentant le temps durant lequel la machine restera en préparation (= 0 si la machine n'est pas en préparation)
- ⑥ Un entier désignant le nombre d'ouvriers employés si la machine est en préparation (0 sinon).

- 1) Renseignements nécessaires aux prises de décisions à l'instant t (Programme RENS)
- 2) Décisions de l'opérateur à l'instant t (programme DEC)
- 3) Recherche de la période Δt (temps au bout duquel la première machine en fonctionnement après les décisions va s'arrêter)
- 4) Calcul de la situation de l'ensemble à l'instant $t + \Delta t$ (programme CAIS)
- 5) Enregistrement à l'instant $t + \Delta t$ des commandes passées entre t et $t + \Delta t$
- 6) Décision quant à la poursuite, à l'interruption ou à la reprise de l'expérience.

1) Renseignements nécessaires aux prises de décisions à l'instant t.

1) Recherche des machines libres à l'instant t

2) Pour chaque machine libre à l'instant t

3) Sortie du message ' La machine X est libre '

4) Recherche des opérations que cette machine peut effectuer

5) Pour chaque opération que cette machine peut effectuer

6) Sortie du message ' Cette machine peut effectuer l'opération X '

7) Recherche des commandes dont la matière aura à subir cette opération en premier lieu

8) Pour chacune de ces commandes

9) Sortie du message ' Cette machine peut recevoir la commande Y suivant le chemin X '

10) Sortie des délais actualisés

11) Recherche des valeurs des fonctions caractérisant cette commande et sa fabrication.

12) Sortie de ces valeurs

2) Décisions de l'opérateur à l'instant t

1) Choix des machines que l'on veut lancer

2) Pour chaque machine choisie

3) Lecture dans ' MA III ' des valeurs des fonctions caractérisant cette machine à l'instant t

4) Si la machine est occupée

5) Sortie d'un message indiquant le refus

6) Sinon

7) Machine acceptée

8) Choix de l'opération et de la commande

3) Recherche de la période Δt

1) Pour chaque machine

2) Lecture dans ' MA III ' des paramètres caracté-
risant la machine

3) Si la machine est en cours de fonctionnement

4) Conservation du temps de fonction-
nement restant

5) Recherche du temps de fonctionnement minimum conservé

6) Attribution de ce temps minimum à Δt .

4) Calcul de la situation de l'ensemble à l'instant $t + \Delta t$ 1) Pour chaque machine

2) Lecture dans ' MA III ' des fonctions caractérisant l'état de la machine

3) Si la machine est en cours de fonctionnement à l'instant t
et si son temps de préparation est inférieur à Δt
conservation du temps de préparation

4) Classement des temps de préparation conservés par ordre croissant

5) Pour chaque période ainsi déterminée entre t et $t + \Delta t$ 6) Pour chaque machine occupée

7) Si la machine est en préparation durant cette période

8) Lecture des paramètres correspondant à cette machine dans ' MA I '

9) Conservation du nombre d'ouvriers nécessaires à la préparation

10) Conservation du coût de préparation

11) Sinon

12) Lecture des paramètres correspondant à cette machine dans ' MA II '

13) Conservation du nombre d'ouvriers nécessaires au fonctionnement

14) Calcul du coût cumulé durant cette période

15) Calcul du nombre d'ouvriers employés durant cette période

16) Pour chaque machine

- 17) Recherche de l'état de la machine à l'instant $t + \Delta t$
- 18) Modification du fichier ' MA III '

19) Pour chaque commande

- 20) Lecture des fonctions caractérisant cette commande dans ' P I B ' et ' P II B '
- 21) Modification des délais actualisés dans ' P I B '
- 22) Si la matière attachée à cette commande était en cours de passage sur une machine qui vient d'être libérée
 - 23) Réintégration de la matière attachée à cette commande dans l'ensemble de la matière libre

5) Enregistrement à l'instant $t + \Delta t$ des commandes passées entre t et $t + \Delta t$

1) Pour chaque commande passée entre t et $t + \Delta t$

2) Si il existe des stocks de même nuance, forme et poids unitaire que le produit fini cherché

et si le poids de ces stocks est supérieur ou égal au poids minimum de la commande

3) Sortie du message ' Commande X exécutée '

4) Modification du stock concerné dans ' ST '

5) Sinon

6) Enregistrement de la commande dans ' P I A ' , ' P I B ' , ' P II A ' et ' P II B '

6) Décision quant à la poursuite, à l'interruption ou à la reprise de l'expérience

1)

C1 : On veut poursuivre l'expérience	O	N	N	
C2 : On veut recommencer l'expérience	N	O	N	
2) Retour en 1	1	2		
3) Réinitialisation des fichiers		1		
4) Sortie du programme			1	

1 - 1) Recherche des machines libres à l'instant t

1) Pour chaque machine

2) Lecture dans ' MA III ' des valeurs des fonctions caractérisant l'état de la machine

3) Si la machine est libre à l'instant t

4) Conservation de son code

1 - 4) Recherche des opérations que cette machine peut effectuer

1) Lecture dans le fichier ' OM ' des codes des opérations
que cette machine peut effectuer

2) Conservation de ces codes

1 - 7) Recherche des commandes dont la matière aura à subir cette opération en premier lieu

1) Pour chaque commande

2) Lecture dans ' P I B ' des fonctions caractérisant la commande

3) Si la commande est sur un carnet

ou si la matière attachée à cette commande n'est pas en cours de transformation

4) Test des chemins de fabrication

1 - 10) Sortie des délais actualisés

- 1) Lecture dans ' P I B ' des valeurs des fonctions concernant cette commande
- 2) Impression des délais actualisés

1 - 11) Recherche des valeurs des fonctions caractérisant cette commande et sa fabrication

1) Lecture dans ' P I A ' , ' P II A ' et ' P II B ' des paramètres et fonctions caractérisant cette commande et sa fabrication à l'instant t

2) Si cette commande est en cours

3) Sortie du message ' commande en cours '

4) Donnée du poids actuel

5) Sinon

6) Sortie du message ' commande sur carnet '

7) Recherche des fractionnements possibles de cette commande

8) Calcul du temps de fabrication minimum de cette commande

2 - 8) Choix de l'opération et de la commande

- 1) Choix d'une opération
- 2) Lecture dans le fichier ' O M ' des opérations qu'est susceptible d'effectuer la machine choisie
- 3) Si l'opération choisie peut être effectuée par la machine choisie
 - 4) L'opération est acceptée
 - 5) Choix de la commande
- 6) Si non, refus de l'opération et de la machine choisies.

4 - 23) Réintégration de la matière attachée à cette commande dans l'ensemble de la matière libre

- 1) Modification des fonctions dans ' P I B '
- 2) Suppression dans ' P II B ' de l'étape qui vient de se terminer
- 3) Si cette étape était la dernière du chemin de fabrication
 - 4) Sortie du message ' Commande X terminée '
 - 5) Sortie des délais actualisés (négatifs si délais dépassés)
 - 6) Suppression dans ' P I A ' , ' P I B ' , ' P II A ' et ' P II B ' des fonctions et paramètres attachés à cette commande
 - 7) Recherche des commandes faisant partie de la même commande initiale et modification des fonctions ' fractionnement ' (Fichier ' P I B ')
- 8) Sinon, modification des fonctions de ' P II A '
- 9) Stockage dans ' S T ' de l'excédant à ce niveau.

1 - 7 - 4) Test des chemins de fabrication

1) Pour chaque chemin de fabrication attaché à commande

2) Lecture dans ' P II B ' des caractéristiques de la première étape

3) Si le code de l'opération concernant cette étape a été conservé

4) Conservation du code de la commande

5) Conservation du code du chemin de fabrication

1 - 11 - 6) Recherche des fractionnements possibles de cette commande

1) Pour chaque étape de fabrication, en commençant par la dernière

2) Calcul du poids maximum de matière correspondant à la commande à cette étape

3) Calcul du poids de chaque élément de matière à cette étape

4) Recherche, dans le fichier ' S T ', des stocks de même nuance, de même forme et de même poids élémentaire

5) Si un tel stock existe

6) Si son poids est supérieur ou égal au poids maximum de matière correspondant à la commande à cette étape

7) Suppression des étapes précédentes

8) Sinon

9) Fractionnement de la commande

10) Sortie du poids restant à fabriquer après fractionnement.

1 - 11 - 7) Calcul du temps de fabrication minimum de cette commande

1) Pour chaque étape restant à effectuer en commençant par la première

2) Calcul du poids de la matière correspondante à la commande après passage par cette étape

3) Calcul du temps de passage par cette étape

4) Conservation de ce temps de passage

5) Somme des temps conservés.

2 - 8 - 5) Choix de la commande

- 1) Choix de la commande dont on veut continuer ou commencer la fabrication
- 2) Lecture dans ' P I B ' des valeurs des fonctions caractérisant l'état de la commande choisie à l'instant T

3)

C1 : La commande choisie figure sur carnet	N	N	0
C2 : La matière attachée à cette commande est en attente	N	0	
4) Refus des choix précédents	1		
5) Commande retenue		1	1
6) Choix du chemin de fabrication			2

4 - 23 - 9) Stockage dans ' S T ' de l'excédant à ce niveau

1) Recherche de l'excédant de poids de matière attribuée à la commande à ce niveau

2) Si cet excédant de poids est supérieur au poids d'un élément de matière à ce niveau

3) Recherche du nombre maximum d'éléments de matière dont le poids total soit inférieur à l'excédant

4) Stockage de ces éléments, modification de ' S T '

5) Modification du poids actuel dans ' P I B '

2 - 3 - 5 - 6) Choix du chemin de fabrication

1) Lecture dans ' P II B ' des paramètres caractérisant la première étape du chemin de fabrication choisi

2) Si la première opération que doit subir la matière attachée à la commande est l'opération choisie

4) Le chemin est retenu

5) Enregistrement définitif des choix précédents

6) Sinon, refus des choix précédents.

2 - 8 - 5 - 6 - 5) Enregistrement définitif des choix précédents

- 1) Si la commande choisie n'est pas en cours
 - 2) Fractionnement éventuel de la commande en fonction des stocks
 - 3) Attribution éventuelle d'une certaine quantité de matière à la fraction non pourvue
 - 4) Suppression dans ' P II A ' et ' P II B ' des chemins inutiles
- 5) Modification dans ' P I B ' des valeurs des fonctions qui caractérisent la situation de la commande choisie
- 6) Modification dans ' M A III ' des valeurs des fonctions caractérisant l'état de la machine choisie.

2 - 8 - 5 - 6 - 5 - 2) Fractionnement éventuel de la commande en fonction des stocks

- 1) Lecture dans ' P I A ' , ' P I B ' , ' P I I A ' et P I I B ' des valeurs des paramètres et fonctions attachés à cette commande à l'instant t
- 2) Pour chaque étape de fabrication, en commençant par l'étape finale
 - 3) Calcul du poids maximum de matière correspondant à la commande à cette étape
 - 4) Calcul du poids de l'élément de matière à cette étape
 - 5) Recherche dans le fichier ' S T ' des stocks de même nuance, de même forme et de même poids élémentaire

6) C1 : Un tel stock existe	O	O	O	N
C2 : Son poids est supérieur au poids maximum de matière correspondant à la commande à cette étape	N	N	O	
C3 : Son poids est égal au poids maximum de matière correspondant à cette étape	N	O	N	
7) Fractionnement de la commande	1			
8) Attribution du stock à la commande générée	2	1	1	
9) Enregistrement de cette commande dans ' P I A ' , ' P I B ' , ' P I I A ' , ' P I I B '	3			
10) Suppression du stock utilisé dans ' S T '	4	2		
11) Suppression des étapes précédentes dans ' P I I B '		3	2	
12) Modification de ' P I I A '		4	3	
13) Modification du stock dans ' S T '			5	

III - CONCLUSIONS TIREES DE L'UTILISATION DE LA MAQUETTE

En principe, cette maquette permet à l'utilisateur de suivre l'évolution du système en fonction des décisions prises, et de recommencer l'expérience jusqu'à ce qu'il obtienne une évolution qu'il juge satisfaisante. Il peut alors appliquer les décisions trouvées à la réalité sans risques et en connaissant leurs conséquences (tout au moins sans les limites de temps couvertes par l'expérience).

L'utilisation d'une telle maquette présente un certain nombre d'inconvénients :

1) Elle est lente, et cette lenteur est due surtout au fait que l'utilisateur a besoin de réfléchir à chaque pas sur les décisions à prendre et doit transmettre (par l'intermédiaire du clavier), ses décisions à l'ordinateur.

2) Elle nécessite un compromis entre le nombre de renseignements disponibles et l'impossibilité dans laquelle se trouve tout utilisateur de tenir compte rapidement, lors de la prise des décisions, d'un nombre de renseignements excédant une dizaine.

Par contre, il faut signaler qu'une telle maquette a l'avantage de rendre palpable l'évolution du système en fonction des décisions prises. Après plusieurs expériences, j'ai pu me rendre compte de l'influence de certains facteurs sur l'évolution des états.

J'ai pu ainsi construire quelques règles de décisions que j'ai testées et que je vais présenter au cours du chapitre suivant.

CHAPITRE IV

TESTS DE QUELQUES REGLES DE DECISIONS

La maquette de calculs avait simplement pour but de permettre à l'utilisateur de se rendre compte des effets d'une suite de décisions prises par lui, et surtout de lui donner la possibilité de recommencer l'expérience avec de nouvelles suites de décisions, jusqu'à obtenir une suite d'états du système qui lui paraisse satisfaisante.

Nous allons maintenant nous proposer de tester quelques politiques de décisions. Ces politiques de décisions m'ont été suggérées lors de l'utilisation de la maquette de calculs. Elles sont contenues dans les règles que nous allons exposer.

La première de ces règles comporte deux paramètres qu'il nous suffira de faire varier pour modifier la politique des décisions.

La seconde et la troisième représentent chacune une politique possible, et nous comparerons les résultats obtenus aux résultats précédents. Il ne faudra pas perdre de vue que les résultats dépendent de la structure du système, et les conclusions tirées ne seront applicables qu'à la structure étudiée.

I - PREMIERE REGLE

Avant d'exposer cette règle, nous avons besoin de donner un certain nombre de définitions.

1) Définitions

a) Coefficient ' rapidité d'exécution d'une opération θ_i '

Ce coefficient, désigné par r_{θ_i} , sera défini comme suit :

$$r_{\theta_i} = \frac{p_{\theta_i}}{\sum_{j=1}^q p_{\theta_j}} \times R$$

où :

- * R est une constante positive
- * p_{θ_i} représente le poids moyen susceptible de subir l'opération θ_i par unité de temps sur l'ensemble des machines
Ce poids moyen est un résultat statistique
- * q est le nombre d'opérations que l'ensemble des machines est en mesure d'effectuer

r_{θ_i} est donc une caractéristique de l'opération θ_i indépendante du temps.

b) Coefficient 'urgence' d'une commande

Pour chaque ensemble d'éléments de matière attribués à une commande P_s et libre à l'instant t_i considéré, on calculera d'abord

$$f_{P_s}^{t_i} = \frac{\text{délais mini} + \text{délai maxi}}{2} - \text{temps de fabrication mini}$$

$$\text{nombre d'opérations restant à effectuer}$$

Soit alors :

$$A = \text{Max}_{S \in L} f_{P_s}^{t_i}$$

$B = \min_{S \in L} \int_{P_S}^{t_i}$, où L est l'ensemble des indices s tels que l'ensemble des éléments de matière attachés à la commande P_S soit libre à l'instant t_i .

Le coefficient 'urgence de la commande P_S à l'instant t_i ' s'écrira alors :

$$C_{P_S}^{t_i} = R \left(1 - \frac{\int_{P_S}^{t_i} - B}{A - B} \right) \quad \text{si } A \neq B$$

Si $A = B$, on prendra $C_{P_S}^{t_i} = R$

Les coefficients $C_{P_S}^{t_i}$ seront ainsi répartis dans l'intervalle $[0, R]$ et la commande P_S sera d'autant plus urgente que $C_{P_S}^{t_i}$ sera grand.

Ces coefficients se calculent à chacun des instants $t_0, t_1, t_2, \dots, t_i, \dots$. Ils sont fonctions de la situation de l'ensemble des commandes.

C) Coefficient 'charge de l'opération θ_k à l'instant t_i '

Soit $\mu_{\theta_k}^{t_i}$ le poids de l'ensemble des éléments de matière libres et attachés aux 'en cours' et de l'ensemble des éléments de matière que l'on doit prévoir pour les commandes sur carnet et qui devraient subir l'opération θ_k à l'instant t_i .

Alors le coefficient 'charge de l'opération θ_k à l'instant t_i ', que l'on notera $K_{\theta_k}^{t_i}$, sera défini comme suit :

$$K_{\theta_k}^{t_i} = \frac{\mu_{\theta_k}^{t_i} / r_{\theta_k}}{\sum_{\theta_l \in S_{t_i}} \mu_{\theta_l}^{t_i} / r_{\theta_l}} \times R$$

où S_{t_i} est l'ensemble des opérations pouvant être effectuées à l'instant t_i et R le même nombre positif qu'en (b). Ce coefficient est d'autant plus élevé que le poids de matière susceptible de subir l'opération θ_k à l'instant t_i est élevé et que l'exécution de l'opération θ_k sur l'ensemble des machines est lente.

2 - EXPOSE DE LA REGLE D'ORDONNANCEMENT

J'ai choisi la règle suivante :

- ① Lorsqu'une commande figurant sur carnet passe au stade ' d'en cours ', on conserve le chemin de fabrication qui assure à la matière attachée à cette commande le temps de passage minimum sur machines.
- ② Pour chaque commande P_s figurant sur carnet où dont les éléments de matière sont libres à l'instant t_i , on calculera :

$$D_{P_s, \theta_k} = \lambda \left(q C_{P_s}^{t_i} + (1 - q) K_{\theta_k}^{t_i} \right)$$

où θ_k est l'opération à subir par la matière attachée à P_s à l'instant t_i .

- * q est une constante comprise entre 0 et 1
 - * λ prend une valeur $v > 1$ si le produit est en cours et la valeur $\frac{1}{v}$ si le produit figure encore sur carnet
- ③ Pour chaque opération θ_k susceptible d'être effectuée à l'instant t_i , on retient la matière attachée à la commande P_s pour laquelle D_{P_s} est maximum.
 - ④ Si plusieurs machines libres sont en mesure d'effectuer l'opération θ_k à l'instant t_i , on choisit celle pour laquelle le temps de préparation est minimum. Ce choix dépend donc des opérations précédemment effectuées.
 - ⑤ Les opérations sont effectuées par ordre décroissant des coefficients D_{P_s}, θ_k .

Nous voyons que les variations de q permettent de modifier l'importance relative de l'urgence du produit et de la charge de l'opération. D'autre part, un accroissement de λ a tendance à freiner les lancements de commande.

Notons que les paramètres et les fonctions définissant l'état initial du système ont été générées au hasard entre des limites imposées (loi uniforme). Au début de la simulation, les valeurs λ et q sont données par l'opérateur.

L'ordinateur calcule alors les valeurs au temps t_0 des différentes fonctions présentées précédemment, puis se sert de ces valeurs pour décider des opérations à effectuer, des machines à utiliser, des commandes à considérer etc , et ceci à partir des critères que je viens de définir. La situation de l'ensemble est alors calculée au temps t_1 , temps au bout duquel la première machine en fonctionnement à l'instant t_0 , s'arrête. Alors les valeurs des différentes fonctions sont

recalculées à partir des mêmes valeurs de l et q et le processus recommence.

Cette simulation se poursuit jusqu'à ce que l'opérateur décide de l'interrompre.

Différentes expériences ont été faites avec des valeurs distinctes du couple (l, q) . Quelques valeurs caractérisant la suite des états observés pour chacun de ces couples ont été portées sur des graphes, et nous les discuterons plus loin. Pour l'instant, nous allons présenter le programme de test comme nous l'avons fait pour le programme de la maquette de calculs.

3 - PRESENTATION DU PROGRAMME

La présentation est faite par niveaux et suit les règles énoncées précédemment. Ici encore, le détail n'a pas été poussé jusqu'aux niveaux ultimes de l'instruction, mais il est suffisant pour pouvoir reconstituer rapidement le programme quel que soit le langage utilisé.

F I C H I E R S

Les fichiers sont identiques aux fichiers du programme précédent. On trouvera cependant en plus le fichier suivant :

FICHER ' C O E '

Il contient, pour chaque opération, le coefficient ' rapidité d'exécution ' qui est, comme nous l'avons dit, un résultat statistique.

P R O G R A M M E

NIVEAU I Page 1

- 1) Donnée des paramètres de la règle d'ordonnancement et initialisation des fonctions.
(programme E N 1)

- 2) Fractionnement éventuel des commandes introduites durant la période précédente.
(programme C A C E)

- 3) Mise en oeuvre de la règle d'ordonnancement.
(programme U P)

- 4) Recherche de la période Δt et calcul de la situation à l'instant $t + \Delta t$.

- 5) Enregistrement à l'instant $t + \Delta t$ des commandes passées entre t et $t + \Delta t$.

2) Fractionnement éventuel des commandes introduites durant la période précédente

1) Pour toutes les commandes introduites durant la période précédente

2) Lecture dans ' P I A ' et ' P I B ' des paramètres et fonctions caractérisant cette commande

3) Si plus d'un chemin de fabrication est donné

4) Recherche du chemin le plus court

5) Lecture dans ' P II A ' et ' P II B ' des paramètres et fonctions caractérisant ce chemin

6) Fractionnement éventuel de la commande.

3) Mise en oeuvre de la règle d'ordonnement

1) Pour chaque commande

2) Lecture dans ' P I B ' des fonctions caractérisant la commande

3) Si la commande figure sur carnet
ou si la matière attachée à la commande est libre

4) Recherche du délai moyen

5) Recherche du temps de fabrication minimum

6) Calcul du coefficient ' urgence du produit '

7) Conservation de l'opération à effectuer en premier

8) Si non , attribution de la valeur 0 au coefficient ' urgence du produit '

9) Recherche, parmi les coefficient ' urgence ' non nuls, du maximum et du minimum

10) Pour chaque opération

11) Calcul du coefficient ' charge de l'opération '

12) Pour chaque commande

13) Calcul de la valeur donnée par la règle d'ordonnement.

14) Classement des commandes par ordre décroissant des valeurs données par la règle d'ordonnement

15) Pour chaque commande, dans l'ordre trouvé

16) Si la valeur donnée par la règle d'ordonnement n'est pas nulle et si aucune commande n'a encore été couplée avec l'opération qu'a à subir la commande envisagée

17) Couplage de la commande et de l'opération qu'elle a à subir

18) Recherche de la machine qui va effectuer cette opération.

4) Recherche de la période Δt et calcul de la situation à l'instant $t + \Delta t$

1) Pour chaque machine

2) Lecture dans ' M A III ' des fonctions caractérisant la machine

3) Si la machine n'est pas libre

4) On retient le temps d'occupation

5) On retient le temps de préparation

6) Recherche du temps d'occupation minimum Δt

7) Classement des temps de préparation par ordre croissant

8) Sortie du message : ' Période retenue : Δt '

9) Pour chaque intervalle de temps déterminé entre t et $t + \Delta t$ par les temps de préparation

10) Pour chaque machine occupée

12) Si la machine est en préparation durant cette période

13) Lecture dans ' M A I ' des paramètres caractérisant cette machine

14) Conservation du nombre d'ouvriers et du coût de préparation

Sinon

15) Lecture dans ' M A II ' des paramètres caractérisant cette machine

16) Conservation du nombre d'ouvriers nécessaires au fonctionnement.

- 17) Calcul du coût cumulé durant cette période
- 18) Calcul du nombre d'ouvriers employés durant cette période
- 19) Pour chaque machine
 - 20) Recherche de l'état de la machine à l'instant $t + \Delta t$
 - 21) Modification du fichier ' M A III '
- 22) Pour chaque commande
 - 23) Lecture des fonctions caractérisant cette commande dans ' P I A ' et ' P II B '
 - 24) Modification des délais actualisés dans ' P I B '
 - 25) Si la matière attachée à cette commande était en cours de passage sur une machine qui vient d'être libérée
 - 26) Réintégration de la matière attachée à cette commande dans l'ensemble de la matière libre
- 27) Calcul du temps moyen d'immobilisation
- 28) Recherche du poids total de stocks
- 29) Recherche du poids total d'en-cours.

5) Enregistrement à l'instant $t + \Delta t$ des commandes passées entre t et $t + \Delta t$

1) Pour chaque commande passée entre t et $t + \Delta t$

2) Si il existe des stocks de même nuance, forme et poids unitaire que le produit fini désiré et si le poids de ces stocks est supérieur ou égal au poids minimum de la commande

3) Sortie du message ' Commande X exécutée '

4) Modification du stock concerné dans ' S T '

5) Sinon , enregistrement de la commande dans ' P I A ' , ' P I B ' , ' P I I A ' et ' P I I B ' .

2 - 4) Recherche du chemin le plus court

1) Pour chaque chemin possible

2) Lecture dans ' P II A ' et ' P II B ' des paramètres et fonctions caractérisant ce chemin

3) Pour chaque étape de ce chemin

4) Lecture dans ' C O E ' du coefficient ' rapidité d'exécution ' de l'opération attachée à cette étape

5) Sommation de tous ces coefficients ' rapidité d'exécution '

6) Conservation du chemin correspondant à la somme minimum

7) Suppression des chemins inutilisés dans ' P I A ', ' P I B ', ' P II A ' et ' P II B '.

2 - 5) Fractionnement éventuel de la commande

1) Pour chaque étape de fabrication du chemin retenu

- 2) Calcul du poids de l'unité de matière en fin d'étape
- 3) Conservation de ce poids et du poids minimum total de la commande
- 4) Conservation de la forme de l'unité de matière en fin d'étape

5) Recherche, dans le fichier ' ST ' , du stock de même poids et de même forme que l'unité de produit fini

6) C1 : Ce stock est vide	O	N	N
C2 : Le poids total de ce stock est supérieur ou égal au poids minimum de la commande		O	N
7) Sortie du message ' un poids X de cette commande est déjà fabriqué '			1
8) Suppression dans ' ST ' du stock considéré			2
9) Modification dans ' P I B ' des poids maximum et minimum de la commande			3
10) Exploitation des autres étapes	1		4
11) Sortie du message ' Commande X déjà exécutée '		1	
12) Modification dans ' ST ' du stock considéré		2	
13) Suppression, dans ' P I A ' ' P I B ' ' P II A ' et P II B ' des paramètres et fonctions attachés à cette commande et à son chemin de fabrication		3	

3 - 5) Recherche du temps de fabrication minimum

1) Lecture dans ' P II B ' des caractéristiques du chemin de fabrication

2) Pour chaque étape du chemin de fabrication

3) Calcul du temps de passage de la matière par cette opération

4) Calcul du poids de matière attribuée à cette commande après passage par cette opération

3 - 11) Calcul du coefficient ' charge de l'opération '

- 1) Calcul du poids de matière ayant à subir cette opération
- 2) Lecture dans le fichier ' C O E ' du coefficient ' rapidité ' d'exécution de l'opération
- 3) Calcul du coefficient demandé

3 - 18) Recherche de la machine qui va effectuer cette opération1) Pour chaque machine

2) Lecture dans ' M A III ' des valeurs des fonctions caractérisant la machine

3) Lecture dans ' O M ' des codes des opérations qu'est susceptible d'effectuer cette machine

4) Si la machine est libre et si elle est susceptible d'effectuer l'opération à subir par la matière attachée à la commande

5) Lecture dans ' M A I ' du temps de préparation

6) Si un temps de préparation au moins a été retenu

7) Recherche du temps de préparation minimum

8) Choix de la machine correspondant à ce minimum

9) Modification dans ' M A III ' des fonctions caractérisant l'état de cette machine

10) Modification dans ' P I B ' des fonctions caractérisant la commande

11) Sortie du message ' Commande X sur machine Y '.

4 - 26) Réintégration de la matière attachée à cette commande dans l'ensemble de la matière libre

- 1) Modification des fonctions dans ' P I B '
- 2) Suppression dans ' P II B ' de l'étape qui vient de se terminer
- 3) Si cette étape était la dernière du chemin de fabrication
 - 4) Sortie du message ' Commande X terminée '
 - 5) Sortie des délais actualisés (négatifs si délais dépassés)
 - 6) Calcul de l'écart et de l'écart moyen pondéré
 - 7) Suppression dans ' P I A ' , ' P I B ' , ' P II A ' et ' P II B ' des fonctions et paramètres attachés à cette commande
 - 8) Recherche des commandes faisant partie de la même commande initiale et modification des fonctions ' fractionnement ' (Fichier ' P I B ')
- 9) Sinon, modification des fonctions de ' P II A '
- 10) Stockage dans ' ST ' de l'excédant à ce niveau.

2 - 5 - 10) Exploration des autres étapes

1) Pour chaque étape, en commençant par l'avant dernière

2) Recherche, dans le fichier ' ST ', du stock de même poids unitaire et de même forme que l'unité de matière à ce niveau

3) Si ce stock n'est pas vide

4) Si le poids total de ce stock est supérieur ou égal au poids minimum de la matière attachée à la commande à ce niveau

5) Modification dans ' P I B ' des fonctions caractérisant la commande

6) Modification de ' P II A '

7) Suppression dans ' P II B ' des étapes précédentes

8) Fin d'exploration des étapes

Sinon

9) Sortie du message ' fractionnement de la commande au niveau X '

10) Suppression dans ' ST ' du stock considéré

11) Modification dans ' P I B ' des poids minimum et maximum de la commande initiale

12) Enregistrement dans ' P I A ', ' P I B ', ' P II A ' et ' P II B ' des paramètres et fonctions caractérisant la nouvelle commande

13) Modification de la fonction ' nombre de commandes générées ' dans ' P I B '

4 - 26 - 10) Stockage dans ' ST ' de l'excédant à ce niveau

1) Recherche de l'excédant de poids de matière attribuée
à la commande à ce niveau

2) Si cet excédant de poids est supérieur au poids d'un
élément de matière à ce niveau

3) Recherche du nombre maximum d'éléments de
matière dont le poids total est inférieur
à l'excédant

4) Stockage de ces éléments : modification de ' ST '

5) Modification du poids actuel dans ' P I B '

4 - CONCLUSIONS TIRÉES DE CES EXPERIENCES

Rappelons que toutes les expériences ont été faites à partir de commandes dont les paramètres et les valeurs initiales des fonctions ont été générés au hasard, entre des bornes données et suivant des lois uniformes.

Les résultats principaux ont été portés sur les onze premiers graphes ci-joints et nous allons les passer en revue.

a) Temps moyen d'immobilisation

Rappelons que le temps moyen d'immobilisation des machines englobe le temps de préparation et le temps durant lequel les machines ne se voient attribuer aucune tâche.

Lorsque λ passe de 1 à 1,5, à q constant, le temps moyen d'immobilisation croît sensiblement. Cela s'explique de la manière suivante : lorsque $\lambda = 1,5$, les commandes sont lancées plus difficilement, et les en - cours ont priorité pour être affectés à une machine. Cela a pour conséquence des changements fréquents dans les opérations effectuées par les machines, donc des immobilisations fréquentes de ces machines.

Le phénomène est moins sensible lorsque λ passe de 1,5 à 10. Cela signifie que, lorsque $\lambda = 1,5$, la priorité des en - cours a atteint un niveau difficilement dépassable.

Par ailleurs, à λ constant, le temps moyen d'immobilisation diminue avec q ; donc ce temps diminue lorsque les charges des opérations prennent plus d'importance que les urgences des produits, et cela s'explique facilement : une opération étant très chargée, les machines s'y consacreront en priorité, donc auront à subir des préparations beaucoup plus rarement.

Nous remarquerons encore que si le temps moyen d'immobilisation diminue avec q , cette diminution est d'autant moins importante que λ est grand.

b) Poids et nombre d'en - cours

Nous constatons qu'à q constant, le poids et le nombre d'en - cours diminue lorsque λ augmente, ce qui ne nous surprend pas. De plus, l'augmentation est très faible lorsque λ passe de 1,5 à 10, alors qu'elle est très forte lorsque λ varie de 1 à 1,5. On peut donc considérer, comme précédemment, que la priorité des en - cours atteint pratiquement son plafond pour $\lambda = 1,5$.

Enfin, nous constatons que les en - cours et leur poids ont tendance à moins varier avec λ lorsque $q = 1$ que lorsque $q = 0,01$: une plus grande importance accordée à l'urgence des produits (au détriment de la charge des opérations) est un facteur stabilisateur pour ces fonctions.

c) Dispersion

A q constant, la dispersion augmente avec λ . Si donc on cherche à diminuer le nombre d'en - cours, il apparaît des écarts de plus en plus importants entre le temps de fabrication et le délais moyen.

Remarquons cependant que ces écarts sont plus importants lorsque $q = 0,01$ que lorsque $q = 1$: une plus grande importance accordée à l'urgence des produits est encore un élément stabilisateur dans ce domaine.

d) Ecart moyen pondéré

A q constant, cet écart moyen pondéré augmente avec λ . Par contre, à λ constant, il augmente lorsque q diminue.

Lorsque cet écart est positif, cela signifie qu'en moyenne les produits sont terminés avant les délais, et il n'y a que des avantages à

voir cet écart diminuer (nous sommes dans ce cas).

Si par contre cet écart était négatif, nous aurions intérêt à ce que cet écart augmente. L'idéal serait évidemment qu'il avoisine 0.

e) Remarques

Nous n'avons considéré, dans ces graphes, que les fonctions qui nous paraissent essentielles. Les résultats obtenus nous ont également permis d'observer :

- 1) Une plus grande fluctuation du nombre d'ouvriers employés lorsque ρ croit (ceci est lié à l'augmentation du temps moyen d'immobilisation)
- 2) Un mouvement moins important des stocks lorsque ρ croit et, de manière générale, un niveau de stocks plus élevé.

Quelle conclusions tirer des observations qui précèdent ?

Ces conclusions dépendront évidemment du but recherché, et, en tout état de cause, elle ne s'appliqueront qu'au système étudié. Nous allons esquisser quelques conclusions possibles.

f) Conclusions

Le système étudié est évidemment inscrit dans un contexte de basse conjoncture (en effet, les écart moyens sont positifs, ce qui signifie qu'en règle générale, les commandes sont terminées avant les délais).

Nous pouvons donc penser qu'il est souhaitable de retenir le couple ($\rho = 1$, $q = 1$) qui diminue, dans la période considérée, ces stocks de produits finis. Par contre, ce couple entraîne un poids d'encours très élevé. Pour diminuer celui-ci, il conviendrait au contraire de retenir le couple ($\rho = 10$, $q = 0,01$). La décision finale dépendra

des coûts respectifs des stocks des produits finis et des en - cours. Il est probable que l'utilisateur sera amené à retenir un compromis qui tiendra compte des contraintes de stockages.

En période de haute conjoncture, le couple ($\rho = 1$, $q = 0,01$) pourrait être envisagé, car il assure un temps d'immobilisation minimum pour l'ensemble des machines. Ce couple assure d'autre part une dispersion faible. Cependant, les en - cours sont importants. Encore dans ce cas, il faudrait certainement envisager un compromis à partir des résultats correspondant à cette période.

Nous voyons que les résultats fournis par les simulations n'imposent aucune politique. Ils permettent simplement de choisir une politique avec une idée précise de ses conséquences dans le contexte envisagé et pour la durée couverte par la simulation. En dernier ressort, la décision revient au chef de fabrication. Il est donc souhaitable que celui-ci participe au choix des fonctions qui caractérisent l'état du système.

II. - SECONDE REGLE : PRIORITE AUX COMMANDES LES PLUS ANCIENNES

Cette règle s'énonce comme suit :

- 1 On conserve, pour chaque commande, le chemin de fabrication (gamme de fabrication) qui assure à la matière attachée à cette commande le temps de passage minimum sur machine
- 2 Lorsqu'une commande peut être élaborée sur plusieurs machines, on choisit la machine nécessitant le temps de préparation minimum.
- 3 Pour chaque opération θ_k susceptible d'être effectuée à l'instant t_i , on retient la matière attachée à la commande la plus ancienne

Une telle règle peut être envisagée lorsque le respect des délais est très important.

2) Présentation du programme

Le fichier désigné par ' C O E ' utilisé précédemment n'est plus nécessaire. D'autre part, le schéma du programme est identique au précédent, car seule la règle change.

Nous allons présenter ce programme comme il a été fait jusqu'ici ; cependant, à partir du niveau II, nous ne répétons pas les modules restés inchangés.

- 1) Initialisation des fonctions
- 2) Fractionnement éventuel des commandes introduites durant la période précédente
(ce module a été développé dans le programme précédent)
- 3) Mise en oeuvre de la règle d'ordonnancement
- 4) Recherche de la période Δt et calcul de la situation à l'instant $t + \Delta t$.
(Ce module a été traité)
- 5) Enregistrement à l'instant $t + \Delta t$ des commandes passées entre t et $t + \Delta t$.
(ce module a été traité)

3) Mise en oeuvre de la règle d'ordonnement

1) Pour chaque machine

- 2) Lecture dans le fichier ' M A III ' des fonctions caractérisant l'état de la machine
- 3) Si la machine est libre à l'instant considéré, choix de la matière première à lancer.

3 - 3) Choix de la matière première à lancer

1) Pour chaque commande, dans l'ordre de son apparition

2) Lecture dans ' P I B ' , des fonctions caractérisant la commande

3) Si la matière attachée à la commande est libre
et si l'opération à subir peut être effectuée par la machine
et si la machine n'est ni en fonctionnement, ni en préparation

4) Modification dans ' P I A ' ,
' P I B ' , ' P II A ' et ' P II B '
des fonctions caractérisant l'état de la commande

5) Modification dans ' M A III ' des fonctions caractérisant l'état de la machine.

3 - 3 - 4) Modification dans ' P I A ' , ' P I B ' et P I I A '
et ' P I I B ' des fonctions caractérisant l'état
de la commande

- 1) Modification dans ' P I B ' de la fonction indiquant que la commande est en fabrication
- 2) Modification dans ' P I A ' du poids de la commande
(poids après passage par la nouvelle étape)
- 3) Modification de ' P I I A '
- 4) Suppression de la première étape dans ' P I I B '.

3 - 3 - 5) Modification dans ' MA III ' des fonctions caracté-
risant l'état de la machine

- 1) Conservation de l'opération précédemment effectuée
- 2) Modification de la fonction indiquant l'opération en cours ou préparée
- 3) Modification de la fonction ' temps de préparation '
- 4) Calcul du temps d'occupation
- 5) Modification de la fonction ' temps d'occupation '

3) RESULTATS ET CONCLUSIONS

Les graphes correspondant à cette règle sont désignés par A. On constate que le temps moyen d'immobilisation a tendance, dans le cas qui nous intéresse, à se stabiliser beaucoup moins rapidement qu'au cours des simulations précédentes. La stabilisation se fait à un niveau très proche de celui que nous avons observé pour le couple ($\ell = 10$, $q = 0,01$), bien que, maintenant, nous ne donnions aucune priorité particulière aux ' en - cours '. L'accroissement de la dispersion est également comparable aux observations faites pour le couple ($\ell = 10$, $q = 0,01$). L'écart moyen pondéré reste aussi au même niveau. Par contre, les poids des en - cours augmentent plus rapidement que pour $\ell = 10$ et $\ell = 1,5$, mais moins rapidement que pour $\ell = 1$.

En résumé, cette règle, du fait de la dispersion élevée qu'elle introduit, ne semble pas très favorable.

Pour la période recouverte par la simulation, il semblerait que la règle associée au couple ($\ell = 10$, $q = 0,01$) soit plus performante et entraîne une dépense de stockage moindre ; mais nous n'avons aucun moyen de désigner celle de ces deux règles qui nous place en l'état \mathcal{E}_n le plus favorable pour la suite.

III) - Troisième règle : priorité aux commandes les plus importantes

Cette troisième règle a pour but de satisfaire en priorité les commandes les plus importantes en poids, dans une certaine mesure, les clients les plus importants.

1 - Exposé de la règle

Comme précédemment :

- ① On conserve, pour chaque commande, le chemin de fabrication qui assure à la matière attachée à cette commande le temps de passage minimum sur machine
- ② Lorsqu'une commande peut être élaborée, à un instant donné, sur plusieurs machines, on choisit la machine nécessitant le temps de préparation minimum.

Par contre :

- ③ Pour chaque opération O_k susceptible d'être effectuée à l'instant t_i , on retient la matière attachée à la commande la plus importante en poids.

2 - Présentation du programme

Le niveau I est identique au précédent. Seul, le module III se développe différemment et c'est à ce module que nous allons nous attacher. Par ailleurs, nous utiliserons encore les mêmes fichiers.

3) Mise en oeuvre de la règle d'ordonnancement

1) Pour chaque commande

2) Lecture dans ' P I A ' , ' P I B ' , ' P I I A ' et ' P I I B ' des paramètres et fonctions caractérisant la commande et son état

3) Si la matière attachée à la commande est en cours de transformation à l'instant considéré

4) On attribue la valeur 0 au poids de sortie

sinon

5) Calcul du poids moyen de sortie

6) Conservation de l'opération à subir par cette commande

7) Classement des commandes par ordre décroissant de poids moyen de sortie

8) Pour chaque commande, dans l'ordre du classement

9) Si le poids moyen de sortie n'est pas nul

10) Recherche d'une machine susceptible de prendre en charge la matière première attachée à cette commande.

3 - 10) Recherche d'une machine susceptible de prendre en charge la matière première attachée à cette commande

1) Pour chaque machine

2) Lecture dans ' M A III ' des fonctions caractérisant l'état de la machine

3) Si la machine est libre à l'instant considéré
et si elle peut recevoir la matière première envisagée

4) Modification de ' M A III '

5) Modification des fonctions caractérisant la commande

3 - Résultats et conclusions

Les graphes correspondant à cette règle sont notés B.

Un résultat particulièrement inattendu mérite d'être commenté : le temps moyen d'immobilisation est extrêmement élevé. Compte tenu du fait que les commandes traitées sont plus importantes, le temps de fonctionnement des machines entre deux arrêts consécutifs est grand en moyenne, donc on est en droit de s'attendre à un temps moyen d'immobilisation beaucoup moins élevé, au moins en début d'expérience. Le phénomène s'explique de la manière suivante : la répartition des opérations que sont susceptibles d'effectuer les machines est tel qu'il arrive que l'une d'elles ne puisse être alimentée. L'immobilisation forcée de cette machine ne pourra cesser que lorsqu'une autre machine s'arrêtera et fournira la matière première attendue, et dans notre cas l'attente est longue. Si bien que, si le temps moyen de préparation diminue effectivement, le temps moyen durant lequel les machines ne sont pas utilisées augmente dans de telles proportions que le temps moyen d'immobilisation croît considérablement. Un tel résultat pouvait difficilement être prévu et montre clairement l'intérêt de la simulation. Nous constatons également que la dispersion croît beaucoup plus vite qu'au cours des simulations précédentes, et cette constatation n'est pas favorable. De même, l'écart moyen pondéré est à un niveau qui dépasse tout ce qui a été observé jusqu'ici. Du fait des temps d'immobilisation importants des machines durant la période étudiée, il est possible que le retard accumulé dans la fabrication entraîne une diminution de cet écart dans la suite. Il n'en reste pas moins vrai que le niveau auquel se situe l'écart moyen pondéré n'est pas propre à nous faire choisir cette règle.

Enfin, le poids des en - cours est inférieur au poids observé lors de l'application de la règle A ou de la première règle pour $\lambda = 1$. Il est cependant nettement supérieur à ce qui a été observé pour $\lambda = 1,5$. Il présente par ailleurs un caractère de régularité qui peut s'expliquer par le fait que la matière en - cours correspond à des commandes de tailles voisines. Ici encore, la décision reste au chef de fabrication.

C O N C L U S I O N S

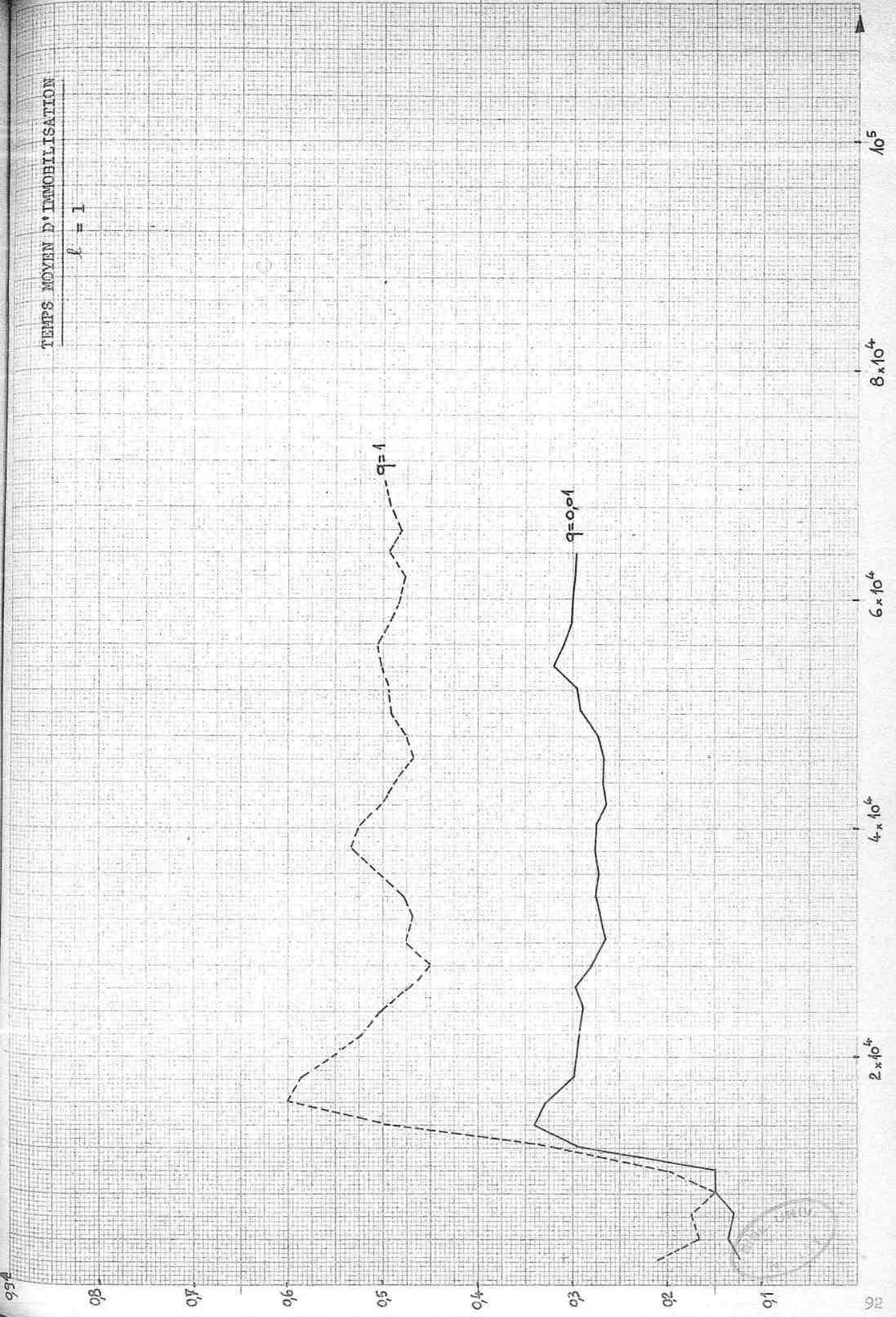
Un certain nombre de conclusions me semblent s'imposer à la suite de l'étude qui vient d'être faite : les problèmes d'ordonnement sont, dans la pratique, d'une complexité telle qu'il serait vain d'essayer d'utiliser les méthodes habituelles du calcul opérationnel. Seule une maquette construisant une image aussi fidèle que possible de la réalité, peut nous donner les éléments qui nous permettront d'agir sur le système en connaissance de cause.

Au niveau de la construction de la maquette, les difficultés sont énormes et souvent, des mois se passent avant que l'on puisse dégager les paramètres qui définissent le système et les fonctions qui la caractérisent. Ces difficultés sont d'autant plus grandes que les usines sont fréquemment une simple juxtaposition d'ateliers, plus ou moins complémentaires, et créés au fur et à mesure des besoins. A cela s'ajoute souvent une tradition, solidement ancrée et qu'il faut à tout prix préserver. Dès que le stade de construction de la maquette est achevé, une seconde difficulté surgit : que souhaite-t-on ? S'agit-il de réduire les stocks ? De satisfaire les délais ? D'assurer un emploi régulier ? Toutes ces questions doivent être posées à l'utilisateur, non pas que l'on attende une réponse, mais elles permettent de dégager les fonctions qui caractériseront l'état du système. L'utilisateur pourra ensuite, à son gré, tester ses décisions ou ses politiques d'ordonnement.

Mais il ne devra en aucun cas perdre de vue que les résultats obtenus ne sont significatifs que durant la période couverte par les simulations et, surtout, que ces résultats dépendent du système et de son état initial. Moyennant ces précautions il pourra obtenir des renseignements qui lui seront des plus utiles.

TEMPS MOYEN D'IMMOBILISATION

$l = 1$



0.95

0.8

0.7

0.6

0.5

0.4

0.3

0.2

0.1

0.0

2 x 10⁴

4 x 10⁴

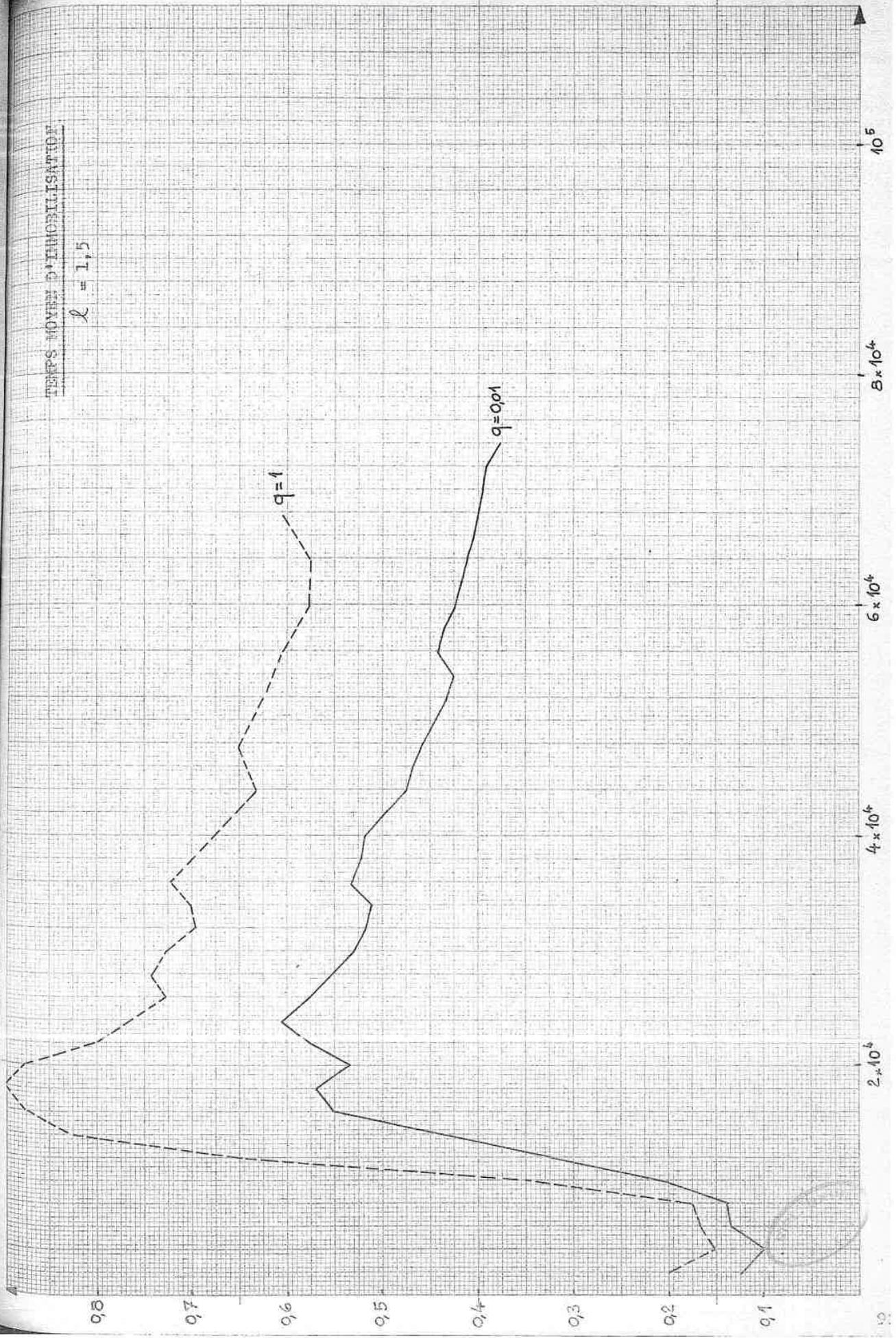
6 x 10⁴

8 x 10⁴

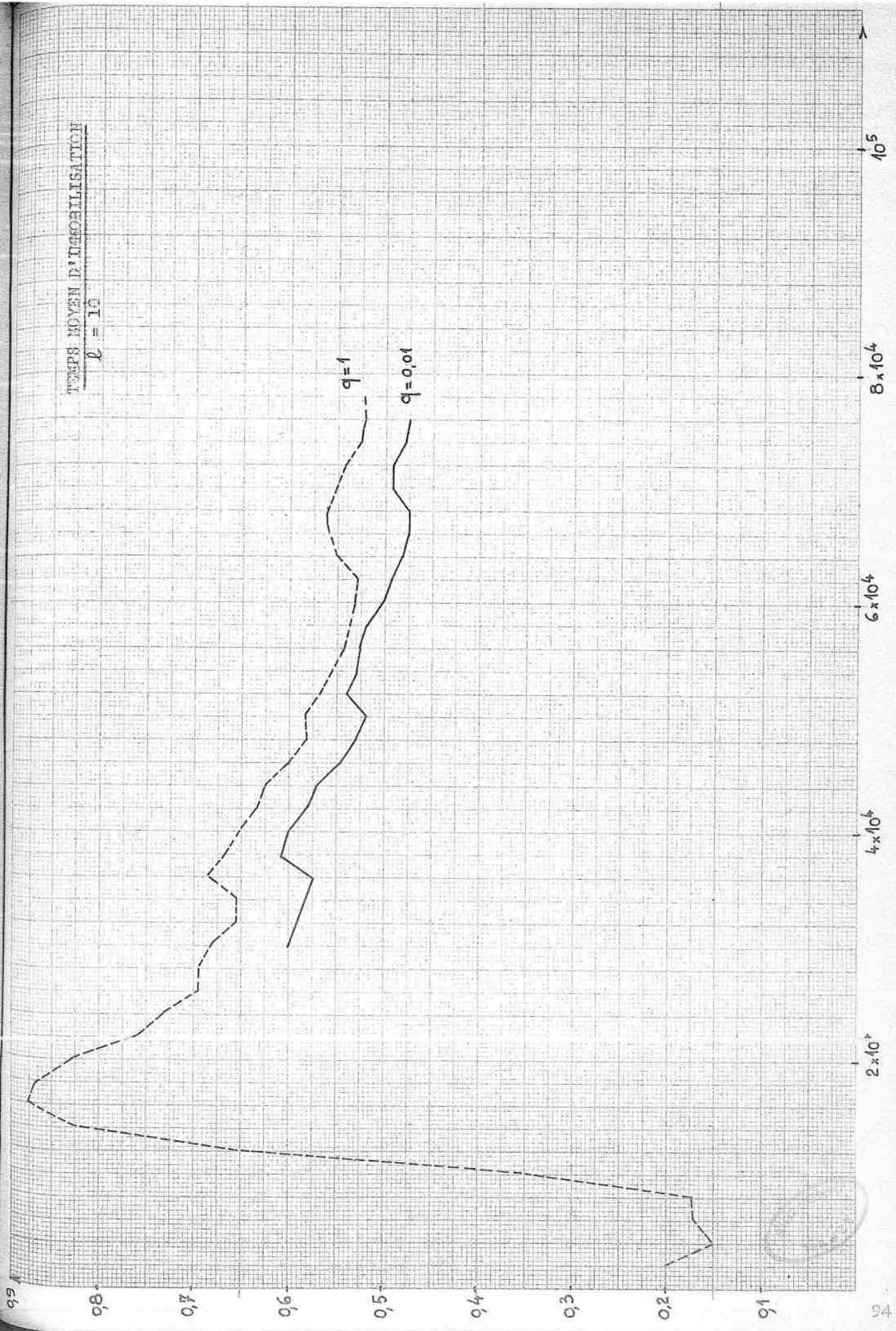
10⁵

TEMP'S MOVING D'INCORPORISATION

$\lambda = 1,5$

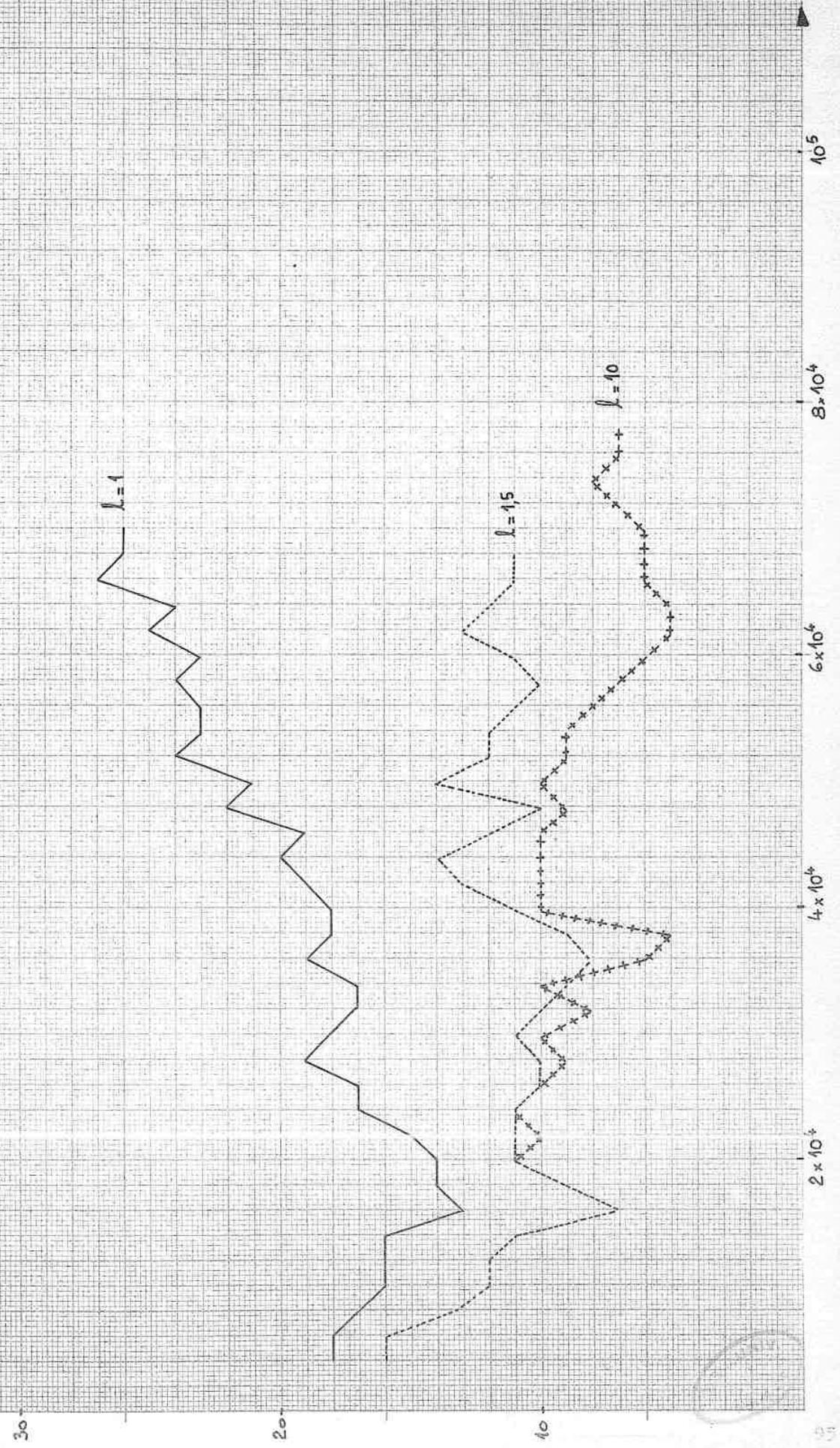


TEMPS MOYEN D'IMMOBILISATION
 $\bar{L} = 10$



REVUE D'EN COURS

$\alpha = 1$



NOMBRE D'EN COURS

$q = 0,01$

$l = 1$

$l = 15$

$l = 10$

30

20

10

10⁵

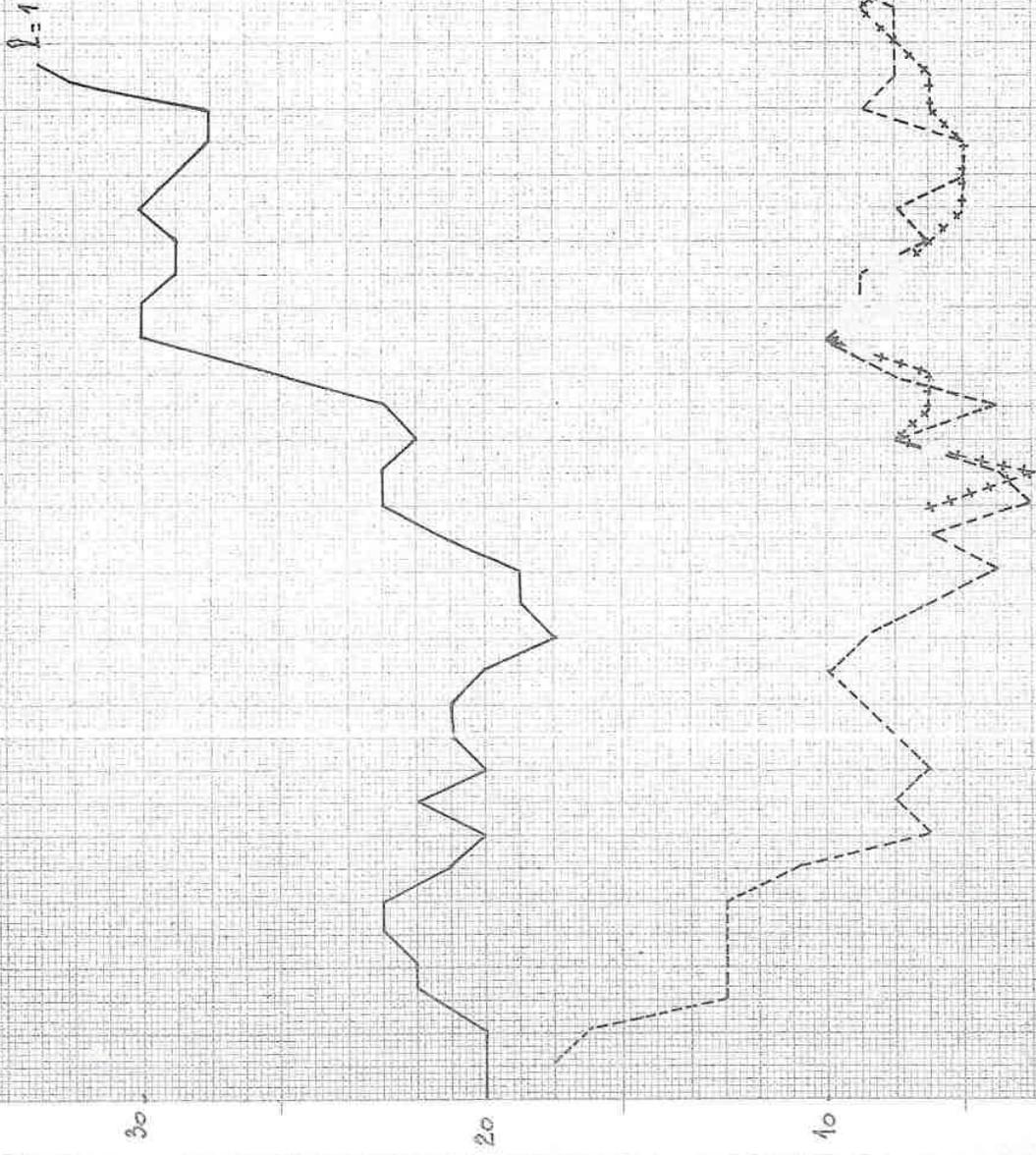
8 x 10⁴

6 x 10⁴

4 x 10⁴

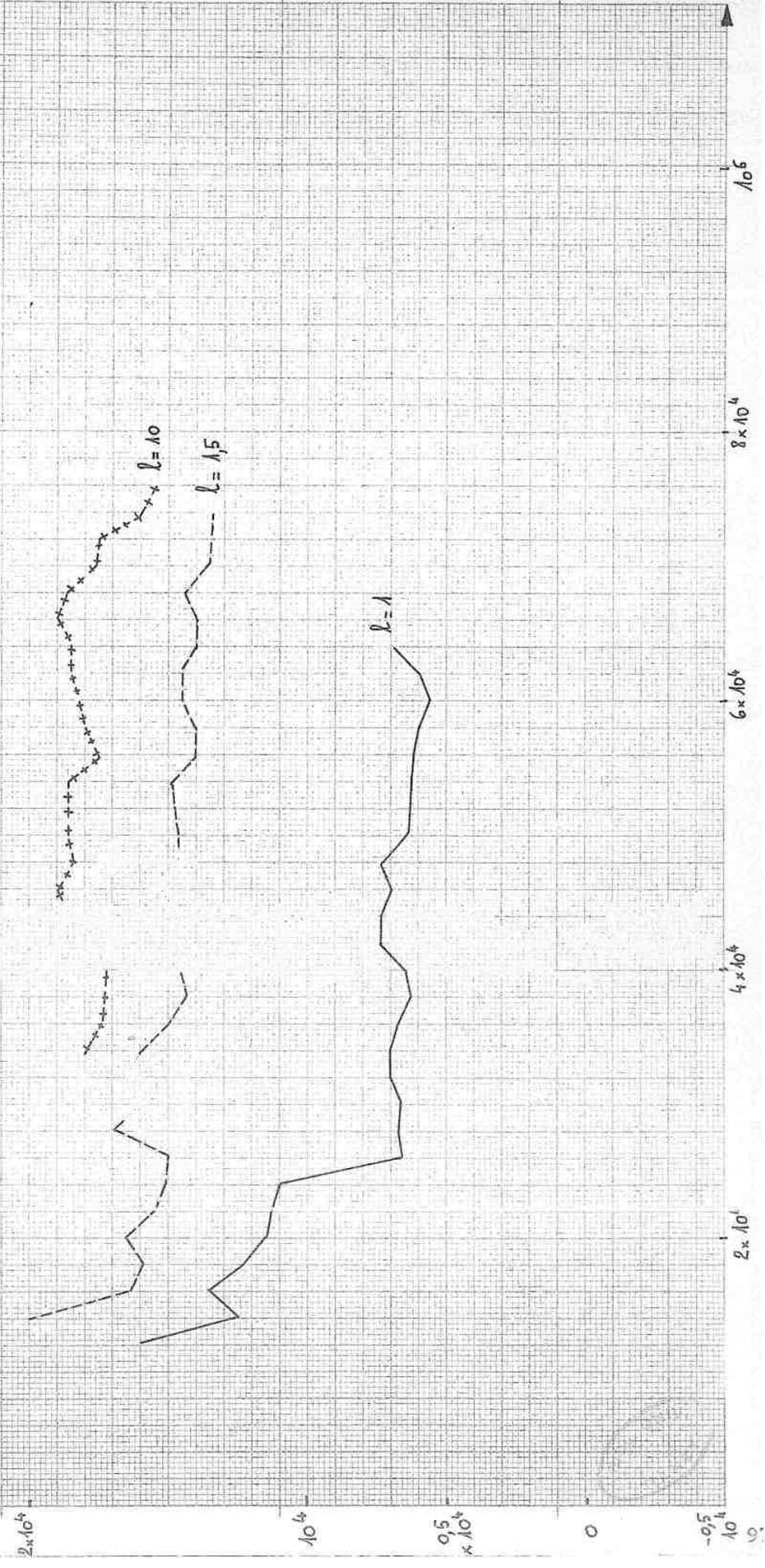
2 x 10⁴

96



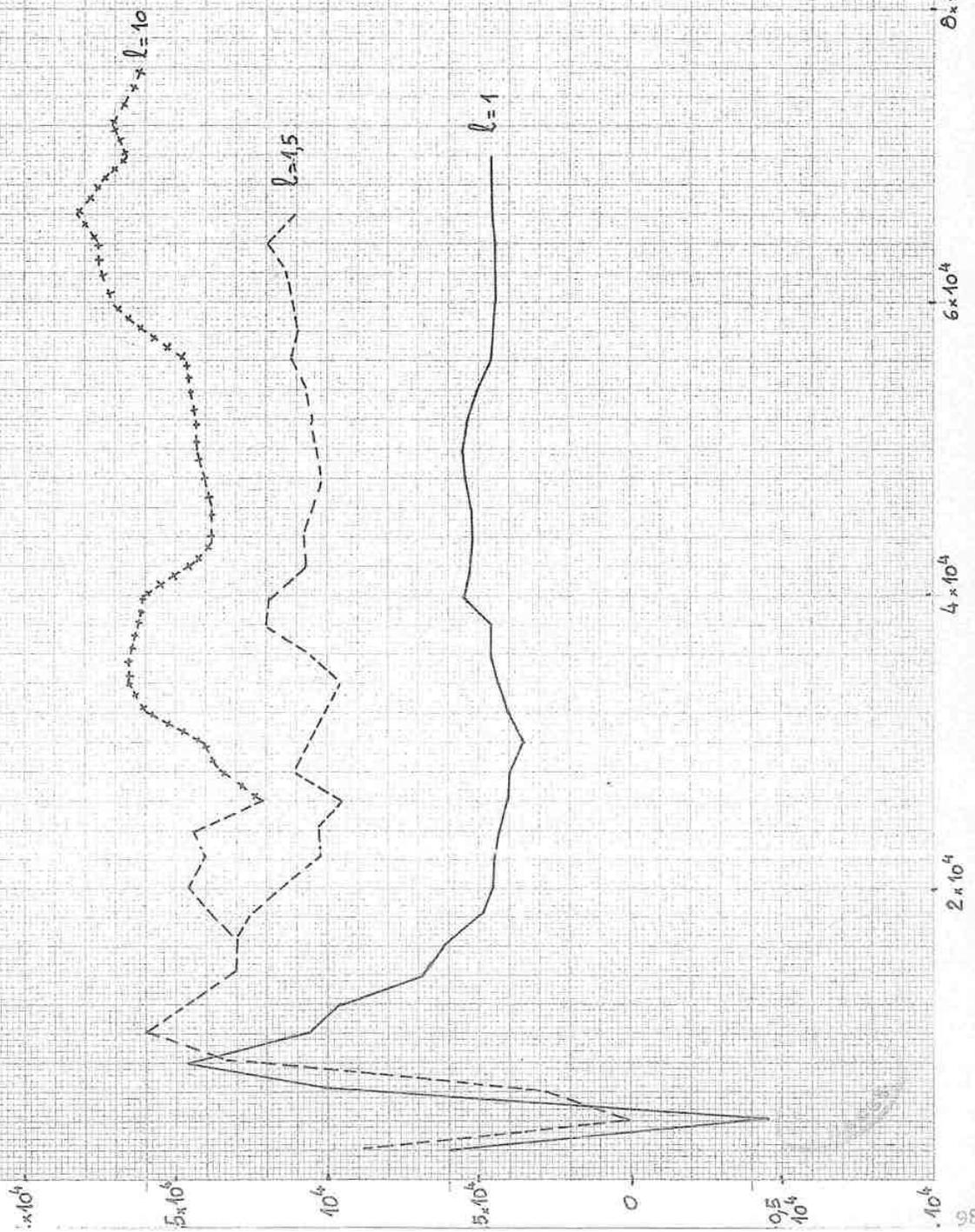
ECART MOYEN PONDERE

$$q = 0,01$$



ECART MOYEN PONDERE

$q = 1$



DISPERSION

$q = 1$

3×10^6

2×10^6

10^6

0

$l = 10$

$l = 15$

$l = 1$

10^5

8×10^4

6×10^4

4×10^4

2×10^4



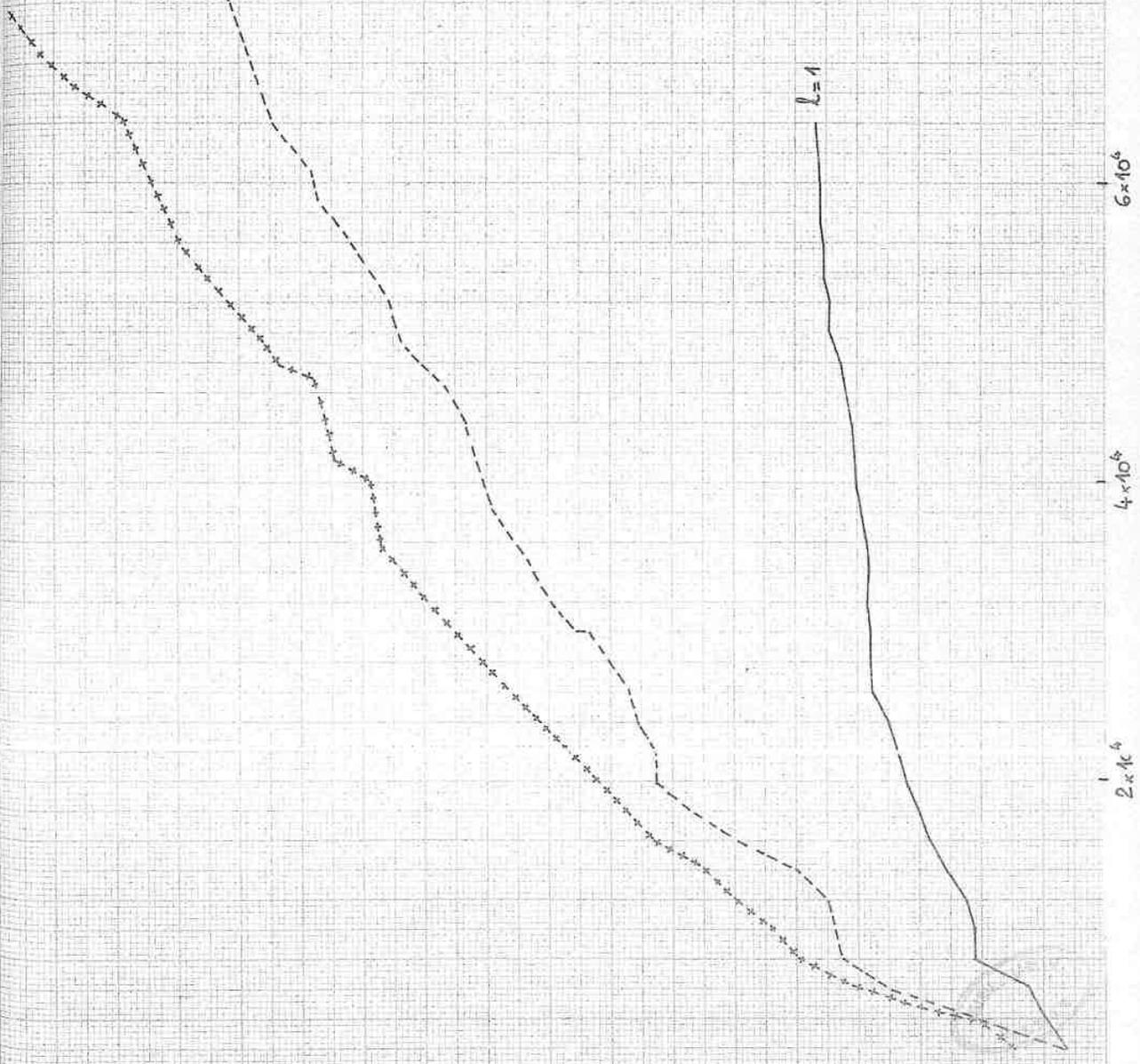
DISPERSION

$$q = 0,01$$

$l = 10$

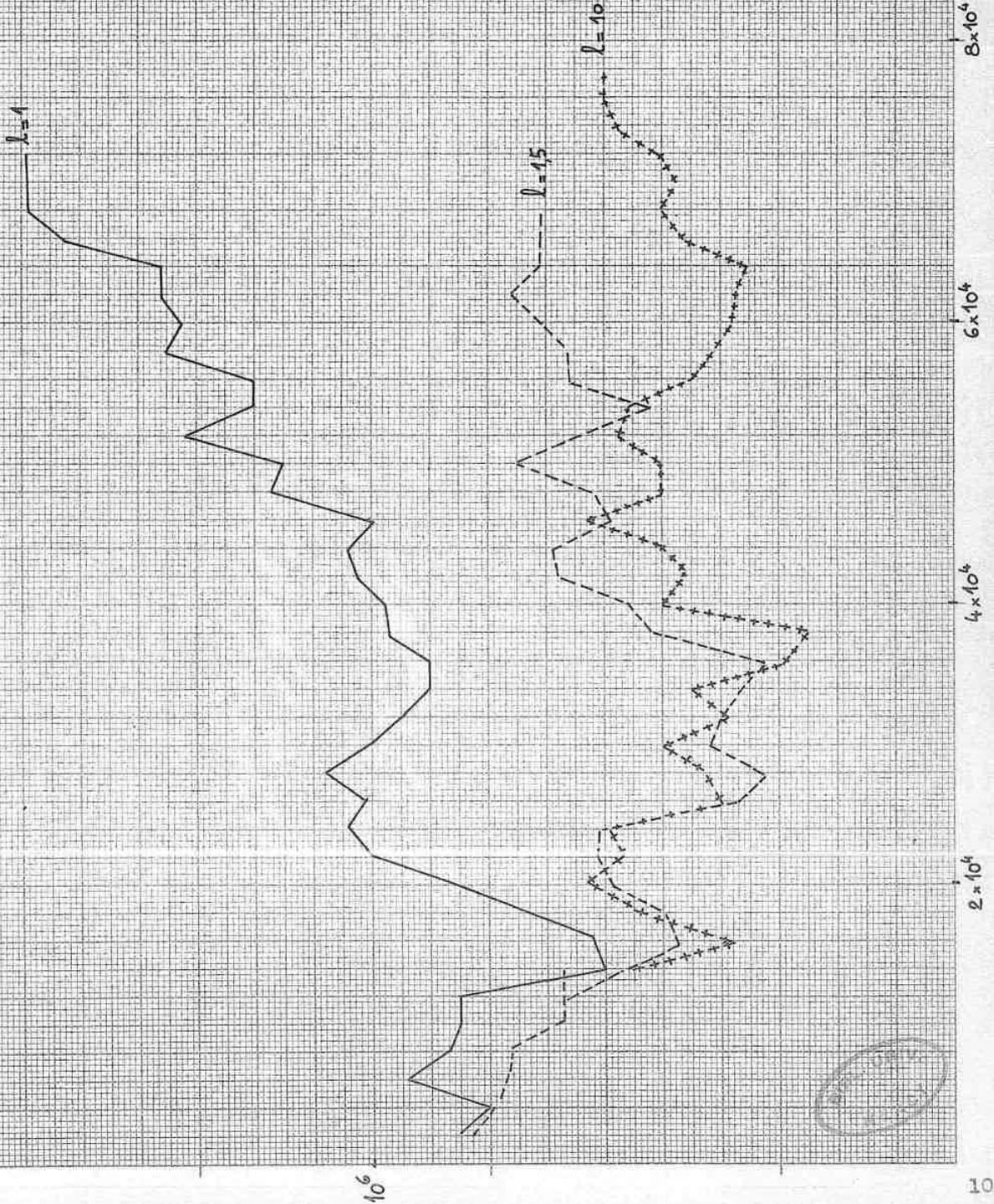
$l = 1,5$

$l = 1$



POIDS DES EN-COURS

$q = 1$



POIDS DES EN COURS

$$q = 0,01$$

$l=1$

$1,5 \times 10^6$

10^6

$0,5 \times 10^6$

102

$l=15$

$l=10$

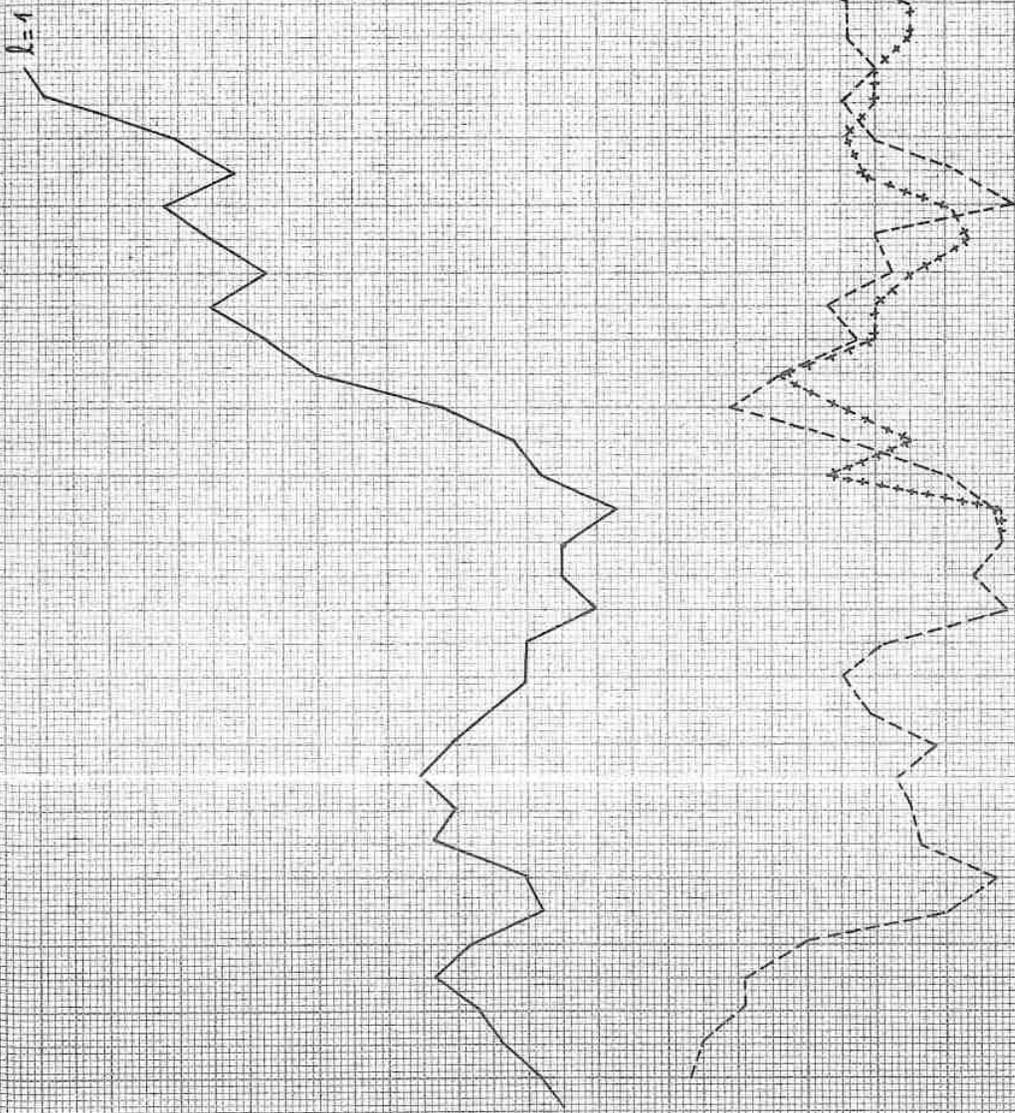
10^5

8×10^4

6×10^4

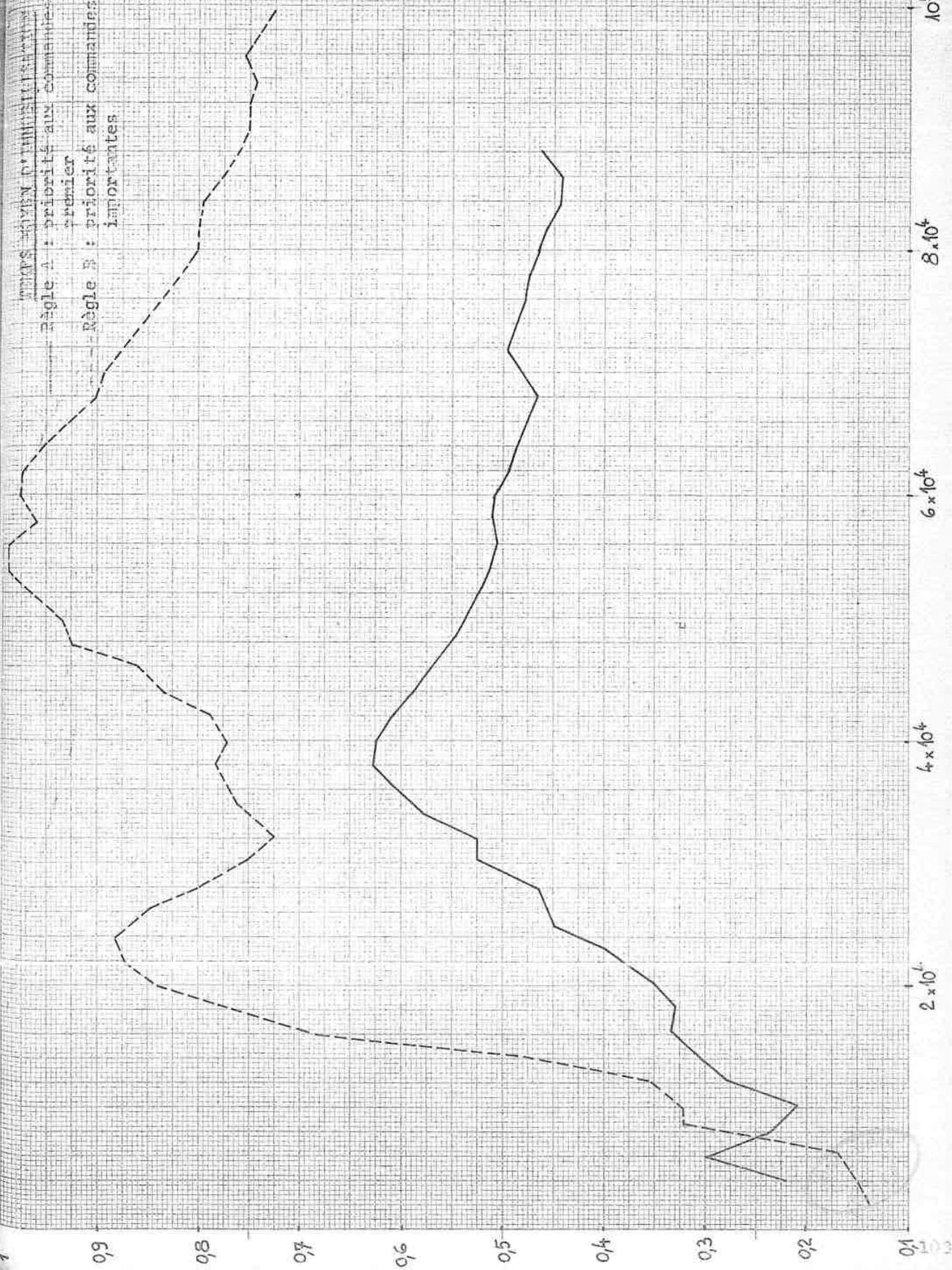
4×10^4

2×10^4



STATS MOYEN COMMUNISTES

Règle A : priorité aux commandes passées en premier
Règle B : priorité aux commandes les plus importantes



EGART MOYEN PONDERE

— série A

- - - - - série B

5×10^4

4×10^4

3×10^4

2×10^4

10^4

0

100

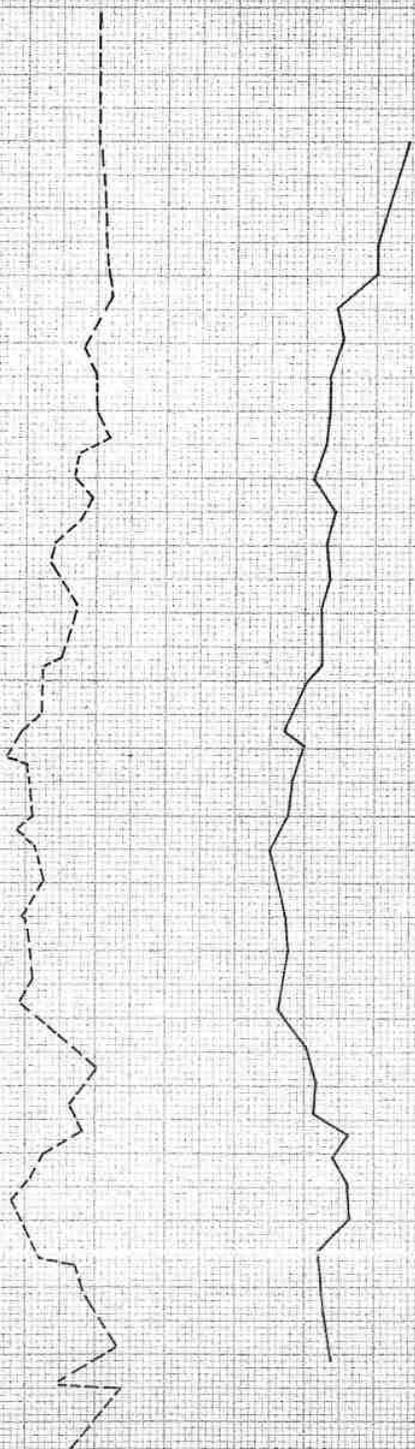
2×10^4

4×10^4

6×10^4

8×10^4

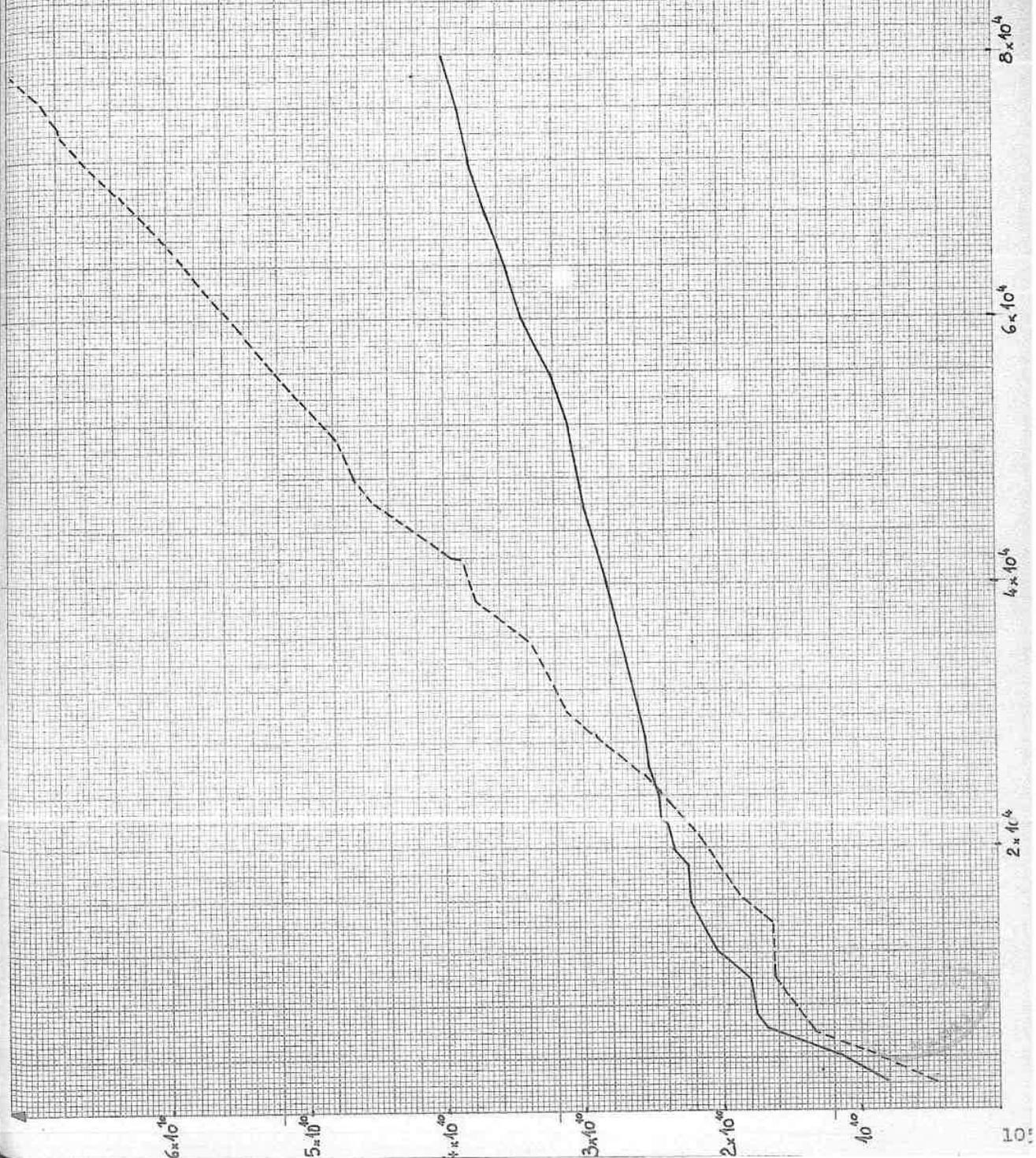
10^5



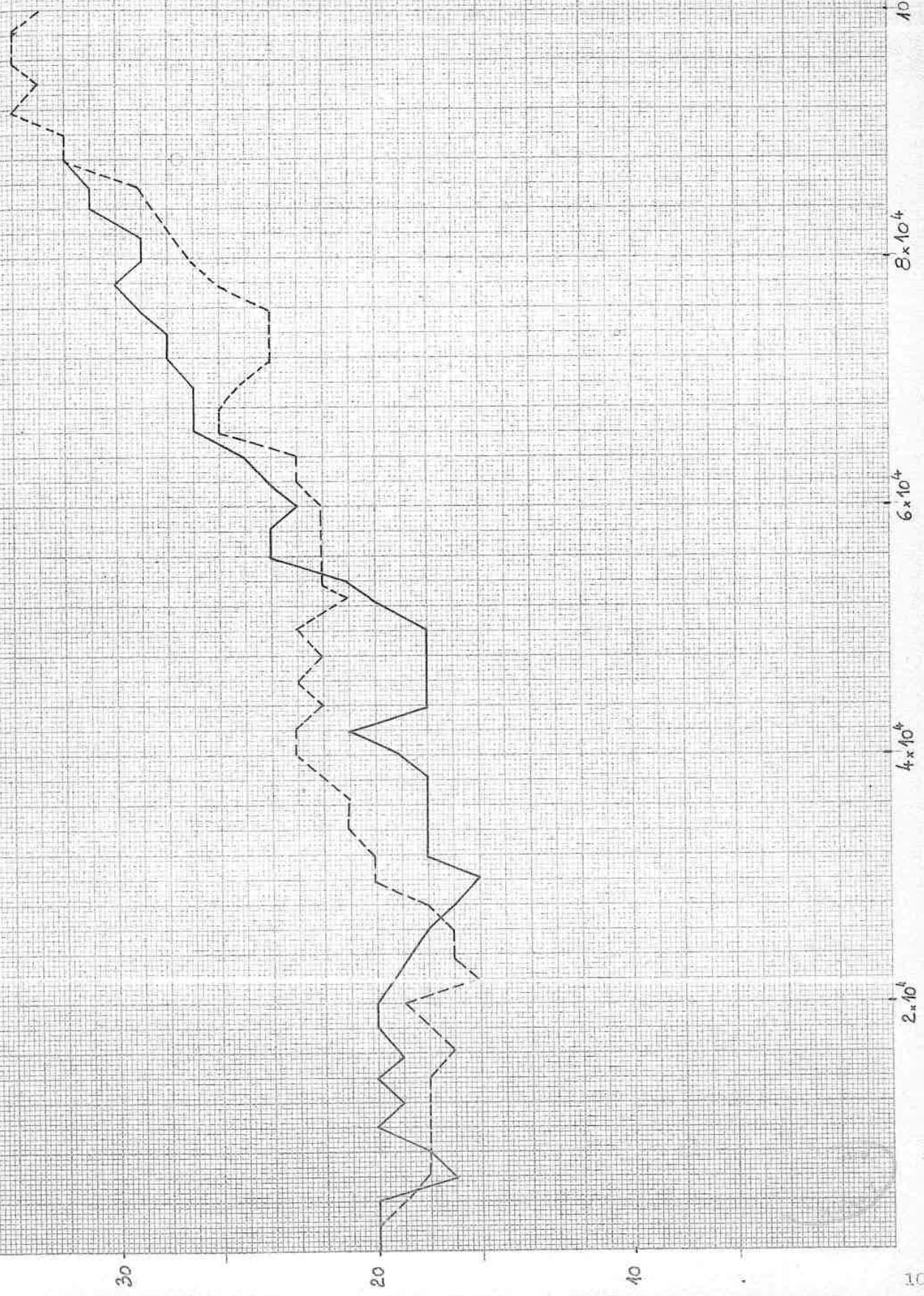
DISPERSION

— Règle A

- - - Règle B



région B

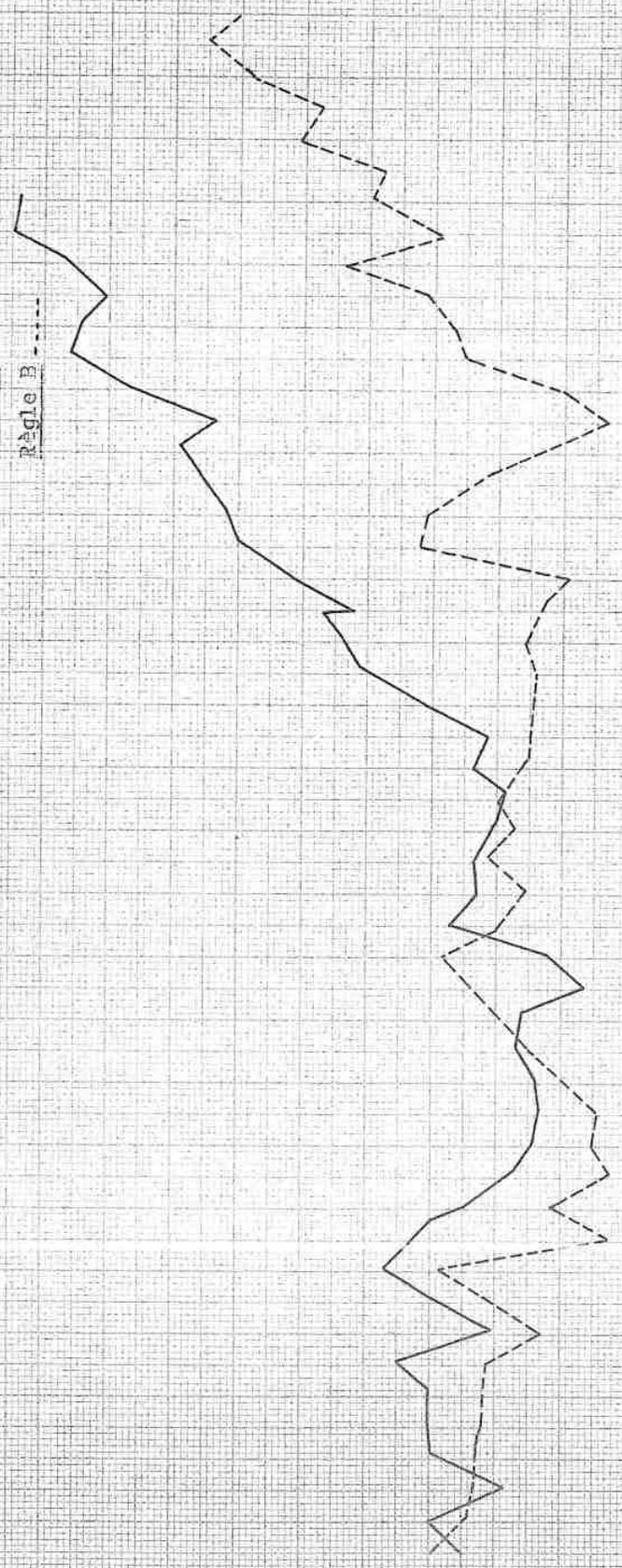


POIDS DES EN COURS EN FONCTION DU TEMPS

Rigle A —
Rigle B - - - -

$1,4 \times 10^6$
 $1,2 \times 10^6$
 10^6
 $0,8 \times 10^6$
 $0,6 \times 10^6$

10^5
 8×10^4
 6×10^4
 4×10^4
 2×10^4
 10^4





Vu, Approuvé
et permis d'imprimer
Nancy le 18 Décembre 1970.
Le Doyen

J. AUBRY.