Dactyl
B

UNIVERSITE DE NANCY I

U.E.R. DE SCIENCES MATHEMATIQUES

Double Sc. N 72/21

RECHERCHE DE REGLES D'AFFECTATION ET D'ORDONNANCEMENT POUR UN SYSTEME DE PONTS-ROULANTS

THESE

pour l'obtention du

DOCTORAT de SPECIALITE MATHEMATIQUES (3ème CYCLE)

Soutenue devant le Jury le 23 Mars 1972

UNIVERSITE NANCY BIBLIOTHEQUE

par

PIERRE LADURE

7

Jury : Mr LEGRAS J, Président
Mr DEPAIX M, Examinateur
Mr LEPLAT J, Examinateur

UNIVERSITE DE NANCY I U.E.R. DE SCIENCES MATHEMATIQUES

RECHERCHE DE REGLES

D'AFFECTATION ET D'ORDONNANCEMENT

POUR UN SYSTEME DE PONTS-ROULANTS

par PIERRE LADURE

Cette étude s'inscrit dans le cadre d'une recherche lancée sur invitation par la Direction des Affaires Sociales de la Commission des Communautés Européennes.

Le thème général de cette recherche est :

"Structure et Organisation du Travail en relation avec la Sécurité".

Ce travail a été réalisé sous la Direction de
Monsieur le Professeur Legras, Directeur de l'Institut Universitaire de Calcul Automatique ; qu'il soit assuré de l'expression de ma profonde gratitude pour la bienveillance qu'il m'a toujours témoignée ainsi que pour les conseils et la formation à la recherche qu'il m'a dispensés.

Je remercie vivement Monsieur le Professeur Depaix pour l'honneur qu'il me fait en participant au jury.

Cette étude a été effectuée dans le cadre du Laboratoire de Psychologie du Travail de Paris sous la direction de Monsieur le Professeur Leplat ; qu'il trouve ici tous mes remerciements pour l'aide et les conseils qu'il m'a prodigués.

Je remercie l'Exploitation du CII 10070 de l'IUCA et les ingénieurs de la S. A. F. E. pour leur collaboration.

Je remercie également Mademoiselle Colin qui a assuré avec rapidité et soin la réalisation matérielle de cette thèse.

INTRODUCTION

Cette étude se place dans le cadre de recherches entreprises par le Laboratoire de Psychologie du Travail de Paris, sur invitation de la Commission des Communautés Européennes. Ces recherches portent sur l'Organisation d'un système technique et son incidence sur la Sécurité. Le système technique choisi pour la présente étude est un ensemble de quatre ponts-roulants qui effectuent des tâches de manutentention dans une aciérie.

A notre connaissance, aucune étude n'ayant été effectuée dans cette optique sur ce système technique nous ne disposions, comme seul élément de départ, que de la situation concrète des ponts telle qu'on l'a trouvée en allant l'observer ≪sur le terrain≫.

Il nous a d'abord fallu analyser le système choisi et définir aussi précisément que possible ses caractéristiques. Nous avons, alors, dégagé les paramètres principaux du système. L'application d'une méthode de reproduction graphique à ces paramètres principaux permet de représenter graphiquement toute activité de ce système de ponts et d'en avoir ainsi une meilleure connaissance.

Cependant nous avons voulu dépasser la simple phase descriptive. Dans ce but, nous proposons une expérimentation sur ordinateur du système technique. Cette expérimentation est différente de l'expérimentation numérique et différente également de la simulation, au sens classique, dans la mesure où on ne fait pas intervenir de phénomène aléatoire.

Nous avons voulu proposer, également, des méthodes de recherche d'une organisation, respectant des règles de sécurité, et destinées à améliorer cependant le fonctionnement général. Il n'est pas question d'optimisation, aucun critère de ce type n'ayant pu être dégagé, mais de recherche d'une aide apportée par l'ordinateur aux responsables de cette organisation.

Dans cette optique d'aide à la décision, nous présentons deux méthodes heuristiques, l'une d'Affectation, l'autre d'Ordonnancement. Ces méthodes font appel à des règles de bon sens, adaptées au système technique étudié et à la structure des tâches dont il a la charge.

Les méthodes présentées ont donné naissance à des programmes* écrits dans l'optique d'une possibilité de dialogue entre l'ordinateur et l'utilisateur. En effet, les programmes ne font pas que donner une solution au problème posé : ils établissent des documents à partir desquels l'utilisateur pourra soit en modifiant certains paramètres, soit en énonçant certaines règles et en les introduisant dans le programme, agir sur la solution proposée initialement par l'ordinateur, et, connaissant ainsi les conséquences des décisions qu'il faut prendre, choisir celles qui conviennent le mieux.

---

* Les programmes sont rédigés en FORTRAN IV et ont été mis au point sur le CII 10070 de l'Institut Universitaire de Calcul Automatique de Nancy.

CHAPITRE I

POSITION DU PROBLEME

A) Cadre de l'étude

Nous avons voulu prendre une aciérie comme terrain d'étude car les interpénétrations entre la sécurité et l'organisation du travail y sont importantes. Cette aciérie comporte deux grands systèmes :

1) Le système de production proprement dit formé de deux fours (l'un de 60T, l'autre de 45T), du four de préchauffage, d'un skip permettant de monter la poche de coulée à la tour de coulée continue, de deux fosses à lingots, de lignes de chemin de fer. Tous ces éléments sont situés dans une halle de près de 150 mètres de long (Voir schéma n° 1).

2) Le système de ponts roulants formé, lui, de quatre ponts roulants dont le chemin de roulement couvre toute la longueur de l'aciérie et permet ainsi au système-ponts d'effectuer une tâche en tout point de l'aciérie.

Le système-pont présente deux caractéristiques essentielles :

a) Les ponts sont disposés en série et utilisent le même chemin de roulement ce qui implique qu'ils ne peuvent pas se croiser. Ils sont donc dans un ordre immuable.

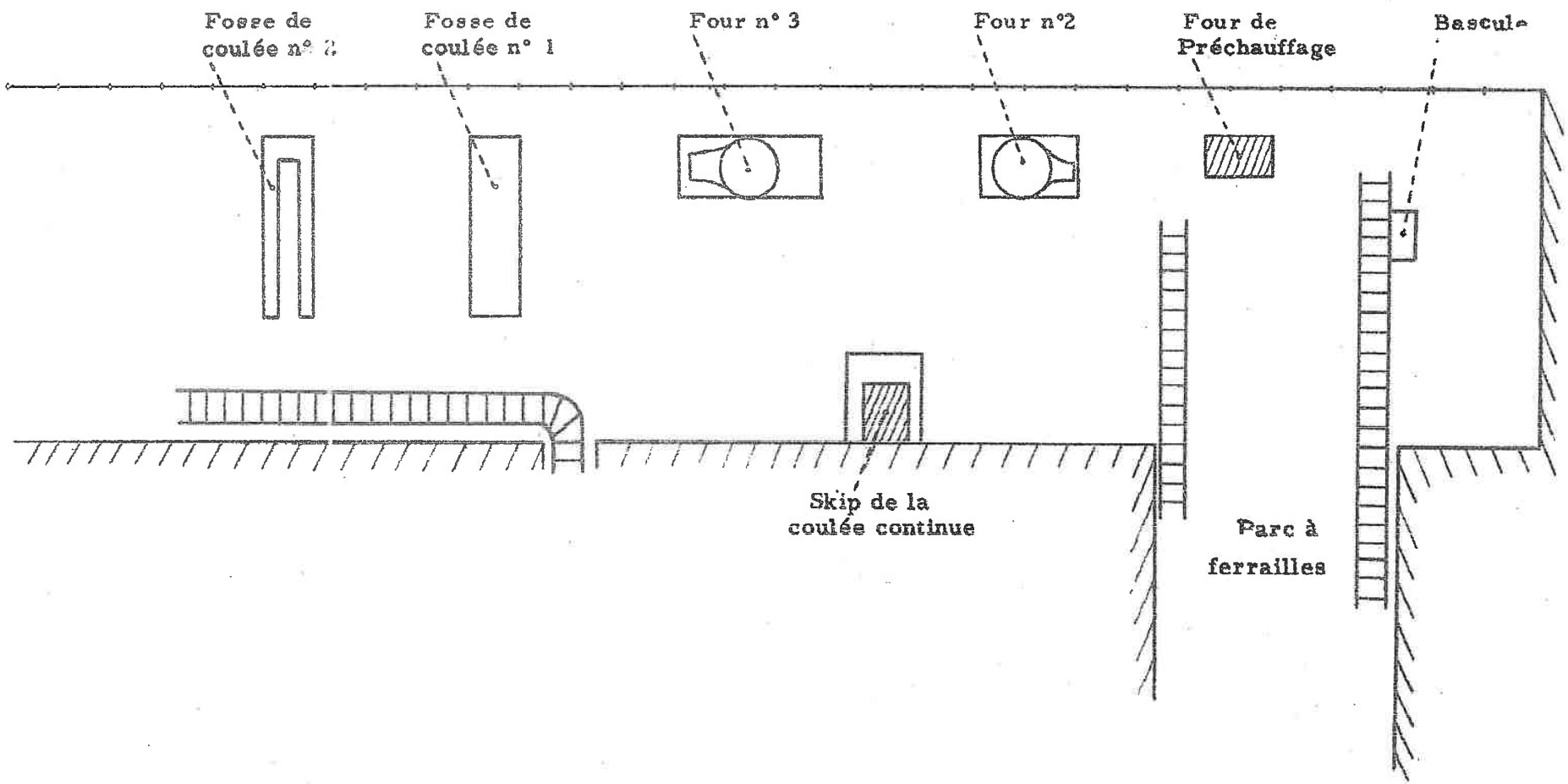


Schéma n° 1 : PLAN DE L'ACIERIE

b) Leurs espaces de travail se chevauchent, c'est à dire que les ponts ne disposent pas, chacun, d'une zone de travail réservée. Chaque pont peut être amené, pour son travail, à se rendre en tout point de l'aciérie.

Les problèmes de sécurité dans un tel système sont particulièrement aigus ; les risques de collisions entre deux ponts, les transports de charges dangereuses sur un long parcours sont fréquents dans ce domaine.

Notons que le système de production et le système ponts roulants n'ont pas d'existence indépendante. Les ponts roulants ne servent qu'à la réalisation de tâches faisant partie du processus technique effectué par le système de production . Ce processus technique vise à la fabrication de l'acier en quatre étapes :

E1 : La préparation du four (Mise de la dolomie, changement éventuel d'électrodes ...)

E2 : L'alimentation du four (Mise de paniers de ferrailles, de bennes de coke)

E3 : L'affinage (Additions éventuelles)

E4 : La coulée (Coulée dans les poches de coulée, mise au skip, mise dans les fosses à lingots...)

On retrouve chacune de ces quatre étapes pour les deux fours. L'ordre en est fixé une fois pour toute et au début de chaque semaine, après l'arrêt dominical, on fixe l'heure de déclenchement de la première étape pour un des fours et celle décalée de la première étape de l'autre four. Ce décalage initial dans la marche simultannée des deux fours est nécessaire car l'étape E4 de coulée ne peut avoir lieu en même temps pour les deux fours : pour des raisons technologiques de charge maximum un seul pont peut effectuer la première coulée. Sur cet exemple, apparaît nettement l'interdépendance de l'activité du système de production et du système-ponts (Voir schéma n° 2).

Afin d'avoir une vue complète de l'activité de chacun des deux fours et par là de l'activité des quatre ponts nous avons débuté nos observations lors de la coulée d'un four (le four II par exemple) et nous les avons arrêtées lorsque l'autre four coule pour la deuxième fois. Nous dirons que pendant ce temps s'est déroulé "un processus" (Voir schéma n° 2).

Un "processus" couvre donc la durée nécessaire à la réalisation de 2 coulées successives sur chacun des deux fours.

B) Etude du Système-Ponts

De façon générale le système-pont est dépendant du système de production. On peut définir deux modes de fonctionnement possible d'un système-pont selon les caractéristiques du processus auquel il collabore.

1°) Si le processus technique est contraignant, nous nous trouvons en présence d'un cas de fonctionnement "Sous guidage fort". L'opérateur chargé de l'ordonnancement des ponts a peu ou pas de décisions à prendre dans la mesure où l'ordre et le moment d'exécution de la plupart des tâches que le système-pont doit effectuer sont fixés à l'avance par le déroulement du processus technique.

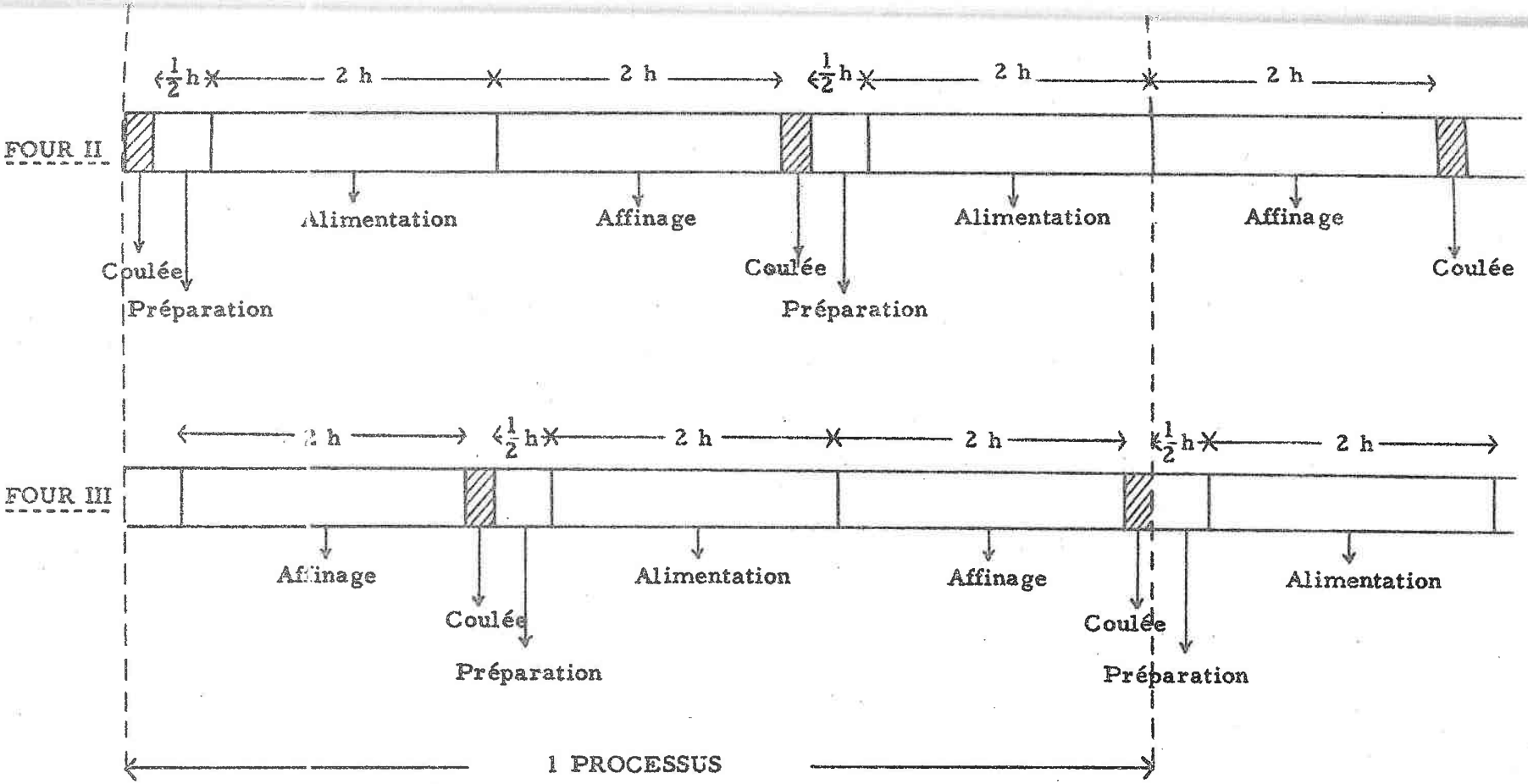


Schéma n° 2 :

LES PRINCIPALES PHASES DE LA MARCHE DES FOURS

2°) Si le processus technique est peu contraignant, c'est à dire qu'il n'existe aucun ordre pré-établi sur l'ensemble des tâches à réaliser, qu'aucune contrainte n'est imposée au système de production. On se trouve en présence d'un cas de fonctionnement "Sous guidage faible". L'opérateur devra alors décider de l'ordre et du moment d'exécution des tâches.

Dans le cas qui nous intéresse, celui de l'aciérie, le processus technique est contraignant. Le système-pont fonctionne "Sous guidage fort". Les pontonniers n'ont aucune initiative à prendre, ils obéissent aux ordres qui leur sont donnés, du sol, par le contremaître de production. Le système-pont est dépendant entièrement du système de production.

Dans le chapitre II notre objectif est de représenter l'activité du système-pont. Nous avons adopté une représentation graphique sur laquelle on identifie en abscisse les éléments pertinents de la halle d'aciérie où circulent les ponts et en ordonnée le temps. On obtient ainsi une vue d'ensemble des déplacements des ponts au cours du temps. Nous avons ensuite mis au point un programme qui permet de faire réaliser cette représentation graphique par l'ordinateur .

Dans ce chapitre la "maquette" ① est purement reproductrice : elle ne fait que traduire graphiquement la séquence des opérations observées réellement. Au niveau description du processus nous en resterons là, disposant d'un outil qui nous permet de représenter graphiquement tout processus fourni à la Machine. Il faut cependant insister sur l'utilisation de cet outil dans l'aide à la décision.

---

① Nous appellerons "maquette" un programme qui permet la modélisation d'un phénomène physique industriel sans passer par la modélisation mathématique.

Il aurait été vain d'écrire un programme à seule fin de lui faire faire la représentation graphique. Nous avons muni la "maquette" d'un certain nombre d'indicateurs et grâce aux résultats fournis, l'utilisateur pourra apporter des modifications soit au système de production, soit au déroulement du processus.

CHAPITRE II

Comme il a été indiqué, l'objectif de ce chapitre est de représenter le fonctionnement du système-pont. Nous allons représenter graphiquement un processus afin de permettre une visualisation globale des déplacements des ponts et de faciliter la détection des zones où il y a risque d'incidents. Nous verrons successivement dans ce chapitre

I) La méthode de représentation adoptée.

II) Les éléments nécessaires à sa réalisation et sa mise en oeuvre.

I) La méthode de Représentation :

Nous voulons connaître, à chaque instant t_i, l'état ξ_{t_i} du système-pont.

Cet état ξ_{t_i} est fonction du n-uplet suivant. Pour chacun des quatre ponts (c'est à dire pour j allant de 1 à 4), nous mettons en évidence :

① la position P_j du pont j dans l'aciérie

② l'activité A_j du pont j : elle-même fonction de :

2-1 la tâche ta_j que le pont j réalise

2-2 l'état e_j du pont.

On peut ainsi écrire la formule suivante :

$$\xi_{t_i} = \xi_{t_i}(P_1, A_1(ta_i, e_1), P_2, A_2(ta_2, e_2), P_3, A_3(ta_3, e_3), P_4, A_4(ta_4, e_4))$$

⊠ Etude de la position P_j du pont n° j (j=1, 4) dans l'aciérie.

Un pont roulant (voir schéma n° 3) est composé :

a) d'un portique reposant de part et d'autre de la halle de l'aciérie sur un chemin de roulement le long duquel il pourra se déplacer.

b) d'une cabine, fixe, placée sous ce portique, à l'intérieur de laquelle se trouve le pontonnier.

c) d'un chariot mobile placé sur le portique. Ce chariot est muni de moteurs et d'appareils de levage permettant de monter ou de descendre une charge accrochée à l'aide d'un palonnier ou d'élingues.
Le chariot peut se déplacer sur toute la longueur du portique entre les points A et B (voir schéma n° 3) et ainsi balayer toute la largeur de l'aciérie. Le pontonnier peut, par déplacement du chariot, atteindre, avec ses instruments de levage, tout point situé en-dessous du portique.

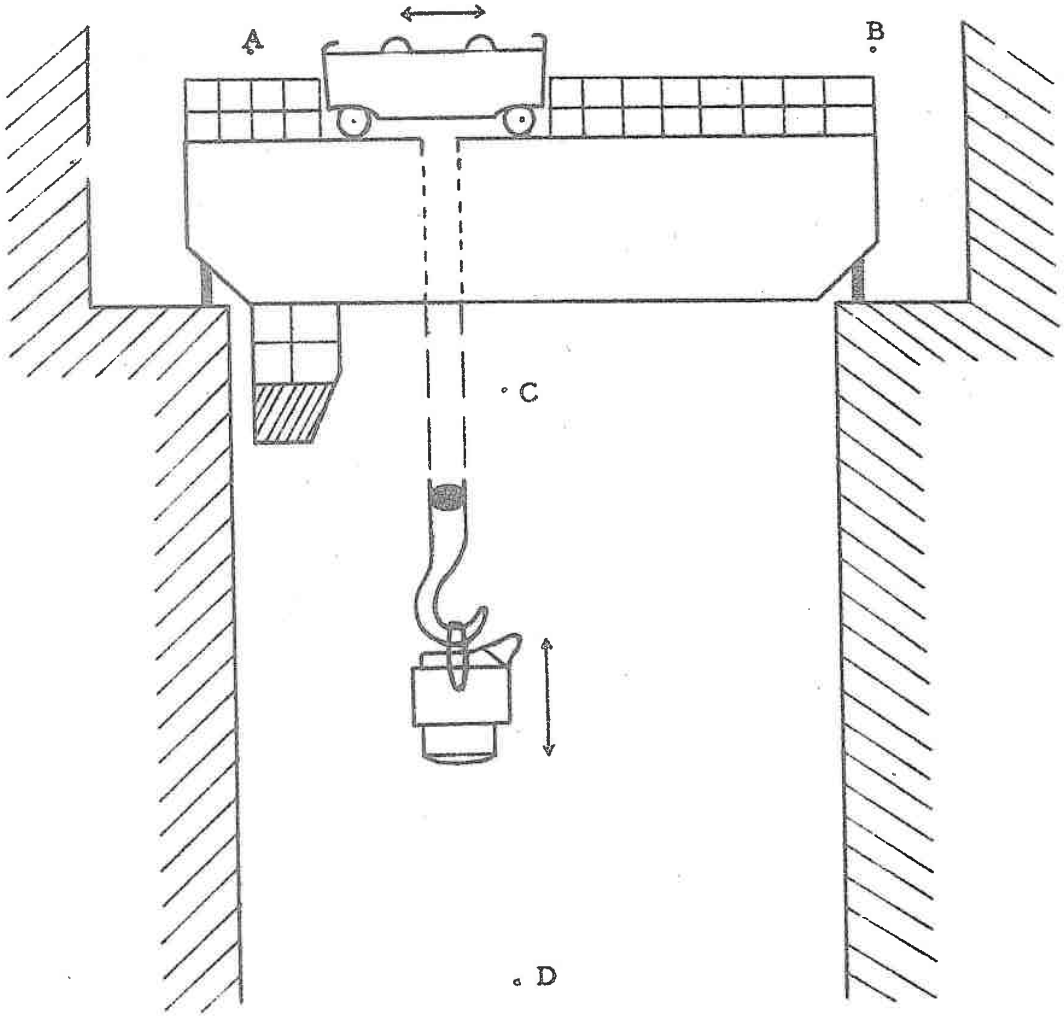


Schéma n° 3 :

COUPE SCHEMATIQUE D'UN PONT ROULANT

Ainsi tout point dans l'aciérie peut être repéré sur un plan par :

1) la position du portique dans la halle

2) la position du chariot sur le portique.

Dans notre étude, lorsque l'on parle de la position P_j du pont j c'est, seule, la position du portique dans la halle qui nous intéresse. Ainsi la seule dimension dont nous ayons besoin, pour positionner les quatre ponts, est la longueur de la halle.

L'abscisse d'un point dans la halle est une variable continue que nous discrétiserons pour des raisons de commodité. Nous prendrons comme unité la largeur d'un pont (soit environ 5 mètres). La halle faisant 150 mètres de long, nous arrivons, dans notre exemple, à une division de l'aciérie en trente "cases" (voir schéma n° 1). Nous pouvons ainsi repérer aisément les ponts grâce aux numéros des cases au-dessus desquelles les portiques se trouvent. Dans un souci d'obtention de résultats plus proches de la réalité nous nous donnons la possibilité de repérer un pont à cheval sur deux cases. Par exemple le pont chargé de placer la poche de coulée dans le skip de la coulée continue se trouvera à cheval sur les cases numérotées 12 et 13

Il est évident que ceci n'est qu'un repérage approximatif de la position des ponts mais il est suffisant pour la méthode employée.

⊠ Etude de l'instant t_i .

Nous désirons connaître l'état du système pont à chaque instant t_i . Pour la raison, déjà évoquée, d'une exploitation sur ordinateur nous sommes obligés de discrétiser le temps.

Nous n'effectuerons une sortie de l'état du système-pont qu'à des instants espacés de Δt. Afin de pouvoir étudier toutes les opérations dont le système-pont est chargé, y compris celle dont la durée est la plus petite, nous prendrons une unité de temps Δt permettant la représentation de la durée de cette tâche «minimale».

Nous ne distinguerons pas la durée des déplacements effectués par les ponts pour aller d'un point à un autre car ces temps sont insignifiants par rapport à ceux des opérations réalisées. Ils sont incorporés aux durées des opérations proprement dites.

Ainsi au lieu de connaître l'état du système-pont à tout moment nous nous contenterons de connaître cet état tous les Δt. Ici, dans notre exemple, nous pourrons nous contenter d'un Δt d'une durée de 15 secondes. Il serait irréaliste de vouloir descendre en-dessous de ce seuil. En effet l'activité de l'aciérie ne peut pas nécessiter une étude extrêmement fixe et précise : il arrive fréquemment qu'un retard de 2 ou 3 minutes survienne dans une opération et que cela ne perturbe pratiquement pas le processus de fabrication. Mais il faut cependant que la valeur de Δt soit quand même suffisamment fine pour permettre l'étude des quelques opérations de courte durée.

En plaçant en abscisse la longueur de l'aciérie et en ordonnée le temps [avec leurs unités respectives la case (de la largeur d'un pont rappelons le) et Δt] nous traçons ainsi le cadre de référence à l'intérieur duquel varie l'état du système-pont.

Pour nous permettre de posséder tous les éléments utiles à la connaissance de l'état du système-ponts à l'instant t_i, il nous reste à expliciter les renseignements relatifs à l'activité A_j de chacun des ponts à cet instant t_i.

⊠ Etude de l'activité A_j du pont j.

L'activité A_j du pont j est fonction de :

① la tâche ta_j que le pont est en train de réaliser

② l'état e_j du pont.

(1) Etude de la tâche ta_j.

Les activités du système-pont sont essentiellement des activités de :

1°) levage avec ou sans accrochage de la charge (LEV)

2°) transfert d'un endroit à un autre, c'est à dire d'une case à une autre (TRA)

3°) dépôt avec ou sans décrochage de la charge (DEP)

Une tâche peut comporter plusieurs fois la séquence LEV-TRA-DEP, chacune de ces séquences pouvant être séparée ou non de la suivante par une activité de transfert TRA.

Etudions le maniement de ces éléments sur des exemples.

Exemple 1 : Tâche "enlever les deux plateaux de la fosse de coulée au four III" : il s'agit pour le pont de prendre le premier plateau qui cache la fosse de coulée (case 16), de l'amener en case 17 et l'y déposer, puis de revenir (en case 16) reprendre le second plateau pour aller le déposer également en case 17.

Structure de cette tâche : LEV-TRA-DEP-TRA-LEV-TRA-DEP

Exemple 2 : Tâche "Coulée au four III" : il s'agit pour le pont de prendre une poche de coulée, d'aller la mettre dans la fosse de coulée, d'y attendre la coulée, et une fois celle-ci effectuée, de remonter la poche pour aller la déposer au skip de la coulée continue.

Structure de cette tâche : LEV-TRA-DEP-LEV-TRA-DEP

Nous avons vu plus haut que la durée des transferts (TRA) est négligeable par rapport aux activités (LEV) de levage et (DEP) de dépôts.

Nous dirons que LEV et DEP sont des opérations élémentaires. Le transfert (TRA) n'apparaîtra que comme ≪séparation≫ entre deux opérations élémentaires.

On appellera "Composante " une séquence d'opérations élémentaires délimitée par des activités de transfert c'est à dire dont le début et la fin sont fixés aux changements de cases.

En reprenant nos deux exemples ci-dessus nous avons :

Exemple 1 : Structure de la tâche :

LEV	TRA	DEP	TRA	LEV	TRA	DEP
C1		C2		C3		C4

Nous avons quatre composantes :

C1 = Prise du premier plateau de la fosse de coulée

C2 = Dépôt de ce plateau

C3 = Prise du second plateau

C4 = Dépôt de ce second plateau.

Exemple 2 : Structure de la tâche :

LEV - TRA - DEP - LEV - TRA - DEP
C1 C2 C3

Nous avons trois composantes :

C1 = Prise de la poche de coulée

C2 = Mise de cette poche dans la fosse et remplissage

C3 = Mise de la poche dans le skip de la coulée continue.

Nantis de ces divers éléments nous pourrons dire qu'une tâche est une succession de composantes, formant une entité, appartenant au processus de production, exécutée par un pont roulant sous la direction d'un ouvrier au sol.

On écrira : $ta_j = \{C_{1,j} ; C_{2,j} ; C_{3,j} ; \ldots ; C_{n,j}\}$

Un pont, pour réaliser une tâche, en effectuera donc les différentes composantes et devra ainsi changer de position à chaque nouvelle composante.

Dans le cas particulier de notre étude le système-pont doit réaliser 62 tâches décomposées en 310 composantes.

(2) Etude de l'état e_j du pont

Les états possibles d'un pont sont au nombre de trois :

a) Soit aucune tâche n'est effectuée au pont à cet instant t_i et on dit alors que le pont est libre et inactif.

b) Soit le pont, à cet instant t_i, est en train de réaliser une tâche.

Deux cas peuvent alors se produire :

b-1) Le pont exécute effectivement une des composantes, on dit alors que le pont est chargé et actif.

b-2) Le pont, alors qu'il effectue une tâche, est obligé d'interrompre momentanément l'exécution de celle-ci pour diverses raisons (par exemple activité prioritaire d'un autre pont, ou attente obligatoire parce qu'il est en avance). Arrêter un pont j alors qu'il se trouve en cours d'exécution de tâche s'appelle provoquer une attente en charge sur le pont j. Le pont est dit, alors, être chargé et inactif.

Nous avons ainsi explicité les renseignements relatifs à l'activité A_j de chaque pont j.

LA REPRESENTATION GRAPHIQUE

Pour représenter graphiquement l'état du système-pont nous allons donc positionner sur la ligne correspondante à l'instant t_i (auquel on étudie le système-pont) chacun des quatre ponts muni des renseignements relatifs à son activité.

Chaque pont j y est matérialisé

- par son numéro j (nous avons conservé la numérotation adoptée par l'usine j variant de 4 à 7).
- par le numéro de la tâche ta_j qu'il effectue (s'il est libre et inactif le numéro de la tâche ta_j sera, par convention, égal à 0).

. par sa position, délimitée par des symboles différents selon son état e_j

a) un trait plein si le pont est chargé et actif.

b) une croix si le pont est chargé et inactif.

c) un trait pointillé si le pont est libre et inactif.

Nous avons établi un exemple de représentation graphique sur le schéma n° 4.

La représentation a, tout d'abord, été effectuée manuellement, mais la longueur du temps nécessaire à sa mise au point et la difficulté d'y apporter des modifications ôtaient à cette représentation graphique manuelle toute valeur d'outil. Ceci nous a conduit à la recherche d'un ensemble de programmes permettant l'obtention de la même représentation graphique, à partir des éléments dont nous disposions, nous-mêmes, pour l'établir.

Ce programme sera appelé "maquette" , par analogie au rôle qu'on pourrait demander à une maquette, c'est à dire un rôle de reproduction simplifiée et expérimentale de l'activité de l'ensemble du système-pont.

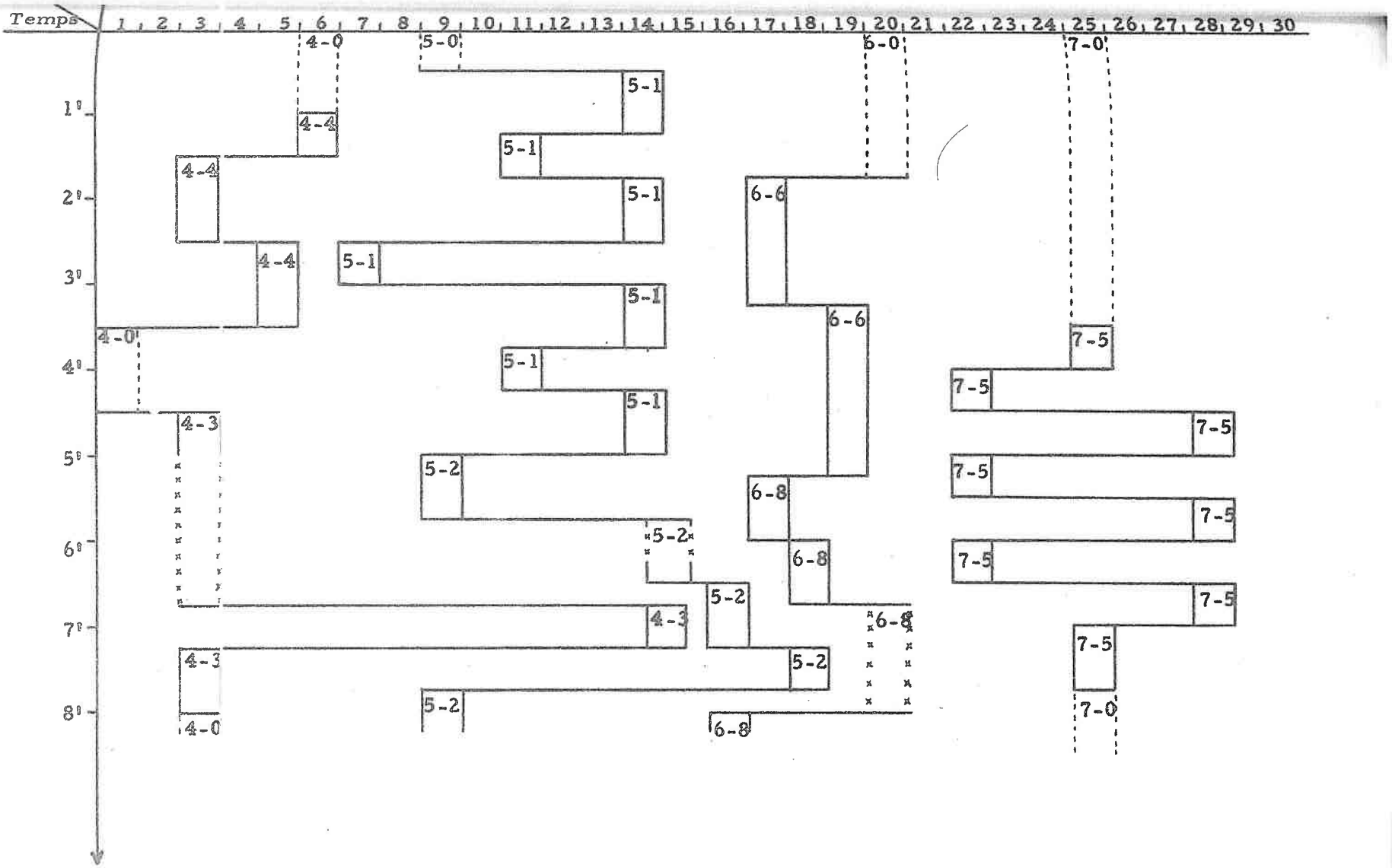


Schéma n° 4 : REPRESENTATION GRAPHIQUE

II) Les éléments nécessaires à sa réalisation et sa mise en oeuvre

"Maquette"* et Fichiers

ANALYSE FONCTIONNELLE

Le système et les différents états qu'il peut prendre dans le temps sont définis à partir de divers éléments

- les uns, représentatifs de l'activité du système de production, sont constants, fixés une fois pour toute.
- les autres, dépendants des processus que l'on traite, sont variables.

Ces éléments sont rangés dans des fichiers et tables que nous présenterons maintenant.

(I) Fichier et table permanents concernant l'activité de production.

① Table "TTACH"

Cette table contient le nombre de composantes de chacune des tâches et, généré par programme, un moyen d'accès aux renseignements de la première composante de chacune des tâches.

② Fichier "TCOMP"

Ce fichier contient les renseignements caractéristiques de chaque composante : des codes indiquant les ponts susceptibles de la réaliser, l'endroit où se passe cette composante, un code indiquant éventuellement l'existence d'un choix entre divers composantes, d'accessoire de levage à utiliser pour cette composante, et la durée moyenne de cette composante.

---

* Rappelons que nous appelons "maquette" le programme qui permet la modélisation de l'activité du système-pont sans passer par la modélisation mathématique.

(II) Fichiers paramètres concernant le processus à traiter.

① Fichier "TPROC"

Ce fichier contient, pour chaque tâche réalisée par le système-pont, son numéro, l'instant de déclenchement et le numéro du pont auquel elle a été affectée.

② Fichier "DV"

Ce fichier contient, pour chaque déplacement d'un pont libre et inactif, l'instant de réalisation de ce déplacement, le numéro du pont qui l'effectue, et l'endroit où il se rend.

③ Fichier "AT"

Il contient, pour chaque attente en charge, l'instant auquel débute cette attente en charge, le numéro du pont concerné et la durée de cette attente.

④ Fichier "TPONT"

Il contient d'abord les renseignements relatifs à l'état initial du système-pont. Puis, remis à jour tous les Δt, il contient les renseignements concernant les divers états du système-pont. C'est à dire pour chaque pont son état, sa position, la tâche et dans la tâche la composante qu'il effectue et l'heure de fin de la composante.

(III) La "Maquette"

La maquette est le programme que nous utilisons pour déterminer les informations suivantes :

Ⓐ A l'instant t_i $(t_i = t_{i-1} + \Delta t)$

1. L'instant t_i auquel on étudie l'état du système-pont.

2. Pour chaque pont :

   a) sa position

   b) son numéro

   c) son état

   d) la tâche qu'il est en train d'effectuer.

3. Pour chaque couple de ponts voisins

   a) l'espacement entre ces deux ponts.

Ⓑ A la fin du processus traité :

Pour chaque pont :

a) la longueur du chemin parcouru par le pont composée de

   a-1 = la longueur des déplacements quand il est libre et inactif.

   a-2 = la longueur des déplacements quand il est chargé.

b) la durée totale du travail qu'il a effectué.

Compte tenu de la complexité du problème, nous avons été amené à présenter celui-ci à l'aide d'une description modulaire. Cette présentation, par niveaux successifs de complexité croissante, en partant des lignes générales du problème, permet de pénétrer progressivement dans les détails du programme. Nous trouverons en Annexe II-1 tous renseignements permettant de comprendre la description modulaire de cette méthode.

Nous ne donnons dans cette partie Analyse fonctionnelle que les grandes lignes du programme, c'est à dire les niveaux 1 et 2. Nous trouverons la suite de la description modulaire (niveaux supérieurs ou égaux à 3) dans la partie Analyse organique.

DESCRIPTION MODULAIRE

NIVEAU I

I - TRAITEMENT

1 - Lecture des fichiers
2 - Sortie de la ligne de référence
3 - Pour chaque Δt
    4 - Etude des attentes chargées
    5 - Etude du processus
    6 - Etude de modification
    7 - Etude de déplacement à vide
    8 - Sortie de ligne

Déclaration : fichiers TPONT, TPROC, TTACH, TCOMP, AT, DEV

Respectivement : fichier occupation des ponts, fichier du processus, fichier des tâches, fichier des composantes, fichier des attentes en charge, fichier des déplacements à vide.

N.B. : Cette description nécessite en réalité une page par module, mais pour faciliter l'impression et la mise en page, la présence d'un trait continu simule le changement de page.

NIVEAU II

I.1 - Lecture des fichiers : (dans cette 1ère étape, on introduit tous les fichiers en mémoire centrale dès le début)

1 - Lecture du fichier de processus TPROC
2 - Lecture du fichier des tâches TTACH
3 - Lecture du fichier des composantes TCOMP
4 - Lecture du fichier occupation des ponts TPONT
5 - Lecture du fichier des attentes en charge AT
6 - Lecture du fichier des déplacements à vide DEV

Déclaration : tous les 6 fichiers.

NIVEAU II

I.2 - Sortie de la ligne de référence LREF

1 - Numérotation de chacune des cases de l'aciérie (il y en a 30) LREF
2 - Impression de LREF

NIVEAU II

I.4 - Etude des attentes en charge

1 - Lecture du fichier des attentes en charge
2 - Pour chaque attente débutant à cet instant
| 3 - Relevé du pont concerné
| 4 - Modification de la caractéristique
| 5 - Modification du compteur de temps

Déclaration : fichiers AT, TPONT

NIVEAU II

I.5 - Etude du processus

1 - Lecture du fichier processus
2 - Pour chaque tâche débutant à cet instant
| 3 - Relevé du numéro de la tâche
| 4 - Recherche des composantes

Déclaration : fichiers TPROC, TCOMP, TPONT, TTACH

NIVEAU II

I.6 - Etude de modification

1 - Lecture du fichier occupation des ponts
2 - Pour toute composante s'achevant à cet instant
| 3 - Relevé du pont concerné
| 4 - Traitement de la modification

Déclaration : fichiers TPONT, TCOMP, TTACH

N.B. : Modification peut être la fin d'une tâche ou la fin d'une composante, tout ce qui peut modifier la position d'un pont ou sa caractéristique et partant modifier l'état du "système-pont".

NIVEAU II

I.7 - Etude de déplacement à vide

1 - Lecture du fichier des déplacements à vide
2 - Pour tout déplacement débutant à cet instant
| 3 - Relevé du pont concerné
| 4 - Modification du numéro de case
| 5 - Modification du compteur de temps

Déclaration : fichiers DEV, TPONT

NIVEAU II

I.8 - Sortie de ligne

1 - Pour chaque pont
  2 - Lecture dans le fichier TPONT de la ligne correspondante
  3 - Préparation du quadruplet caractéristique
  4 - Mise de ce quadruplet dans la ligne de sortie
5 - Impression de la ligne de sortie
6 - Remise à blanc de cette ligne

Déclaration : fichier de sortie TPONT

La maquette permet d'avoir une vision dynamique d'un processus observé. Elle fournit une vue globale de l'activité du système-pont et des problèmes de sécurité qui peuvent se poser.

Elle donne les éléments de chacun de ces problèmes de sécurité mais, en aucun cas, ne fournit de solution au problème soulevé. Cette maquette doit être considérée comme une aide à la décision. Elle permet à l'utilisateur de détecter des zones dangereuses, par exemple :

a) des zones de rapprochement des ponts : c'est dans ces zones que la survenue d'incidents du type collision est la plus probable.

b) des zones dans lesquelles un pont se déplace rapidement, balayant un large secteur de l'aciérie. Le fait de transporter, sur une longue distance, des charges dangereuses augmente les risques d'accident.

Toute diminution du nombre ou de l'importance de telles zones est une aide apportée à la sécurité. Ces diminutions peuvent être réalisées par modification d'une ou des deux variables suivantes :

1) La position géographique de certains éléments du système de production

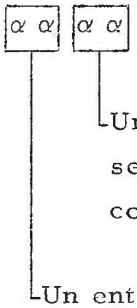
2) L'instant de déclenchement d'une tâche.

ANALYSE ORGANIQUE

(I) Fichier et Table permanents concernant l'activité de production.

I. 1) Table "TTACH"

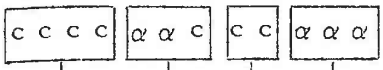
C'est une table ordonnée dans laquelle on trouvera pour chaque tâche :



I. 2) Fichier "TCOMP"

Ce fichier contient tous les renseignements caractéristiques de l'activité du système de production.

Il contient pour chaque composante :



└ Un entier pour la durée moyenne de la composante, formé d'un entier sur 2 chiffres pour le nombre de minutes et d'un entier sur 1 chiffre pour le nombre de fractions de minute. .

└ Deux codes sur un caractère chacun, le premier indiquant l'existence d'un choix à faire, le second indiquant le numéro de code de l'accessoire utilisé. Ils sont inutiles actuellement.

└ Elément composé d'un entier de 2 chiffres indiquant le numéro de la case où se déroule la composante et d'un code sur un caractère indiquant la position du pont par rapport à la case. ①

└ 4 codes (un par pont). La valeur du code est 1 si la composante peut être effectuée par le pont correspondant,

0 sinon.

(II) Fichiers paramètres concernant le processus traité.

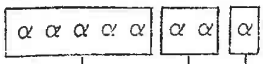
II. 1) Fichier "TPROC"

Ce fichier contient tous les renseignements relatifs à chaque étape du processus de fabrication.

Il contient pour chaque étape :

---

① Voir page II. 28



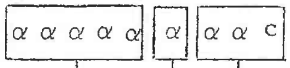
Un entier indiquant le numéro du pont auquel la tâche a été affectée.

Un entier sur 2 caractères indiquant le numéro de la tâche.

L'heure de déclenchement de la tâche composée de trois entiers. L'heure sur 2 caractères, le nombre de minutes sur 2 caractères et le nombre de fractions de minute sur un caractère.

II. 2) Fichier "DV"

Pour chaque déplacement d'un pont libre et inactif, ce fichier contient :



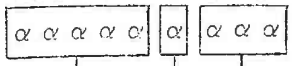
Elément codé (voir dans "TCOMP" l'explicitation) indiquant la case dans laquelle se rend le pont.

Un entier d'un caractère spécifiant le numéro du pont concerné.

L'instant du déplacement composé de 5 entiers. L'heure sur 2 caractères, le nombre de minutes sur 2 caractères et le nombre de fractions de minute sur un caractère.

II.3) Fichier "AT"

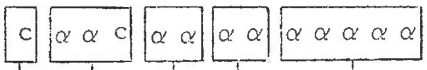
Pour chaque attente en charge relevée, le fichier contient :



- α α α : La durée de l'attente, le nombre de minutes sur 2 caractères et le nombre de fractions de minute sur 1 caractère.
- α : Un entier d'un caractère pour le numéro du pont concerné.
- α α α α α : L'instant de déclenchement de l'attente composé de trois entiers, l'heure sur 2 caractères, le nombre de minutes sur 2 caractères et le nombre de fractions de minute sur 1 caractère.

II.4) Fichier "TPONT"

Le fichier contient les renseignements relatifs à l'état du système-pont. Pour chaque pont il contient :



- α α α α α : L'heure de fin de réalisation de la composante traitée, composée de l'heure sur 2 caractères, des minutes sur 2 caractères et d'1 caractère pour le nombre de fractions de minute .
- α α : Entier de deux caractères indiquant le numéro de la composante en cours de réalisation.
- α α : Entier de deux caractères pour le numéro de la tâche en cours de réalisation.
- α α c : Position géographique définie par le numéro de case sur 2 caractères et par 1 code de position (voir codage déjà explicité).
- c : Code d'un caractère pour caractériser l'état du pont.

(III) "La maquette"

Suite de la description modulaire (niveaux supérieurs ou égaux à 3).

NIVEAU III

I.5.4 - Recherche des composantes

1 - Lecture du fichier tâche
2 - Calcul de la borne inférieure IDEB
3 - Calcul de la borne supérieure IFIN
4 - Lancement de la composante IDEB
5 - Test de déclenchement précoce

Déclaration : fichiers TTACH, TCOMP, TPCNT

N.B. : Les points 2 et 3 ne sont pas des calculs mais plutôt des relevés, dans le fichier des tâches (TTACH), des deux bornes entre lesquelles se trouvent les composantes de la tâche que l'on étudie. Le point 5 peut être mis avant le point 4.

NIVEAU III

I.6.4 - Traitement de la modification

1 - S'il s'agit d'une fin de tâche
| 2 - Mise à jour "Fin de tâche"
3 - Sinon
| 4 - Traitement de la composante intermédiaire.

Déclaration : fichiers TPCNT, TCOMP

NIVEAU III

I.8.3 - Préparation du quadruplet caractéristique de chaque pont

1 - Recherche de la case occupée sur la ligne de référence soit CASEREF (i)

2 - Recherche du nombre de cases occupées soit TESTCASE (i)

3 - Recherche des quatre caractères composant le quadruplet soit QUAD (i, j) (j=1, 4)

4 - Test de vérification - ordre des ponts

Déclaration : fichier TPONT

NOTICE

I.8.3 -

- On dispose d'un nombre de trois chiffres qui forme le code d'une case d'aciérie occupée par un pont.

  Il s'agit de positionner le quadruplet relatif à ce pont sur la ligne de référence.

- Les nombres sont formés (comme vu plus haut) de deux chiffres relatifs au numéro de case (A) + 1 chiffre relatif au code d'occupation de cases.

  La valeur prise par le code est :

  0 si le pont occupe exactement la case A

  1 si le pont est à cheval sur la case A et la case A+1

  2 si le pont occupe la case A et la case A+1. En réalité le pont n'occupe toujours que la largeur d'une case mais lors d'allers et retours répétés sur un même intervalle très faible (sur 2 cases comme ici) on préfère considérer comme occupées les 2 cases.

1er type : exemple 160 : le pont est sur la case 16.

TESTCASE(i) = 1, CASEREF(i) = 16x4 = 64,
i - e le quadruplet relatif à ce pont occupera les caractères 64, 65, 66, 67 de la ligne d'imprimante.

2ème type : exemple 161 : le pont à cheval sur 16 et 17.

TESTCASE(i) = 1, CASEREF(i) = 16x4+2 = 66

3ème type : exemple 162 : le pont "occupe" les deux cases 16 et 17.

CASEREF(i) = 64, TESTCASE(i) = 2 ce qui aura pour effet de bisser le quadruplet à partir de 68 dans cet exemple.

NIVEAU IV

I.5.4.4 - Lancement de la composante IDEB

1 - Lecture dans le fichier des composantes

2 - Recherche du pont qui effectue la tâche

3 - Modification du numéro de tâche

4 - Modification du numéro de la composante

5 - Modification du compteur de temps

6 - Modification de la caractéristique

7 - Modification du numéro de case

Déclaration : fichiers TCOMP, TPONT

NIVEAU IV

I.5.4.5 - Test de déclenchement précoce

1 - Pour chaque pont
2 - Vérifier s'il a fini la tâche qu'il était en train d'effectuer si non impression d'un libellé.

Déclaration : fichier TPONT

NIVEAU IV

I.6.4.2 - Mise à jour "fin de tâche"

1 - Mise à 1 du premier caractère de la ligne occupée par le pont i concerné dans TPONT
(c'est à dire que le pont i redevient libre)

2 - Mise à 0 des caractères (5-6) et (7-8) dans cette même ligne (c'est à dire des positions relatives au numéro de tâches et au numéro de composantes).

Déclaration : fichier TPONT

NIVEAU IV

I.6.4.3 - Traitement de la composante intermédiaire

1 - Modifier le numéro de composante
2 - Modifier le numéro de case
3 - Modifier le compteur de temps
4 - Modifier la caractéristique du pont

Déclaration : fichiers TPONT, TCOMP

NIVEAU IV

I.8.3.1 - Recherche de la case de référence

1 - Calcul du numéro de la case
2 - Si le code est 1
    | 3 - Modification de la position

Déclaration : fichier TPONT

NIVEAU IV

I.8.3.2 - Recherche du nombre de cases

1 - Si le code d'occupation de cases est 0 ou 1
| 2 - Prévoir l'occupation d'une seule case
3 - Si ce code est 2
| 4 - Prévoir l'occupation de deux cases

Déclaration : fichier TPONT

NIVEAU IV

I.8.3.3 - Recherche des 4 caractères composant le quadruplet

1 - QUAD(i, 1) = i — Numéro du pont
2 - QUAD(i, 2) = TPONT(PONT, 1) — Caractéristique du pont
3 - QUAD(i, 3) = TPONT(PONT, 5) } Numéro de la tâche
4 - QUAD(i, 4) = TPONT(PONT, 6) }

Déclaration : fichier TPONT

NIVEAU IV

I.8.3.4 - Test vérification ordre des ponts

1 - Comparaison des numéros de case occupées respectivement par le pont i et son voisin i+1
2 - Si l'ordre n'est pas respecté : impression d'un libellé.

Déclaration : fichier TPONT

CHAPITRE III

A partir des éléments permanents de l'activité (fichiers " "tâche" et "composante") et des éléments spécifiques d'un processus réalisé (fichiers "Processus", Attente en charge" et Déplacement à vide") on obtient, par le programme (REPRO) que nous venons d'expliciter au chapitre II, la représentation graphique de ce processus observé. La "Maquette" ne fait que traduire, conformément à la représentation graphique adoptée, les éléments qui lui étaient fournis en entrée. Elle ne prend aucune initiative.

Dans la suite, nous désirons obtenir, par programme, des éléments qui auparavant étaient considérés comme des données. Nous pouvons le faire en diminuant les entrées et en lui fournissant, à la place, des règles de traitement. En utilisant ces règles, la "Maquette" devra retrouver les éléments qui lui sont nécessaires pour réaliser l'objectif choisi.

Ainsi, dans une première phase (chapitre III) on supprimera les deux fichiers paramètres annexes Déplacement à vide et Attente en charge). Nous fournissons toujours en données les deux fichiers permanents et le fichier paramètre principal (celui du processus). Munie d'un certain nombre de règles de traitement que nous allons expliciter, la "Maquette" créera les fichiers paramètres annexes dont elle aura besoin. L'objectif du programme sera de simuler, de tester, d'aménager au besoin le processus proposé. Ce programme (SIMUL) peut avoir une exploitation intéressante dans le cadre d'une formation du personnel impliqué dans l'activité du système-pont.

Puis dans une deuxième phase (chapitres IV et V) nous diminuerons encore les données fournies au programme en supprimant le dernier fichier paramètre (celui du processus). Pour faire générer ce fichier par la "Maquette" à partir desseuls fichiers permanents, nous proposerons une méthode d'affectation et une méthode d'ordonnancement.
L'importance donnée au facteur Sécurité dans ces deux méthodes de Recherche Opérationnelle y est essentielle.
Soit ROSEC le programme correspondant.

Regroupons dans un tableau comparatif les divers rôles des fichiers (en entrée ou en sortie) selon les programmes REPRO, SIMUL et ROSEC qui les mettent en jeu.

		Données			Résultats		
		REPRO	SIMUL	ROSEC	REPRO	SIMUL	ROSEC
Fichiers permanents	TTACH	x	x	x			
	TCOMP	x	x	x			
Fichiers paramètres	TPROC	x	x				x
	AT	x				x	x
	DV	x				x	x
Etat du système	TPONT				x	x	x

Tableau III. 1

Nous voyons, sur ce tableau, que les données fournies à la "Maquette" sont de moins en moins nombreuses. En revanche, grâce aux règles de traitement que nous introduisons dans le programme, la "Maquette" peut générer de plus en plus d'éléments intervenant dans le cadre d'une aide à la décision proposée à l'utilisateur.

Dans le chapitre III, nous allons expliciter le programme SIMUL qui simule, teste et aménage (par création des fichiers paramètres "Attente en charge" et "Déplacement à vide") un processus (fichier Processus) établi par un opérateur extérieur.

Nous verrons successivement :

A) L'exposé de la méthode et son exploitation

B) L'analyse fonctionnelle.

A - L'EXPOSE DE LA METHODE

Dans sa dernière forme (Tab. III. 1, 1ère ligne) la "Maquette" avait pour seul rôle de fournir une représentation du processus observé à partir

1) des tâches que le système-pont effectuait
2) des déplacements que les ponts libres et inoccupés ont effectués
3) des attentes en charge auxquelles ils ont été soumis

Tous ces renseignements étaient notés, codés et fournis en données à la "maquette".

Les résultats obtenus par la méthode de représentation graphique mettent en évidence la présence de déplacements à vide et d'attentes en charge inutiles. Ces attentes en charge superflues sont souvent dûes à des "anticipations" prématurées des pontonniers. Par exemple, le pontonnier "anticipe" en prenant son panier de ferraille trop longtemps avant l'ouverture du four à l'intérieur duquel il doit décharger ce panier. Cette attente, d'une dizaine de minutes très souvent, est en «contravention» avec les consignes de sécurité qui interdisent le levage de charges sans réalisation effective de travail.

Le principe de la méthode vise à supprimer ces éléments superflus et à ne générer que ceux indispensables. Il consiste à entrer comme données à la "Maquette"

. d'une part les deux fichiers permanents contenant tous les renseignements relatifs aux tâches et aux composants (fichiers TTACH et TCOMP)

. D'autre part le seul fichier paramètre du processus (fichier TPROC)

Par programme (la "Maquette"), ainsi, à partir de ces éléments doit fournir, selon les cas, les éléments suivants :

1er cas : Le processus qu'elle simule est réalisable compte-tenu de la seule contrainte que nous lui imposons : l'ordre dans lequel se trouvent les ponts est immuable. La "Maquette" sort alors comme résultats ceux que nous avions vus dans le chapitre précédent, c'est à dire

α) les divers états successifs du système-pont

β) les divers indicateurs dont nous avions muni la "Maquette"

2ème cas : Le processus qu'elle simule n'est pas réalisable (parce que, à un moment donné, deux ponts sont amenés à occuper le même secteur géographique de la halle au même moment). La "Maquette", alors, réalise le travail suivant :

α) elle signale le conflit géographique et la position respective des ponts concernés dans la halle.

β) elle essaie de fournir une solution acceptable et c'est là qu'apparaît l'aspect générateur. Cette recherche se décompose en deux actions ; une action d'analyse du problème et du choix du pont à déplacer, et une action de création d'une situation acceptable.

(A1) Analyse du problème

Nous ne traitons que les cas où seuls deux ponts sont impliqués. Pour analyser le problème posé, nous utilisons trois critères d'analyse que nous allons expliciter.

Critère 1 : La priorité des ponts impliqués

On va attribuer à chacun des deux ponts impliqués dans la situation conflictuelle un ordre de priorité. Celui dont la priorité est la plus faible verra son activité interrompue pour permettre la réalisation de l'activité de l'autre pont.

L'attribution de ces priorités se fait en deux temps :

a) Attribution d'une priorité aux ponts selon leur état e_j.
Par ordre de priorité décroissante :

- Si le pont est occupé et actif (O. A) = Priorité 3
- Si le pont est occupé et inactif (O. I) = Priorité 2
- Si le pont est libre et inactif (L. I) = Priorité 1

Il est normal d'attribuer au pont libre et inactif la plus faible priorité car c'est celui dont on peut modifier le plus facilement la position.

De même nous avons attribué la plus forte priorité au pont occupé et actif, la modification de son activité altérant le plus le déroulement du processus.

Nous avons regroupé dans le tableau n°III.2 les différents cas qui peuvent se présenter.

Situations	1	2	3	4	5	6	7	8	9
Pont n° i	OA (3)	OA (3)	OA (3)	OI (2)	OI (2)	OI (2)	LI (1)	LI (1)	LI (1)
Pont n° i+1	OA (3)	OI (2)	LI (1)	OA (3)	OI (2)	LI (1)	OA (3)	OI (2)	LI (1)
comparaison des priorités P(i) et P(i+1)	P(i)= P(i+1)	P(i)> P(i+1)	P(i)> P(i+1)	P(i+1)> P(i)	P(i)= P(i+1)	P(i)> P(i+1)	P(i+1)> P(i)	P(i+1)> P(i)	P(i)= P(i)

Tableau III.2

Commentaires :

1) Les quatre cases hachurées (situations 5, 6, 8, 9) sont des situations dites ≪spéciales≫ c'est à dire qu'elles ne peuvent pas se rencontrer lors d'un déroulement normal d'un processus. Elles ne peuvent survenir qu'après une génération par la "Maquette" soit d'une attente en charge, soit d'un déplacement à vide.

2) Dans la situation n° 9 les deux ponts sont libres et inactifs. Imposons pour des raisons de commodité le choix du pont i comme pont à déplacer.

3) Six situations fournissent des priorités différentes pour les deux ponts et permettent ainsi le choix du pont à déplacer. Ce sont les situations 2, 3, 4, 6, 7, 8.

4) Les situations 1 et 5 où les deux ponts sont occupés et possèdent la même priorité ne peuvent être réglées à l'aide de ce seul critère. Nous allons introduire une attribution de priorité relative à la tâche qui les occupe.

b) Attribution d'une priorité aux ponts selon la tâche qu'ils réalisent.

Là encore, nous distinguerons par ordre de priorité décroissante :

- Les tâches prioritaires, celles dont la réalisation doit avoir lieu à une date fixée (dans la réalité, avec une légère marge). Priorité 3.
- Les tâches semi prioritaires, celles dont la réalisation doit être faite entre telle heure et telle heure. Ainsi munies d'une date de réalisation au plus tôt et une date de réalisation au plus tard, elles deviennent prioritaires à partir de cette dernière date. Priorité 2.
- Les tâches libres, celles dont la réalisation peut avoir lieu à tout moment. Priorité 1.

La comparaison entre les priorités ainsi définies favorise le choix du pont à déplacer. Cependant si les deux ponts effectuent des tâches de priorités égales, la "Maquette" devra recourir à un autre critère d'analyse.

N. B. : Dans la réalisation proprement dite du programme nous avons considéré toutes les tâches comme libres, n'ayant pu obtenir tous les renseignements qui nous auraient permis de classer les tâches. L'attribution des priorités selon les tâches n'est pas faite, mais elle doit pouvoir être facilement ajoutée.

La "Maquette" dispose alors de deux critères d'analyse, un critère de temps et un critère géographique. La "Maquette" n'utilise pas simultanément ces deux critères, mais elle y a accès et sur option l'utilisateur pourra choisir l'un et/ou l'autre selon les buts qu'il désire atteindre, par exemple le niveau de finesse de la solution qu'il veut voir générer.

Critère 2 : C'est un critère simple, basé sur la durée des composantes restant à effectuer. La "Maquette" accorde la priorité au pont qui aura fini le premier sa tâche, car il deviendra plus tôt libre et donc plus facilement manipulable. Cependant la "Maquette" n'estimera les résultats satisfaisants que si la différence entre les durées résiduelles est significative ; par exemple, supérieure à cinq minutes.

Un problème se pose, alors, pour choisir cette marge minimum $\mathcal{E}$ au delà de laquelle la "Maquette" accepte de choisir le pont à déplacer. Il est évident qu'une marge réduite à Δt l'unité de temps (dans notre cas $\Delta t = 15$ secondes) permettra de traiter un pourcentage très élevé des conflits, mais cela implique une résolution «approchée» de ces conflits. Elle peut cependant s'avérer suffisante.

On peut formaliser cette règle comme suit :

- Soit i et i+1 les deux ponts concernés. Supposons que la situation conflictuelle intervienne quand le pont i traite la $m^{ième}$ composante de la tâche ta (qui en possède n) et quand le pont i+1 traite la $p^{ième}$ composante de la tâche tb (qui en possède q).

- Soit d(t, s) la durée de la composante s de la tâche t.

Nous aurons alors :

$$A = \sum_{k=m}^{n} d(ta, k)$$

$$B = \sum_{k=p}^{q} d(tb, k)$$

Avec $\mathcal{E}$ la marge minimum choisie, le critère sera significatif si $|A-B| \geq \mathcal{E}$.

Dans le cas favorable, deux possibilités se présentent :

1°) $A+\mathcal{E}<B$ c'est alors le pont i qui dispose de la priorité.

2°) $B+\mathcal{E}<A$ c'est alors le pont i+1 qui possède la priorité.

Critère 3 : C'est un critère basé sur la position future de chacun des ponts concernés. Deux éléments importants nous intéressent dans ces positions futures.

α) la position relative des ponts entre eux

β) la position absolue de chacun des ponts dans la halle.

Nous nous plaçons dans le cas d'une situation conflictuelle impliquant les ponts i et i+1, ayant lieu à l'instant t_K.

α) Position relative (PR) des ponts entre eux

(α1) A l'instant t_{K-1} ($t_K = t_{K-1} + \Delta t$) les positions respectives (i-e le numéro de case occupée) de chaque pont sont :

. pour le pont i : $p_1(i, t_{K-1})$

. pour le pont i+1 : $p_1(i+1, t_{K-1})$

A cet instant t_{K-1}, la situation conflictuelle ne s'est pas encore déclarée, nous avons donc :

$$p_1(i, t_{K-1}) < p_1(i+1, t_{K-1})$$

Quand nous avons cette inégalité, nous pouvons dire que la position relative des deux ponts est normale (NOR)

(α2) A l'instant t_K auquel se déclenche le conflit nous avons, puisque l'ordre des ponts est interverti.

$$p_1(i, t_K) \geq p_1(i+1, t_K)$$

Quand nous avons cette inégalité, nous pouvons dire que la position relative des deux ponts est ANORMALE (ANCR).

β) Positions absolues des ponts dans la halle

(β1) Positions respectives à l'instant t_K.

Supposons que les ponts i et i+1 réalisent respectivement la composante n de la tâche ta et la composante de la tâche tb, leurs positions respectives dans la halle peuvent alors se formaliser ainsi :

$$p_2(i, ta, n) \quad \text{et} \quad p_2(i+1, tb, m)$$

On peut rapprocher ces fonctions des fonctions précédentes.

$$p_1(i, t_K) = p_2(i, ta, n)$$

et

$$p_1(i+1, t_K) = p_2(i+1, tb, m)$$

De plus, comme nous sommes en situation de conflit

$$p_2(i, ta, n) \geq p_2(i+1, tb, m) \qquad \text{(PR est ANOR)}$$

(β2) Positions respectives à l'instant $t_{K'}$.

Nous nous intéressons alors au premier changement dans les positions qui se produirait si on laissait le processus se dérouler sans intervenir. Cette modification interviendrait lors d'un passage d'une composante à la suivante pour, au moins, l'un des deux ponts. Rappelons que le changement de composante impose un changement de case, de position.
Soit $t_{K'}$ l'instant auquel se passe cette modification de la position. On a $t_{K'} = t_K + r\,\Delta t \quad (r \geq 1)$

Lors de cette modification, trois cas peuvent se produire :

1°) Le changement de composante se fait sur le pont i+1

$$p_1(i, t_{K'}) = p_1(i, t_K) = p_2(i, ta, n)$$

$$p_1(i+1, t_{K'}) \neq p_1(i+1, t_K) \quad \text{et} \quad p_1(i+1, t_{K'}) = p_2(i+1, tb, m+1)$$

2°) Le changement de composante se fait sur le pont i

$$p_1(i, t_{K'}) \neq p_1(i, t_K) \quad \text{et} \quad p_1(i, t_{K'}) = p_2(i, ta, n+1)$$

$$p_1(i+1, t_{K'}) = p_1(i+1, t_K) = p_2(i+1, tb, m)$$

3°) Le changement de composante intervient sur les deux ponts

$$p_1(i, t_{K'}) \neq p_1(i, t_K) \quad \text{et} \quad p_1(i, t_{K'}) = p_2(i, ta, n+1)$$

$$p_1(i+1, t_{K'}) \neq p_1(i+1, t_{K'}) \quad \text{et} \quad p_1(i+1, t_{K'}) = p_2(i+1, tb, m+1)$$

Nous allons introduire les notions de Prolongation et d'Inversion afin d'expliciter les inégalités entre les diverses positions par les ponts entrainées par les changements de composantes.

. Nous dirons que l'on a Prolongation (PRO) de l'état des ponts si :

α) pour i : le pont va au moins occuper une position plus à droite dans la halle

$$p_1(i, t_{K'}) \geq p_1(i, t_K)$$

On dit alors que la position du pont i est en état de prolongation.

β) pour i+1 : le pont va, au moins, occuper une position plus à gauche dans la halle

$$p_1(i+1, t_{K'}) \leq p_1(i+1, t_K)$$

On dit alors que la position du pont i+1 est en état de prolongation.

. Nous dirons que l'on a Inversion (INV) si :

α) pour i : le pont va occuper une position plus à gauche dans la halle

$$p_1(i, t_{K'}) < p_1(i, t_K)$$

La position du pont i est alors en état d'inversion.

β) pour i+1 : le pont va occuper une position plus à droite dans la halle

$$p_1(i+1, t_{K'}) > p_1(i+1, t_K)$$

La position du pont i+1 est alors en état d'inversion.

Nous avons ainsi étudié :

a) la position relative PR des ponts avec les deux états possibles NORmal et ANORmal.

b) la position absolue de chacun des ponts i et i+1 respectivement POS(i) et POS(i+1) et l'état de ces deux positions INVersion ou PROlongation.

Formalisons la situation afin d'y appliquer le critère de choix.

L'étude du conflit nécessite la connaissance de l'état du couple de pont (ECP)

1) Avant le déclenchement de la situation conflictuelle. Soit E_1 cet état.

$$E_1 = ECP(t_{K-1}, POS(i), POS(i+1), PR)$$

2) Au moment du déclenchement de la situation conflictuelle. Soit E_2 cet état.

$$E_2 = ECP(t_K, POS(i), POS(i+1), PR)$$

3) Au moment de la première modification de cette situation. Soit E_3 cet état.

$$E_3 = ECP(t_{K'}, POS(i), POS(i+1), PR)$$

avec $t_K = t_{K-1} + \Delta t$

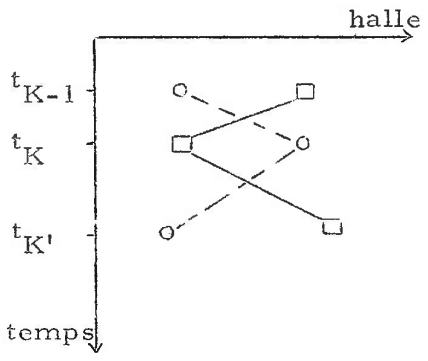
$t_{K'} = t_K + r\,\Delta t$ $\quad (r \geq 1)$

Nous avons regroupé dans le tableau III.3 ci-dessous les sept valeurs possibles de E_3. Il existe un cas impossible : c'est celui tel que, partant d'une position relative ANORmale et les positions absolues de chacun des ponts se PROlongeant, on obtienne une position relative NORmale.

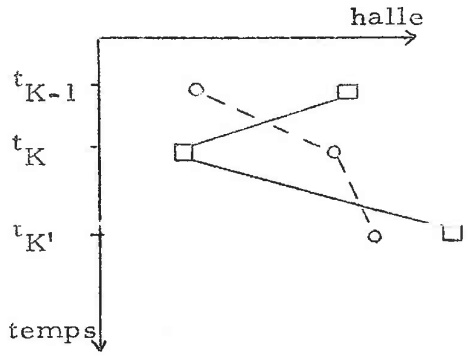
Etat du pont i	INV	PRO	INV	PRO	PRO	INV	INV	PRO
Etat du pont i+1	INV	INV	PRO	PRO	INV	INV	PRO	PRO
Position relative	NOR	NOR	NOR	ANOR	ANOR	ANOR	ANOR	NOR
Cas n°	1	2	3	4	5	6	7	imposs.

Tableau n° III.3

Représentons graphiquement les sept cas en plaçant en abscisse la halle, en ordonnée le temps et en indiquant les trois valeurs E1, E2, E3 aux instants t_{K-1}, t_K, $t_{K'}$.



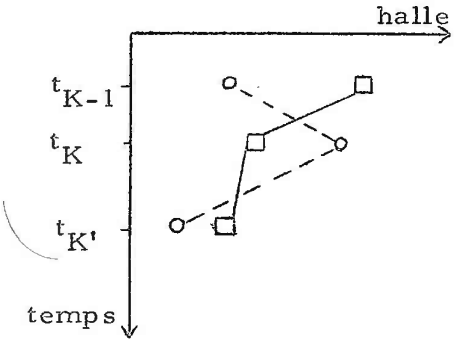
Cas n° 1



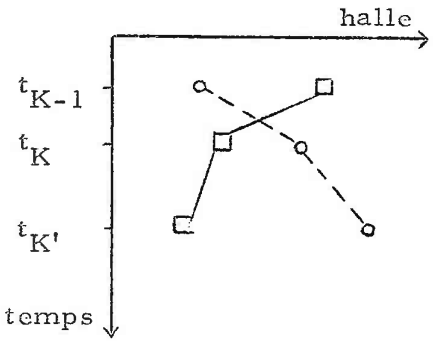
Cas n° 2

Le pont i est représenté par un o

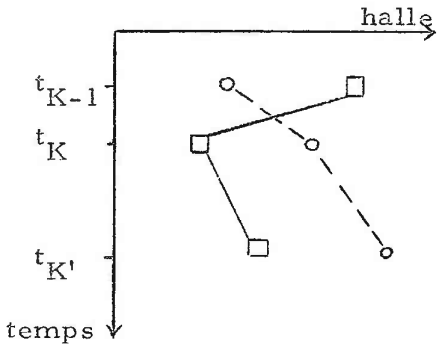
Le pont i+1 " par un □



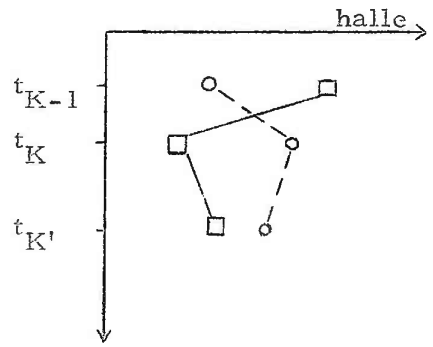
Cas n° 3



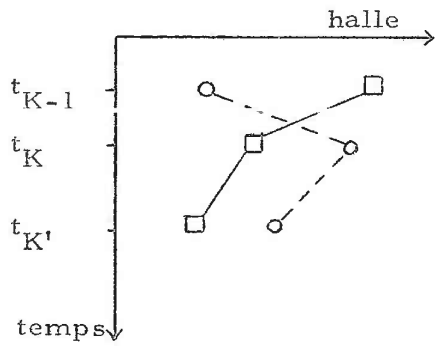
Cas n° 4



Cas n° 5



Cas n° 6



Cas n° 7

Commentaires : Le choix du pont à déplacer (but de l'étude) apparaît nettement dans les cas n° 2 et n° 3. On fera attendre et on déplacera le pont i dans le cas n° 2, le pont i+1 dans le cas n° 3 car l'autre pont (celui auquel on a accordé la priorité) ne fait une incursion dans la zone conflictuelle que le temps de réaliser sa composante. Pour les cinq cas restants nous avons accordé la priorité au pont dont la position future est en état d'INVersion même si ce n'est pas suffisant pour restaurer immédiatement la position relative NORmale (pour le cas n° 1, nous avons choisi de faire déplacer et attendre le pont i+1).

Résumons les règles que nous avons fixées selon les cas

n° du cas	1	2	3	4	5	6	7
choix du pont déplacé	i+1	i	i+1	i	i	i+1	i+1

Nous avons ainsi vu en détail les trois critères d'analyse dont nous avons muni la maquette pour lui permettre de choisir le pont à déplacer (celui dont la priorité est la plus faible). Les deux premiers critères sont d'une mise en place facile et donnent de bons résultats. Le troisième critère peut paraître d'une mise en oeuvre complexe mais sa réalisation technique est aisée. De toutes façons, c'est à l'utilisateur de choisir parmi ces trois critères celui (ou ceux) qui lui semble(nt) le(s) mieux convenir à l'étude qu'il mène et aux objectifs qu'il s'est fixé. Pour cette raison, nous avons tenu à en faire un exposé détaillé.

L'étude de l'activité d'analyse ainsi faite, pour terminer l'exposé de la résolution du problème par la "Maquette", il nous faut voir le retour à une situation normale.

(A2) Etude du retour à une situation normale du système-pont

Lorsque se produit une situation anormale (c'est à dire en infraction avec les contraintes imposées au système-pont) nous désirons restaurer, par programme, une situation du système-pont compatible avec ces contraintes. Nous proposons trois règles que la "Maquette" devra utiliser pour rendre à nouveau normal l'état du système-pont.

Ce sont :

a) déplacer le pont non prioritaire (et ce faisant, générer un déplacement à vide)

b) déplacer le pont non prioritaire et le faire attendre (et ce faisant générer à la fois un déplacement à vide et une attente en charge)

c) retarder le déclenchement d'une tâche (si le pont chargé de la réaliser n'est pas encore libre et inactif)

Dans le cas a) : il lui faut calculer l'endroit où le pont non prioritaire doit se rendre pour ne pas géner. Cet endroit sera la case contigüe à celle occupée par le pont prioritaire (à gauche ou à droite selon que le pont prioritaire est le pont i+1 ou i)

Dans le cas b) : il lui faut calculer d'une part l'endroit où le pont déplacé doit se rendre (le calcul se fera de la même façon que dans le cas a),

d'autre part,

la durée de l'attente en charge. Cette durée sera celle de la composante qu'effectuera pendant ce temps le pont prioritaire. En effet, la "Maquette" laissera le pont non prioritaire reprendre son travail quand le pont prioritaire aura fini d'occuper la zone conflictuelle.

Dans le cas c) : il lui faut calculer de combien sera le retard apporté au déclenchement d'une tâche. Ce retard sera calculé de telle sorte que le nouvel instant de déclenchement corresponde à l'instant auquel le pont chargé de la réalisation devient libre et inactif et peut ainsi effectuer la tâche.

Ainsi les fichiers auxquels la "Maquette" peut avoir accès, lors de l'utilisation des règles de traitement, sont les fichiers paramètres. Les opérations que la "Maquette" peut effectuer sur ces fichiers sont :

α) des créations d'enregistrements pour les fichiers paramètres annexes (déplacement à vide et attente en charge)

β) des modifications d'enregistrements pour le fichier paramètre principal (le processus)

Intérêt :

Cet aspect "simulation" est très important, en particulier au point de vue de la formation. Son utilisation peut être la suivante :

On fournit, comme données fixes à la "Maquette", les deux fichiers permanents représentatifs de l'activité de production.

Puis on demande à un "sujet" (ingénieur, agent de maîtrise, pontonnier, ouvrier au sol) de proposer un processus (i-e de donner pour chaque tâche son instant de déclenchement et le pont chargé de la réalisation). Ces données, une fois codifiées, sont introduites dans la "Maquette" et celle-ci imprime la représentation graphique du processus proposé par le sujet. Si, à un moment donné, le processus devient irréalisable (par exemple parce que deux ponts sont amenés à travailler en même temps au même endroit), la "Maquette" indique qu'il y a

situation-problème à cet instant et propose une solution à ce problème. En effet, une fois la situation redevenue normale, la "Maquette" peut continuer l'étude et la représentation du processus proposé jusqu'à son achévement. Cela permet ainsi d'en avoir une vue globale. Les enseignements que l'on peut tirer de cette "simulation" devraient être intéressants. Ils permettraient de cerner la connaissance qu'ont les opérateurs de l'activité de l'ensemble du système de production à laquelle ils participent : connaissance des durées moyennes de chaque tâche, secteur géographique nécessaire à leur réalisation. Cela permettrait aussi de voir la façon dont chaque opérateur effectue la répartition des tâches entre les ponts :

α) si cette répartition est judicieuse et compatible avec les règles imposées.

β) si certains ponts sont privilégiés par rapport à d'autres (ceci grâce aux indicateurs de durée d'occupation et de déplacements).

Cela permettrait aussi de voir l'importance que chacun attache aux espaces de sécurité entre les ponts.

B - ANALYSE FONCTIONNELLE

(I) Fichier et Table permanents concernant l'activité de production.

① Table "TTACH" fichier des tâches

Revoir la définition donnée au chapitre II page 21.

② Fichier "TCOMP" fichier des composantes

Revoir la description donnée au chapitre II page 22.

(II) Fichiers paramètres concernant le processus à traiter

① Fichier "TPROC" fichier du processus

Revoir la description donnée au chapitre II page 23.

② Fichier "TPONT" fichier d'occupation des ponts

Revoir la description donnée au chapitre II page 24.

(III) Les traitements

Le programme définit :

Ⓐ A l'instant t_i ($t_i = t_{i-1} + \Delta t$)

A-1 - Si l'état du système est anormal

1 - La position des ponts concernés dans le conflit

2 - La solution proposée pour résoudre le problème posé.

A-2

1 - L'instant t_i auquel on étudie l'état du système-pont

2 - Pour chaque pont

a) sa position
b) son numéro
c) son état
d) le numéro de la tâche qu'il effectue

3 - Pour chaque couple de ponts voisins

a) l'espacement entre ces deux ponts

Ⓑ A la fin du processus traité

B-1 Pour chaque pont

a) la longueur du chemin parcouru

b) la durée totale du travail qu'il a effectué

B-2 Pour le système-pont

a) le planning de l'activité du système-pont, c'est à dire, rangés par ordre chronologique, les ordres suivants donnés au système-pont:

- A telle heure, on lance la réalisation de telle tâche

- A telle heure, tel pont va en telle case

- A telle heure, on fait attendre tel pont tant de temps.

DESCRIPTION MODULAIRE

NIVEAU I

I - TRAITEMENT

1 - Lecture des fichiers

2 - Sortie de la ligne de référence

3 - Pour chaque Δt

| 4 - Etude du processus
| 5 - Etude de modification
| 6 - Etude de situation conflictuelle
| 7 - Sortie de ligne

8 - Sortie du planning

Déclaration : fichiers TPONT, TPROC, TTACH, TCOMP

N.B. : Comme pour la description modulaire vue au chapitre II, afin de faciliter l'impression et la mise en page, la présence d'un trait continu simule le changement de pages.

NIVEAU II

I-1 - Lecture des fichiers

1 - Lecture du fichier TPROC

2 - Lecture du fichier TTACH

3 - Lecture du fichier TCOMP

4 - Lecture du fichier TPONT

Déclaration : les 4 fichiers

NIVEAU II

I-2 - Sortie de la ligne de référence

Module inchangé. Revoir la description au chapitre II, page 16.

NIVEAU II

I-4 - Etude du processus

1 - Lecture du fichier processus

2 - Pour chaque tâche débutant à cet instant

| 3 - Relevé du numéro de la tâche
| 4 - Recherche des composantes
| 5 - Si le pont chargé de la réalisation n'est pas libre et inactif
| | 6 - Impression d'un libellé le signalant
| | 7 - Différer le déclenchement de cette tâche de la durée nécessaire à la libération du pont

Déclaration : fichiers TPROC, TCOMP, TPONT, TTACH

NIVEAU II

I-5 - Etude de modification

Module inchangé. Revoir la description au chapitre II, page 18 (module I-6).

NIVEAU II

I-6 - Etude de situation conflictuelle

1 - Pour chaque couple de ponts voisins

2 - Si l'ordre est respecté, passer directement à la sortie de ligne (I-7)

3 - Sinon, faire

4 - Analyse du conflit par l'étude des critères

5 - Calcul des éléments nécessaires à la résolution du conflit

6 - Générations ou modifications des fichiers concernés.

Déclaration : fichiers TTACH, TCOMP, TPROC, TPONT, AT, DV.

NIVEAU II

I-7 - Sortie de ligne

Module inchangé. Revoir la description au chapitre II, page 19 (module I-8).

NIVEAU II

I-8 - Sortie du planning

1 - Interclassement des trois fichiers paramètres représentatifs de l'activité du système-pont (le processus proprement dit, les attentes en charges et le déplacement à vide)

2 - Impression du tableau ainsi obtenu (sortie en ordre chronologique)

Déclaration : fichiers TPROC, AT, DV

CHAPITRE IV

Par le programme SIMUL que nous venons d'expliciter au chapitre III, nous pouvons, à partir des deux fichiers permanents (tâche et composante), du fichier paramètre principal et de règles de traitement, simuler et éventuellement aménager différentes activités du système ponts roulants proposées par un opérateur.

L'aménagement éventuel de l'activité se faisait par génération des fichiers paramètres annexes (attentes en charge et déplacement à vide).

Dans ce programme SIMUL, un seul fichier paramètre restait fourni en entrée à la "Maquette" : le fichier paramètre principal (du processus). En fournissant à la "Maquette" de nouvelles règles de traitement nous essayerons dans le programme ROSEC de lui faire générer ce dernier fichier paramètre.

Comme dans le chapitre précédent, nous supposons, a priori, toutes les tâches libres, c'est à dire sans aucune contrainte entre elles. Mais, pour des raisons de commodité, contrairement au précédent programme SIMUL, on n'impose pas le fait que certaines tâches ne peuvent pas être faites par certains ponts. En d'autres termes, tout pont est susceptible de réaliser toute tâche.

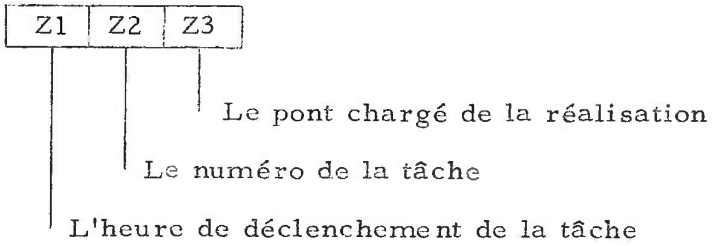
A une activité déterminée du système-pont correspond un fichier paramètre "Processus". Ce fichier contient les éléments qui permettent d'indiquer pour chaque tâche de ce processus :

α) son instant de déclenchement

β) le numéro du pont chargé de sa réalisation.

Ces renseignements associés à une tâche sont strictement dépendants de l'activité à laquelle appartient cette tâche.

Les enregistrements de ce fichier ont la composition suivante :



Ce fichier est rangé par ordre chronologique croissant.

Disposant des seuls fichiers permanents (i-e de l'ensemble des tâches et de leurs caractéristiques) et d'un ensemble de règles de traitement, par programme, nous devons, pour générer les éléments de ce fichier Processus :

α) Associer à toute tâche un instant de déclenchement. Nous effectuons un ordonnancement en réalisant cette association pour toutes les tâches.

β) Associer à toute tâche un pont qui sera chargé de la réaliser. Nous effectuerons une affectation en réalisant cette association pour chaque tâche.

Selon l'ordre dans lequel nous effectuons ces deux opérations, nous obtenons deux méthodes différentes dans lesquelles l'objectif sécurité prend des aspects différents.

Dans ce chapitre IV, nous explicitons la méthode dite par les Bandes (PLB) qui réalise d'abord l'affectation des tâches aux ponts, l'ordonnancement des tâches n'étant fait qu'ensuite par l'utilisateur.

La méthode réalisant en premier lieu l'ordonnancement des tâches aux ponts, l'affectation n'étant réalisée qu'après, est appelée méthode par les contours (ou encore par les cartons). Méthode PLC , elle fera l'objet du chapitre V.

Pour l'étude de la méthode PLB , nous verrons successivement :

A) L'exposé de la méthode et son exploitation

B) L'analyse fonctionnelle

A - L'exposé de la méthode

A-1) Son objectif

Nous savons déjà que cette méthode a pour but initial de réaliser l'affectation des tâches aux ponts. Mais cette affectation peut être réalisée de diverses manières selon l'objectif que l'on veut atteindre. Cette méthode PLB , basée sur un critère de "position" que nous allons développer, vise un double objectif.

① Améliorer le niveau de sécurité en permettant la suppression des problèmes de frontières entre les pontonniers. Par problèmes de frontières, nous entendons les problèmes qui peuvent se poser à la suite de co-activité sur un même lieu géographique. Chaque pontonnier, habitué à travailler dans une certaine zone, y prend plus de risque que lorsqu'il travaille dans un secteur où il a peu l'occasion de se rendre. Dans le cas où ces zones peuvent se chevaucher, les risques d'accidents (collisions par exemple) sont accrus.

② Répartir équitablement le travail entre les quatre ponts.

A-2) Son principe

La méthode consiste, à partir de l'ensemble E des tâches à affecter, à diviser automatiquement la halle en autant de zones qu'il y a de ponts. Chacune de ces zones devient alors le secteur préférentiel d'un pont. Après une répartition des tâches en diverses catégories selon leurs positions par rapport à ces zones, les tâches se déroulant dans un secteur sont affectées au pont chargé de ce secteur. Pour les tâches couvrant plusieurs secteurs, elles sont affectées au pont (d'un des secteurs concernés) qui est le moins occupé.

A-3) Exposé détaillé

A-3-1) Division de la halle en autant de bandes qu'il y a de ponts.

L'objectif de cette partition de la halle est d'obtenir des zones telles que les opérations qui se dérouleront à l'intérieur de chacune d'elles soient d'une durée sensiblement proportionnelle à des coefficients qui leur seront affectés, a priori , selon que l'on veut privilégier l'une ou l'autre de ces zones (ou l'un ou l'autre de ces ponts). Dans nos exemples, nous avons pris les quatre coefficients égaux à 1 afin de satisfaire à notre objectif qui est, en outre, d'équirépartir l'activité entre les quatre ponts.

Nous nous préoccupons alors de la durée de cette activité, donc des tâches qui la composent. En accord avec les définitions que nous avons données pour les tâches et les composantes, la durée d'une tâche est la somme des durées de ses composantes. A chacune de ces composantes correspond une case de la halle. Donc, lors de la réalisation d'une tâche, chaque case de la halle sera effectivement occupée le temps que dure(nt) la (ou les) composante(s) qui s'y déroule(nt).

On pourrait, en effectuant ce travail pour toutes les tâches que nous avons à traiter, obtenir la répartition exacte de la durée d'activité entre toutes les cases de la halle.

En réalité, nous nous contenterons d'une approximation. En effet, deux composantes paraissent plus importantes parmi toutes celles qui composent une tâche i , ce sont celles qui occupent les cases extrêmes (i-e la case la plus à gauche et la case la plus à droite dans le plan de la halle (revoir le plan chapitre I)).
L'intervalle situé entre ces deux cases forme ce que nous appellerons la "Bande de Travail" de la tâche i . Nous utiliserons souvent, par la suite, cette notion, très importante au point de vue de la sécurité, de Bande de travail. Pour chaque tâche nous relevons les renseignements relatifs

a) à la case extrême la plus à droite (renseignements rangés dans le tableau NAX)

b) à la case extrême la plus à gauche (renseignements rangés dans le tableau NIN)

En réalité, chaque tableau est formé de deux sous-tableaux. Pour chaque case extrême de la tâche i , nous relevons :

1) son numéro (élément d'indice pair)

2) le numéro de la composante pour laquelle elle est obtenue (élément d'indice impair).

Ces éléments sont rangés en NAX(2i) et NAX(2i+1) (respectivement NIN(2i), NIN(2i+1)).

Voyons un exemple avec la représentation graphique d'une tâche que nous avons développée dans le chapitre II.

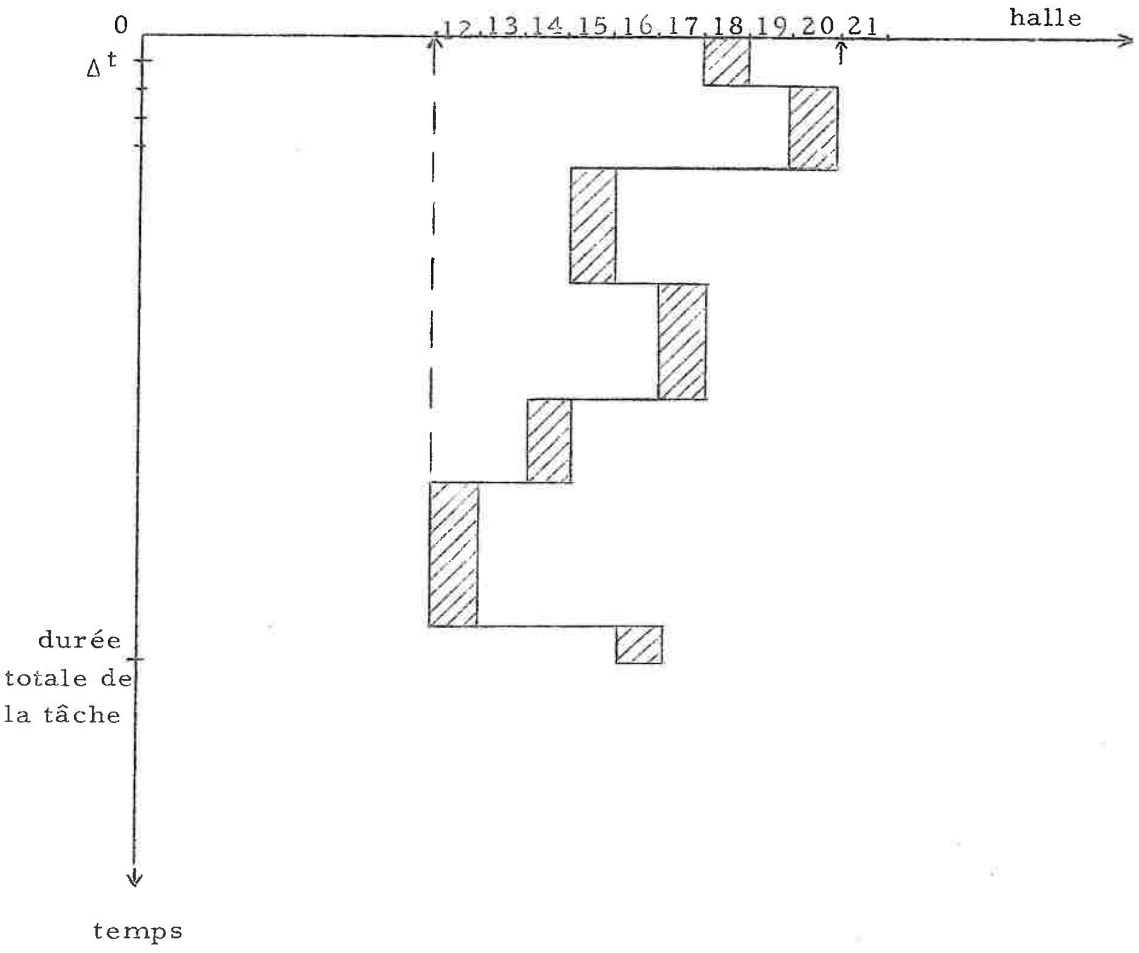


Figure n° IV. F1

La case la plus à gauche (que nous appelons NIN(2i)) est la case n° 12.

La case la plus à droite (NAX(2i)) est la case 20.
Elles sont respectivement obtenues pour les composantes n° 2 et 6 (NIN(2i+1) et NAX(2i+1)).

La Bande de travail de la tâche i est limitée par les cases 12 et 20.

L'approximation dont nous nous contenterons consiste à affecter à chacune des deux composantes extrêmes la moitié de la durée totale de la tâche. Nous parlerons alors de "durée généralisée" des composantes. Sur l'exemple graphique nous aurons alors

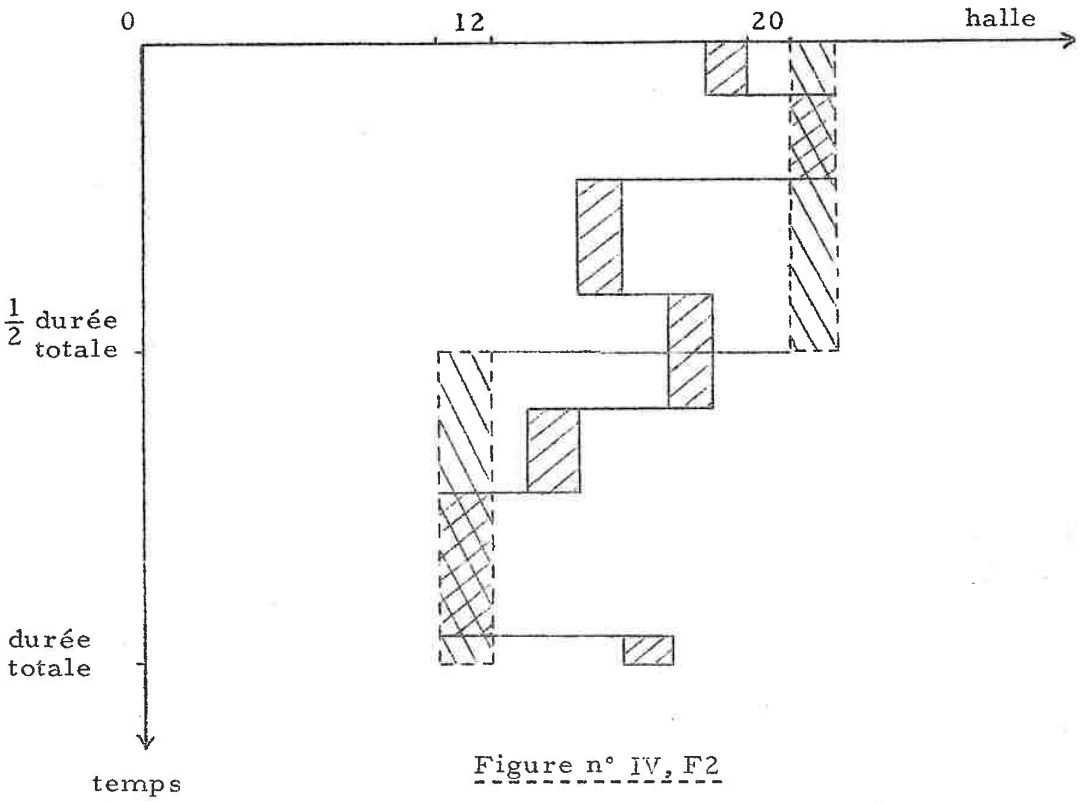


Figure n° IV, F2

En prenant les tâches les unes après les autres et en y appliquant cette approximation, nous calculons pour chaque case la "durée généralisée" totale, somme des durées généralisées des composantes qui l'occupent. Ayant ainsi la "durée généralisée totale" (DGT(n)) d'occupation de chaque case n, nous allons déterminer la largeur de chacune des zones que nous voulons obtenir et pour cela fixer les cases limites de ces zones.

Explicitation de la Méthode sur le graphique n° IV.α

Nous portons en abscisse les cases numérotées (de 1 à NT) de la halle, en ordonnée le temps.

A chaque case n° i nous faisons correspondre la durée d'occupation de la partie de la halle comprise entre l'extrêmité de cette halle (la case n° 1) et cette case n° i (incluse). Nous effectuons donc le cumul des durées d'occupation de chaque case.

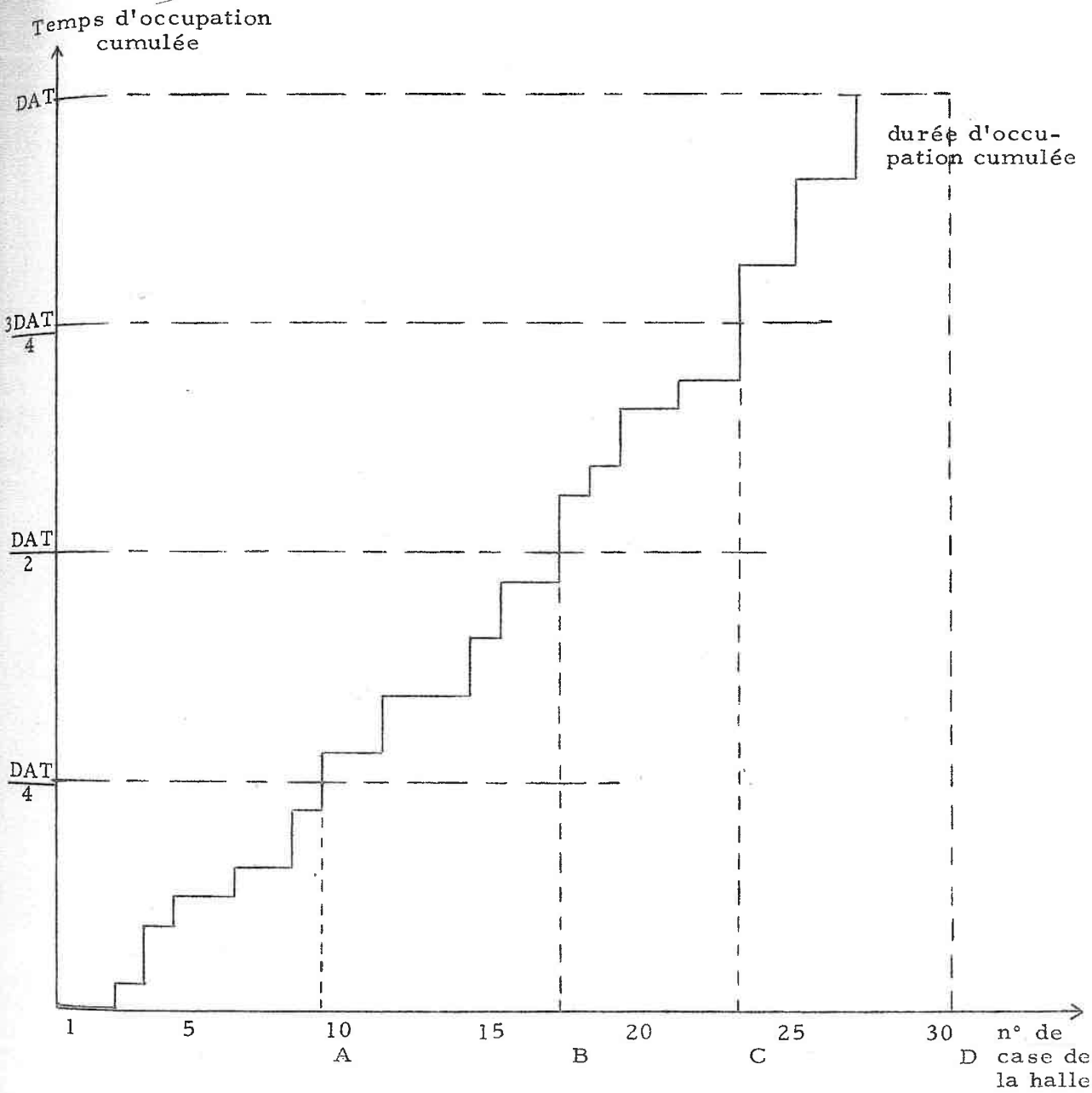
Nous obtenons ainsi une fonction monotone croissante étagée. La valeur prise par cette fonction pour la case de numéro le plus élevé (NT) est égale à la somme des durées de chaque tâche à réaliser dans la halle. Soit DAT cette somme. Traçons la courbe représentative (graphique IV.α).

Nous cherchons à obtenir une partition géographique de la halle en quatre secteurs tels que les sommes des durées des travaux qui s'y déroulent soient égales entre elles (donc au quart de la somme totale DAT). Connaissant la durée de cette somme totale, nous allons tracer quatre droits, parallèles à l'axe des abscisses, dont les intersections avec l'axe des temps seront respectivement égales à

$$\frac{DAT}{4}, \frac{2DAT}{4}, \frac{3DAT}{4}, \frac{4DAT}{4}$$

Les abscisses des points d'intersection de ces droites avec la courbe nous donneront les bornes supérieures des secteurs recherchés. Nous prendrons les cases dont les durées d'occupation sont immédiatement supérieures, ce sont sur notre graphique les cases A, B, C, D*.

* paramètres qui pourront être modifiés par l'utilisateur dans le cadre de l'aide à la décision.



Graphique n° IV.α

Dans le cas où l'intersection de l'une de ces droites avec la courbe n'est pas réduite à un point, c'est à dire le cas où cette droite rencontre la courbe à un palier horizontal de celle-ci, l'abscisse que nous retiendrons sera celle de la première case dont la durée d'occupation cumulée est strictement supérieure à la durée représentée par le palier.

En formalisant, nous obtenons :

1°) Calculer, pour chaque case n, la somme cumulée des "durées généralisées totales" des cases situées à sa gauche (i-e de numéro inférieureà la sienne).

- Soit FCUMUL(n) la somme cumulée des "durées généralisées totales" d'occupation des cases de n° inférieur à n

- Soit DGT(i) la "durée généralisée totale" d'occupation de la case n° i

- Soit NT le nombre de cases de la halle

Nous avons
$$\begin{cases} FCUMUL(1) = DGT(1) \\ FCUMUL(n) = FCUMUL(n-1) + DGT(n) \end{cases}$$

- La durée totale des tâches effectuées par le système-pont sera égale à FCUMUL(NT). Soit FMAX cette durée.

2°) Calculer, en fonction des coefficients attachés à chacune des zones, les durées d'activité cumulées que l'on veut voir se réaliser dans ces zones.

- Soient K1(i) ces coefficients (i=1, 4)

- Soient KF(i) (i=1, 4) les durées recherchées qu'on obtient comme suit :

$$KF(i) = FMAX * \frac{\sum_{j=1}^{i} K1(j)}{\sum_{j=1}^{4} K1(j)}$$

N. B. : Dans l'exemple graphique, nous avons pris K1(j)=1 (j=1, 4).
La Méthode reste valable quelles que soient les valeurs de K1(j).

3°) Rechercher les cases limites de chacune des bandes telles que la borne inférieure soit incluse dans la bande et la borne supérieure exclue.

. Pour la bande 1 : la borne inférieure est, bien sûr, 0

la borne supérieure est M1 telle que :

$FCUMUL(M1-1) \leq KF(1) < FCUMUL(M1)$

. Pour la bande i : la borne inférieure est M_{i-1}

la borne supérieure est M_i telle que :

$FCUMUL(M_{i-1}) \leq KF(i) < FCUMUL(M_i)$

. Pour la dernière bande (i=4) :

la borne inférieure est M_3

la borne supérieure est NT+1.

Nous avons ainsi effectué la partition de la halle en autant de bandes qu'il y a de ponts. Chacune de ces bandes devient secteur préférentiel du pont correspondant. Par secteur préférentiel, nous entendons

que le pont sera chargé de toute tâche se déroulant dans la zone à laquelle il correspond. Pour cela nous allons classer, maintenant, les tâches en trois catégories, selon la position de leurs composantes extrêmes par rapport aux zones ainsi définies.

A-3-2) Répartition des tâches

Les tâches sont réparties selon la position de leur bande de travail en trois catégories, les monobandes, les bibandes et les tribandes.

. Une tâche est dite monobande si ses composantes extrêmes appartiennent à la même bande

. Une tâche est dite bibande si ses deux composantes extrêmes appartiennent à deux bandes distinctes mais contigües.

. Une tâche est dite tribande si ses deux composantes extrêmes appartiennent à deux bandes distinctes et disjointes.

A-3-3) Règles d'affectation

Affectation proprement dite des tâches aux ponts.

. Si la tâche est monobande, elle est affectée d'office au pont chargé du secteur à l'intérieur duquel elle se déroule.

. Si la tâche est bibande, elle est affectée à celui des deux ponts concernés dont la durée d'occupation est, à ce moment là, la plus faible

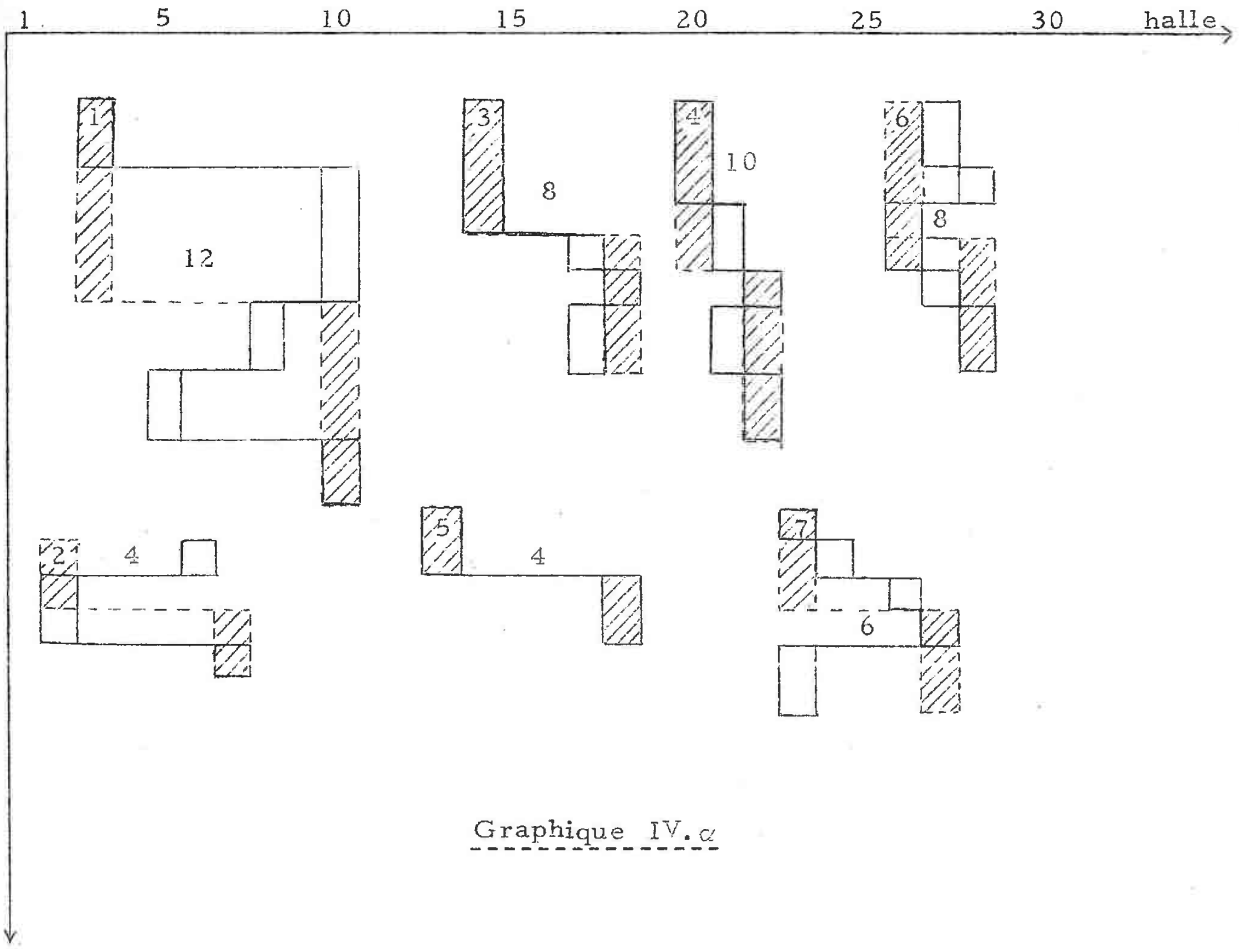
. Si la tâche est tribande, elle est affectée à l'un des ponts concernés. Dans ce cas les ponts concernés sont :

1) d'une part les deux ponts relatifs aux bandes dans lesquelles se déroulent les composantes extrêmes

2) d'autre part le(s) pont(s) compris entre deux ponts. Le pont choisi est, là encore, celui dont la durée d'occupation est la plus faible.

A-3-4) Application à un exemple simple

α) Exemple



Graphique IV.α

β) Calcul de la durée généralisée totale (DGT) de chaque case et de la somme cumulée de ces valeurs (FCUMUL)

Les cases ne figurant pas dans le tableau ci-dessous ont une durée généralisée totale d'occupation égale à 0 et donc la fonction somme cumulée ne change pas de valeur pour ces cases.

n° de cases	2	3	7	10	13	14	18	20	22	23	26	27	28
DGT	2	6	2	6	2	4	6	5	5	3	4	3	4
FCUMUL	2	8	10	16	18	22	28	33	38	41	45	48	52

Tableau IV.β1

Afin d'obtenir une équirépartition de l'activité entre les quatre ponts, nous avons pris 1 comme valeur pour les quatre coefficients attachés à chacune des bandes.

$K1(j) = 1$ avec $j=1,4$

De ces coefficients et de la valeur de la durée totale de l'activité FMAX(=52), on tire les valeurs de KF(i) (i=1,4) valeurs permettant d'obtenir les bornes délimitant les bandes

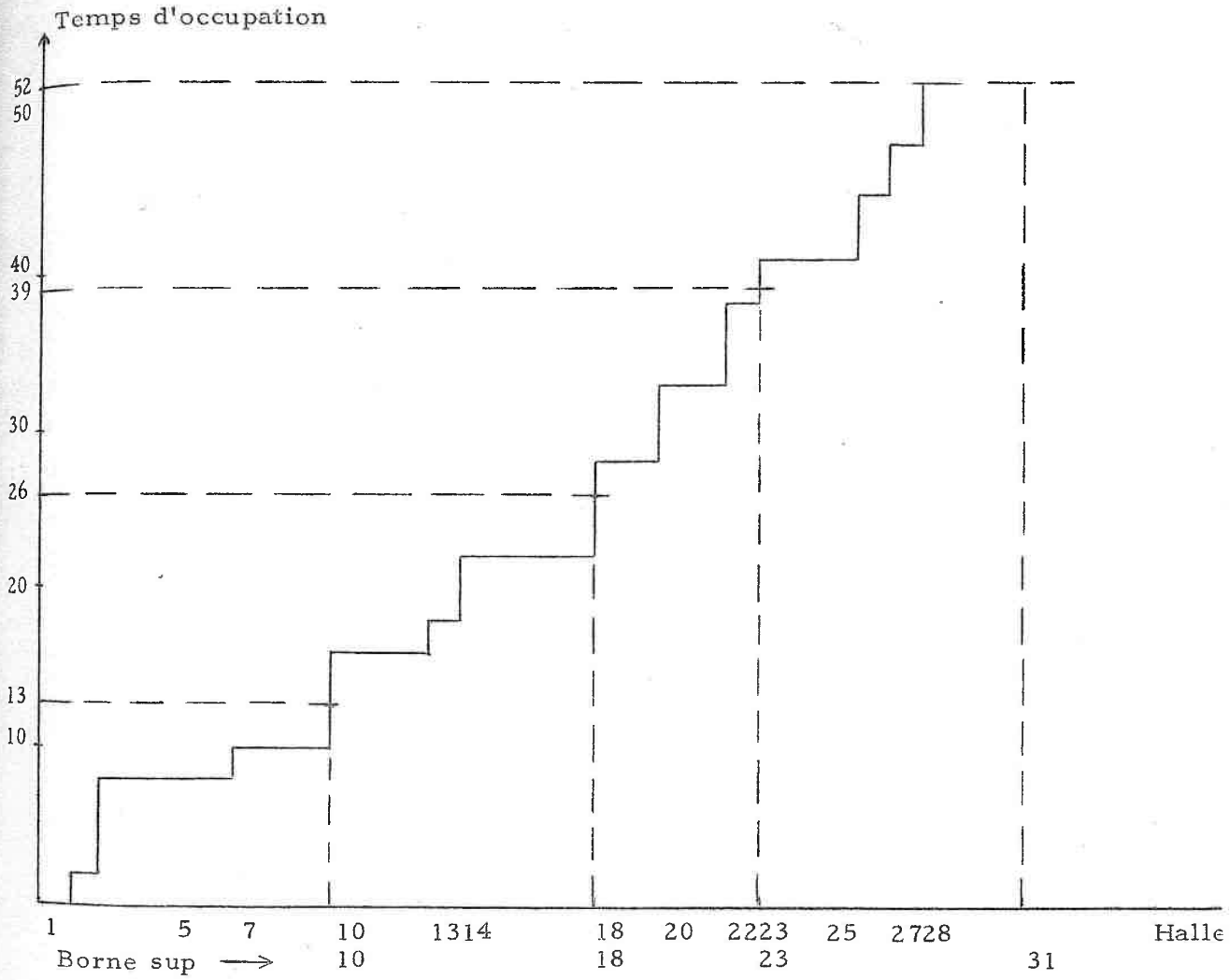
$KF(1) = 52 \times \frac{1}{4} = 13$ $KF(2) = 26$ $KF(3) = 39$ $KF(4) = 52$

On reporte les divers éléments du tableau IV.β1 sur le graphique IV.β2

Les bandes sont alors les suivantes :

Bande(1) = [1, 10[Bande(3) = [18, 23[

Bande(2) = [10, 18[Bande(4) = [23, 30]



Graphique IV.82

γ) Répartition des tâches en catégories

Tâche n°	1	2	3	4	5	6	7
Catégorie	Bi	Mono	Bi	Mono	Bi	Mono	Mono

On voit, sur ces résultats, que la partition de la halle telle que nous l'avons effectuée divise les tâches en un groupe de 4 monobandes et en un groupe de 3 bibandes.

δ) Affectation

En appliquant la règle adoptée on obtient les résultats suivants :

Tâche n°	1	2	3	4	5	6	7
n° du pont chargé de la réalisation	1	1	2	3	2	4	4

On calcule également la durée effective de l'activité de chacun des ponts.

n° Pont	Durée (unité = Δt)
1	16
2	12
3	10
4	14

. Compte tenu du fait que la durée théorique de travail des ponts est de 52/4 = 13. L'affectation donne donc un bon résultat. Mais ceci porte sur un simple exemple. Nous allons voir dans le paragraphe suivant ce que donne cette méthode appliquée à divers jeux d'essais.

A-4) Exploitation des résultats

Rappelons que les deux objectifs visés sont :

d'une part : la suppression de problème de frontières entre les ponts et par là une augmentation du niveau de sécurité. En d'autres termes l'obtention de tâches monobandes en plus grand nombre possible puis de tâches bibandes et ceci grâce à la partition de la halle en autant de secteurs préférentiels que de ponts.

d'autre part : l'obtention d'une répartition équitable du travail entre les ponts.

Nous avons appliqué cette méthode à six jeux d'essais différents. Les quatre premiers jeux d'essais comporte chacun 62 tâches, le cinquième en comporte 24 et le sixième 12. Tous doivent être réalisés par quatre ponts. Un des quatre premiers jeux comporte les 62 tâches différentes qui forment le processus de production d'une aciérie.

A-4-1) Etude selon l'objectif : "Répartition équitable"

Nous avons regroupé les résultats dans les tableaux n° IV. 1, IV. 2, IV. 3, IV. 4, IV. 5, IV. 6. Les éléments qui nous permettent d'émettre une opinion sur l'efficacité de la méthode sont les suivants :

α) La durée de l'activité qui se déroule dans chacune des quatre bandes construites. Ces quatre durées figurent dans la première ligne.

β) La durée de travail effectif de chacun des quatre ponts. Ces quatre durées figurent dans la deuxième ligne.

γ) Les coefficients d'activité de ces ponts en rapportant la durée de travail de chacun des ponts à la durée totale de l'activité du système-ponts. Ces coefficients seront à rapprocher des coefficients théoriques d'activité dans le cas d'une équirépartition exacte entre les ponts, ces coefficients théoriques sont de 0, 25.

Jeu n° 1 : 62 tâches

Durée totale de l'activité du système-pont = 1849 unités de temps (Δt) (ici $\Delta t = 15$ secondes, donc environ 7h 40').

Durée théorique de l'activité d'un pont $\simeq 462$

durée d'occupation des bandes	374	524	451	482
durée effective de l'activité des ponts	376	360	559	554
coefficients d'activité des ponts	0, 203	0, 196	0, 302	0, 298

Tableau IV. 1

Jeu n° 2 62 tâches

Ce jeu d'essai est très proche du jeu d'essai précédant par sa composition (il ne dure que quatre minutes de moins environ). Nous verrons que la méthode donne des résultats homogènes à ceux obtenus plus.haut.

Durée totale de l'activité du système-pont = 1831

Durée théorique de l'activité d'un pont ≃ 458

durée occupation bande	410	504	451	466
durée effect. activité ponts	371	364	554	542
coeff. activité ponts	0, 203	0, 198	0, 302	0, 296

Tableau IV. 2

Jeu n° 3 62 tâches

Durée totale de l'activité du système-pont = 1319 (environ 5h 30')

Durée théorique de l'activité d'un pont ≃ 330

durée bandes	265	371	346	337
durée ponts	303	335	345	336
coeff. activité ponts	0, 230	0, 254	0, 262	0, 254

Tableau IV. 3

Jeu n° 4 62 tâches

Durée totale de l'activité du système-pont = 1171 (environ 4h 50')

Durée théorique de l'activité d'un pont ≃292

durée bandes	280	270	305	316
durée ponts	262	312	302	295
coeff. activité ponts	0, 223	0, 266	0, 258	0, 252

Tableau IV. 4

Jeu n° 5 24 tâches

Durée totale de l'activité du système-pont = 486 (environ 2h)

Durée théorique de l'activité d'un pont ≃121

durée bandes	117	114	127	128
durée ponts	128	127	105	126
coeff. activité ponts	0, 263	0, 261	0, 216	0, 259

Tableau IV. 5

Jeu n° 6 12 tâches

Durée totale de l'activité du système-pont = 255 (environ 1h 4')

Durée théorique de l'activité d'un pont ≃64.

durée bandes	46	57	71	80
durée ponts	63	92	45	55
coeff. activités ponts	0, 247	0, 360	0, 176	0, 215

Tableau IV. 6

De plus, afin de permettre une visualisation rapide de ces résultats nous avons regroupé tous les résultats α) et β) dans un graphique en les rapprochant, ici encore, de la durée théorique des ponts. Bien sûr, ce graphique n'apporte rien de nouveau par rapport aux éléments précédents, mais il est une aide utile, par sa rapidité de lecture, dans le diagnostic de la valeur des résultats.

Nous présentons par graphique les résultats de deux jeux d'essai.

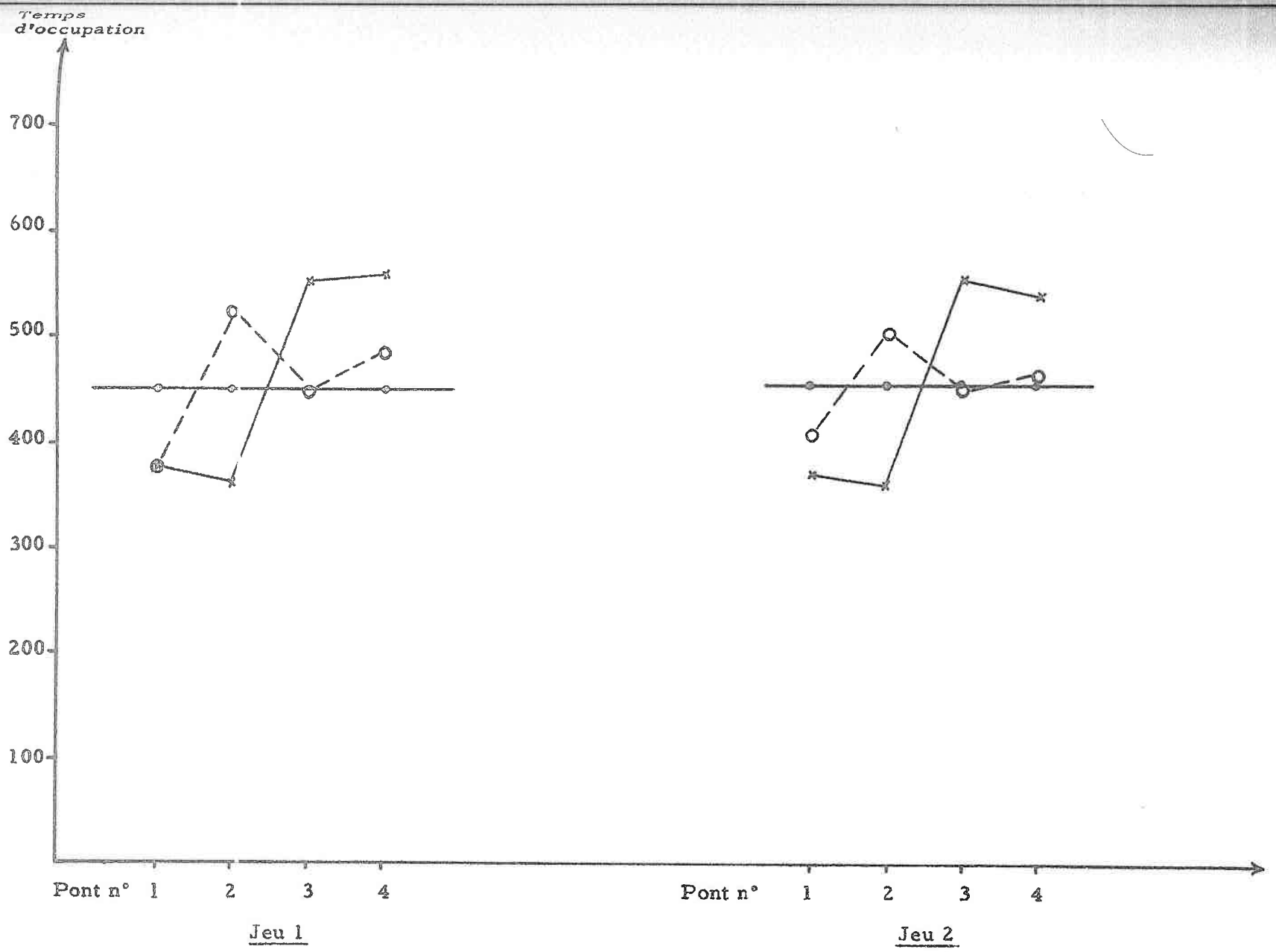
Pour chaque jeu, nous portons en abscisse les quatre numéros de pont et en ordonnée le temps d'occupation.

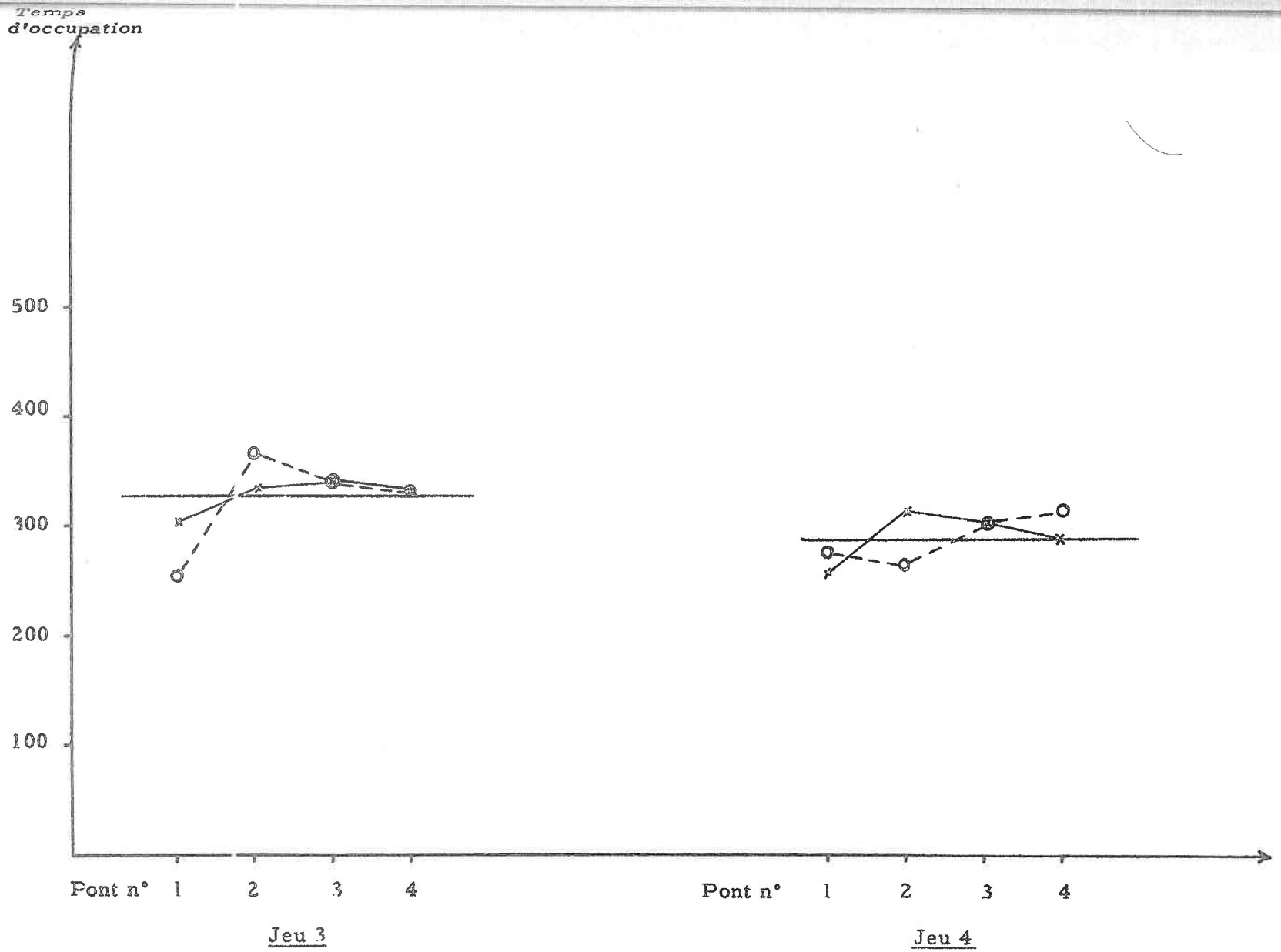
Nous portons sur ce graphique pour chaque pont :

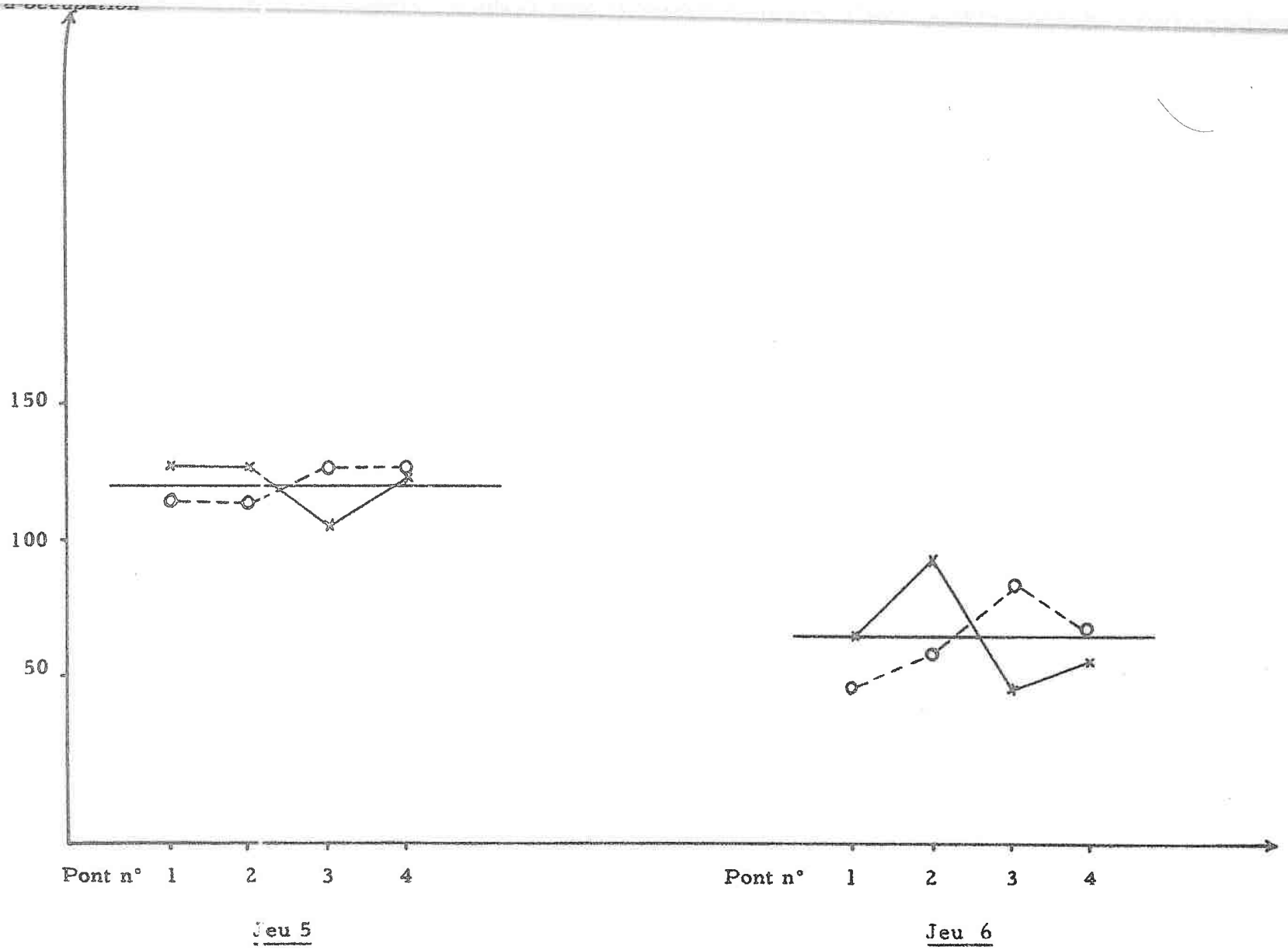
. 1) sa durée théorique de travail (.) . La durée théorique d'activité étant identique pour chaque pont, la droite joignant ces points est parallèle à l'axe des abscisses.

. 2) sa durée effective de travail (x)

. 3) la durée de l'activité se déroulant dans la bande à laquelle ce pont est associé (0).


Temps
d'occupation
700
600
500
400
300
200
100
Pont n° 1 2 3 4
Jeu 1
Pont n° 1 2 3 4
Jeu 2


Temps
d'occupation
500
400
300
200
100
Pont n° 1 2 3 4
Jeu 3
Pont n° 1 2 3 4
Jeu 4



Jeu 5

Jeu 6

Pour cet objectif "Répartition équitable" les résultats fournis par la "Maquette" peuvent être considérés comme :

α) bons dans les exemples 3, 4 et 5.

En effet, alors que le coefficient théorique d'activité est de 0, 25 pour chaque pont, la valeur des quatre coefficients d'activité effective varie

pour le jeu n° 3	entre	0, 23 et 0, 26
pour le jeu n° 4	entre	0, 22 et 0, 27
pour le jeu n° 5	entre	0, 21 et 0, 26

β) assez bons dans les exemples n° 1 et 2

Toujours pour le même coefficient théorique de 0, 25, les valeurs des quatre coefficients dans les jeux 1 et 2 sont situés entre 0, 20 et 0, 30.

γ) moyens dans l'exemple n° 6

Les quatre coefficients ont une valeur qui varie entre 0, 17 et 0, 36. Une raison de cette faiblesse réside dans le fait que ce jeu d'essai ne comporte que 12 tâches et donc que chacune d'elles a une importance relative plus grande, par sa position et par sa durée.

En conclusion pour cet objectif "Répartition équitable" la méthode donne des résultats satisfaisants.

A-4-2) Etude selon l'objectif : "Suppression des problèmes de frontières"

Nous avons regroupé les résultats dans le tableau n° IV. 7. Les éléments intéressants sont les répartitions des diverses tâches en monobandes, bibandes, tribandes.

Jeu n°	Monobande	Bibande	Tribande
1	14	39	9
2	13	40	9
3	26	31	5
4	22	23	17
5	12	10	2
6	3	5	4

Tableau n° IV. 7

Compte tenu des caractéristiques des divers jeux d'essai, les résultats fournis par la "Maquette" peuvent être considérés comme :

α) bons dans les exemples n° 3 et 5. En effet, le pourcentage des tâches monobandes (celles qui ne posent aucun problème de frontière) y est très important : 40% de l'effectif dans le jeu n° 3, et 50% dans le jeu n° 5.

De plus, le pourcentage des tâches bibandes (posant un problème de frontière limité à deux ponts) est lui aussi élevé (50% dans le jeu n° 3 et de 40% dans le jeu n° 5).

Les tâches dont la réalisation, a priori, posent le plus grand nombre de problèmes de frontières (les tribandes) ne représentent dans les deux jeux que 10% de l'effectif.

β) assez bons dans les jeux n° 1 et 2

Pour ces deux jeux (très semblables) les résultats sont moins bons. En effet, le pourcentage des tâches bibandes augmente au détriment de celui des monobandes (respectivement 63% et 22%).

Mais ils sont encore assez bons compte tenu du fait que le pourcentage des tâches tribandes reste assez faible (15%).

γ) moyens dans les jeux n° 4 et 6

Si dans les deux cas, les pourcentages des tâches monobandes et bibandes sont à peu près équivalentes (35% et 37% dans le jeu n° 4, 25% et 42% dans le jeu n° 6) c'est à un niveau plus faible par rapport au pourcentage des tâches tribandes (i-e les tâches-problèmes). Le pourcentage de celles-ci s'élève à 28% dans le jeu n° 4 et 33% dans le jeu n° 6.

Bien sûr, on peut émettre la même réserve sur le jeu n° 6 pour expliquer partiellement ces résultats.

Usage du programme comme aide à la décision

Une des idées principales de cette recherche est que la "Maquette" est essentiellement un outil d'aide à la décision mis à la disposition de l'utilisateur.

Rappelons que, dépendante des règles de traitement que nous lui fournissons, appliquant des méthodes souvent empiriques, la "Maquette" fournit une solution : cette solution est en général bonne ou relativement bonne, mais on ne peut pas assurer qu'elle soit la meilleure possible. Cela peut arriver mais ce n'est pas systématique Aussi, nous demandons à la "Maquette" de fournir, en plus, des renseignements qui permettront à l'utilisateur, à la vue de ceux-ci, de proposer des modifications à la "Maquette". Celle-ci pourra, alors, utiliser ces éléments proposés par l'utilisateur pour générer la solution correspondante.

Dans le cas présent, l'utilisateur pourra proposer un découpage en 4 bandes de la halle différent de celui proposé par la "Maquette" (et même plusieurs s'il le désire).. Il proposera ses découpages en fonction des objectifs qu'il veut satisfaire, la "Maquette" répartira alors les tâches en catégories et les affectera.

A dessein, nous avons pris un jeu d'essai (le n° 2) dont les résultats fournis par la "Maquette" étaient assez bons (pour les deux objectifs) pour voir quelles améliorations donne cette possibilité d'affinage.

Nous avons proposé cinq découpages légèrement différents, très proches de la solution proposée par la "Maquette".

Nous avons réuni les résultats dans les tableaux suivants. Nous y avons indiqué les mêmes éléments que pour les tableaux n° IV.1 à IV.7.

Découpage 1

durée bandes	503	395	398	536	Monobandes	16
durée ponts	391	365	533	542	Bibandes	37
coeff. activité ponts	0,213	0,199	0,291	0,286	Tribandes	9

Découpage 2

durée bandes	337	561	389	536	Monobandes	17
durée ponts	345	390	538	558	Bibandes	36
coeff. activité ponts	0,188	0,212	0,296	0,304	Tribandes	9

Découpage 3

durée bandes	337	327	531	536	Monobandes	18
durée ponts	345	390	538	558	Bibandes	34
coeff. activité ponts	0,188	0,212	0,296	0,304	Tribandes	10

Découpage 4

durée bandes	337	682	266	536	Monobandes	24
durée ponts	391	378	520	540	Bibandes	31
coeff. activité ponts	0,213	0,206	0,283	0,297	Tribandes	7

Découpage 5

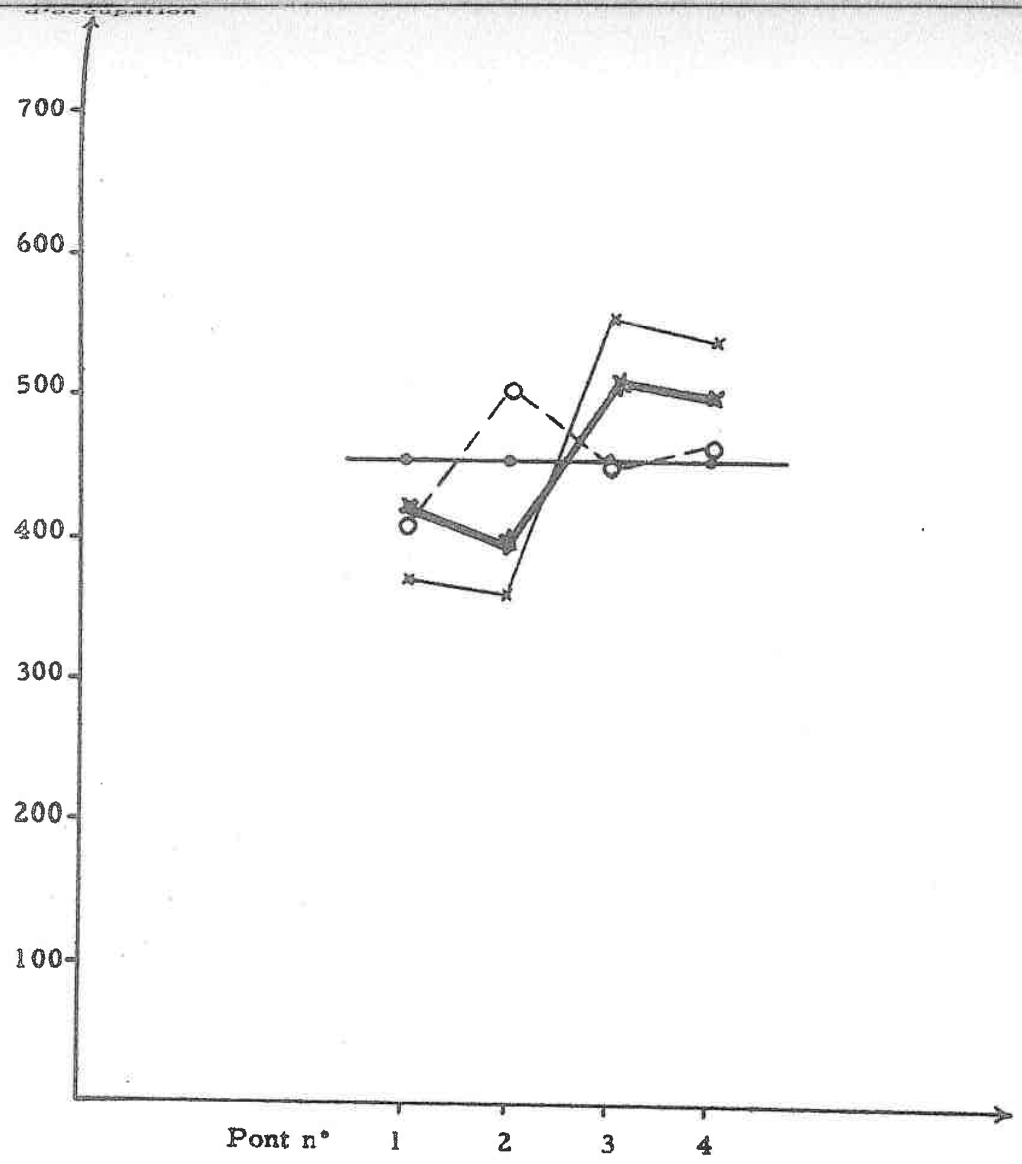
durée bandes	374	698	278	482	Monobandes	24
durée ponts	417	388	518	508	Bibandes	31
coeff. activité ponts	0,227	0,211	0,283	0,279	Tribandes	7

Commentaires sur ces résultats selon les deux objectifs.

α) "Répartition équitable"

Certains découpages donnent des résultats moins bons (n° 2 et 3), d'autres conservent sensiblement les mêmes valeurs (n° 1 et 4). Un seul découpage (le n° 5) donne des résultats légèrement supérieurs (valeurs des coefficients de travail : 0,21 à 0,28 au lieu de 0,20 à 0,30) à ceux fournis par la "Maquette".

(Voir sur le graphique (page suivante) les résultats obtenus pour le découpage n° 5).



Jeu 2

β) "Suppression des problèmes de frontières"

Pour cet objectif, les résultats vont en s'améliorant. Ils varient d'une répartition (Mono=26%, Bi=59%, Tri=14%) à une répartition (38%, 50%, 11%). En rapprochant ces résultats de ceux fournis par la "maquette" (20%, 65%, 15%) on se rend compte de l'apport de cette possibilité offerte à l'utilisateur.

Ainsi, en regroupant les deux objectifs, avec le découpage n° 5, on obtient des résultats supérieurs à ceux de la "Maquette", des résultats que l'on peut considérer comme bons.

Nous voyons ainsi, sur un exemple concret, quel peut être l'intérêt de l'utilisation de cette "Maquette" comme outil d'aide à la décision.

La seconde intervention de l'utilisateur sur cette "Maquette" sera de proposer un ordonnancement des tâches ainsi affectées.

B - L'ANALYSE FONCTIONNELLE

(I) Les seuls fichiers dont nous avons besoin sont les fichiers permanents.

- . TTACH table des tâches
- . TCOMP fichier des composantes

Pour la description de ces fichiers se reporter au chapitre II, page 21.

(II) Les traitements

Le programme définit

a) Pour chaque case de la halle :

- la valeur de la fonction DGT (durée d'occupation généralisée cumulée de cette case)

b) Pour chacune des quatre zones créées

- sa limite géographique supérieure
- la durée de l'activité s'y déroulant

c) Pour le découpage ainsi créé et pour tous les découpages proposés par l'utilisateur

c1) Pour chaque tâche

- la catégorie à laquelle elle appartient
- le pont auquel elle est affectée

c2) Des tableaux récapitulatifs

- tableaux regroupant les tâches de même catégories
- tableau comparatif
  - α) de l'activité se déroulant dans chacune des quatre bandes
  - β) de l'activité effective, réalisée par le pont correspondant
- tableau regroupant les coefficients d'activité effective de chaque pont (objectif 1)
- tableau regroupant le nombre de tâches de chaque catégorie (objectif 2).

DESCRIPTION MODULAIRE

NIVEAU I

I - TRAITEMENT

1 - Lecture du nombre de jeux d'essai que l'on veut traiter

2 - Pour chaque jeu d'essai

- 3 - Lecture de ses caractéristiques
- 4 - Lecture et traitement des fichiers correspondants
- 5 - Répartition de la halle en secteurs préférentiels
- 6 - Classification des tâches
- 7 - Affectation des tâches aux ponts
- 8 - Impression des résultats
- 9 - Dans le cas de l'intervention par l'utilisateur (proposition d'autres découpages)
  - 10 - Lecture du nombre de modifications proposées
  - 11 - Pour chacune des modifications
    - 12 - Classification des tâches
    - 13 - Affectation des tâches aux ponts
    - 14 - Impression des résultats

Déclaration : fichiers TTACH, TCOMP

NIVEAU II

I-3 - Lecture de ses caractéristiques

1 - Lecture du nombre de tâches à réaliser et du nombre de ponts chargés de ces réalisations

2 - Lecture des coefficients attribués à chaque bande

NIVEAU II

I-4 - Lecture et traitement des fichiers correspondants

1 - Lecture de la table TTACH

2 - Calcul des adresses des composantes originelles de chaque tâche

3 - Pour chaque tâche

- 4 - Lecture de ses composantes
- 5 - Calcul de sa durée
- 6 - Calcul de sa bande de travail
- 7 - Mise à jour des durées généralisées d'occupation des cases extrêmes

Déclaration : fichiers TTACH, TCOMP

NIVEAU II

I-5 - Répartition de la halle en secteurs préférentiels

1 - Calcul de la fonction somme cumulée des durées d'occupation totale (de chaque case)

2 - Calcul des durées d'occupation théorique de chacune des bandes (fonction des coefficients qui leur sont affectés)

3 - Calcul des bornes de chacune de ces bandes et des durées correspondantes

4 - Impression des résultats

NIVEAU II

I-6 - Classification des tâches

1 - Pour chaque tâche

| 2 - Comparaison de ses bornes extrèmes avec les bornes des bandes

| 3 - Attribution d'un numéro selon sa catégorie

4 - Impression des résultats

NIVEAU II

I-7 - Affectation des tâches aux ponts

1 - Pour chaque tâche

2 - Si elle est monobande

3 - Recherche de la bande correspondante

4 - Affectation au pont correspondant à la bande trouvée

5 - Si elle n'est pas monobande

6 - Recherche des bandes correspondantes

7 - Recherche parmi les ponts concernés de celui qui est le moins occupé à cet instant

8 - Affectation de la tâche à ce pont

NIVEAU II

I-8 - Impression des résultats

1 - Impression, pour chacune des tâches, du pont chargé de sa réalisation

2 - Impression des tableaux regroupant les tâches selon leur type (mono, bi, tri) et du nombre d'éléments de chaque type

3 - Impression d'un tableau comparatif comportant les durées d'activité de chaque bande et les durées d'activité effective des ponts correspondants après affectation.

4 - Impression des coefficients d'activité de chaque pont.

CHAPITRE V

Dans ce chapitre, nous allons expliciter la méthode PLC (méthode par les cartons ou encore par les contours), méthode qui réalise d'abord l'ordonnancement des tâches, l'affectation n'étant effectuée par l'utilisateur qu'en second lieu. Rappelons succintement les contraintes que l'on impose au système-pont.

α) Deux tâches ne peuvent avoir lieu au même endroit et en même temps.

β) Les tâches sont supposées libres (c'est à dire sans contrainte entre elles et réalisables par tout pont).

A partir des deux seuls fichiers permanents (le fichier des tâches et celui des composantes) représentatifs d'une activité de production, nous allons générer un ordonnancement de ces tâches. Nous verrons successivement :

A) L'exposé de la méthode et son exploitation

B) L'analyse fonctionnelle

A - L'EXPOSE DE LA METHODE

A-1 - Son but

Comme pour l'affectation, l'ordonnancement peut être réalisé de différentes façons selon l'objectif à atteindre. La méthode que nous présentons vise le double objectif suivant.

. Garder un niveau de sécurité satisfaisant

. Compte tenu de ce premier objectif, réaliser toutes les tâches dans le temps le plus court possible.

A-2 - Son principe

a) - Cette méthode P.L.C. n'est pas une méthode de recherche rigoureuse d'un optimum unique. L'unicité de cet optimum peut même ne pas exister.

- La méthode fournit à l'utilisateur une solution parmi un ensemble de ≪solutions bonnes≫.

- Elle fournit, en plus, les éléments qui permettront à l'utilisateur :

. Soit de trouver une autre solution parmi les ≪bonnes solutions≫

. Soit d'améliorer la solution proposée (ce qui est souvent possible).

Nous retrouvons, toujours, sous jacente, l'idée fondamentale d'aide à la décision fournie à l'utilisateur.

b) La méthode PLC consiste, à partir d'une représentation approchée (satisfaisant à la règle de sécurité que nous nous imposons : objectif n° 1) de chaque tâche, à ordonnancer ces tâches. Dans ce but on cherchera d'abord la case de la halle la plus occupée (sa durée d'occupation nous servira de limite inférieure pour la durée du processus que l'on génère : objectif n° 2).

On construira ensuite le sous-ensemble (dit de BASE) des tâches dont la réalisation nécessite l'occupation de cette case (la plus occupée). Les tâches ne faisant pas partie de ce sous-ensemble seront réparties en un sous-ensemble GAUCHE (ou G(1)) et en un sous-ensemble DROITE (ou D(1)) selon que leur position géographique respective se trouve à gauche ou à droite de cette case "centrale". Puis, par itérations successives, on ordonnera en priorité l'ensemble de Base tout en lui imposant une certaine régularité qui permettra d'ordonnancer au mieux les deux autres sous-ensembles.

A-3 - Son exposé détaillé

A-3-1) Approximation-sécurité de la tâche

Nous utilisons, à nouveau, la notion de "Bande de Travail" intervalle situé entre les deux cases extrêmes d'une tâche (voir chapitre IV, page

La règle de Sécurité que nous imposons est la suivante :

≪Lors de la réalisation d'une tâche : on interdit l'accès du secteur géographique relatif à cette tâche i à toute activité pendant la durée de la réalisation de cette tâche i≫

En d'autres termes si deux tâches i et j possèdent des bandes de travail qui se chevauchent, leurs réalisations ne pourront pas avoir lieu en même temps.

Voyons, sur un exemple, la nouvelle représentation graphique d'une tâche matérialisée par le contour extérieur, contour au sens large : c'est le rectangle R à l'intérieur duquel est inscrit (au sens mathématique), l'enveloppe E_{v_i} de la tâche i (représentation graphique adoptée au 2ème chapitre).

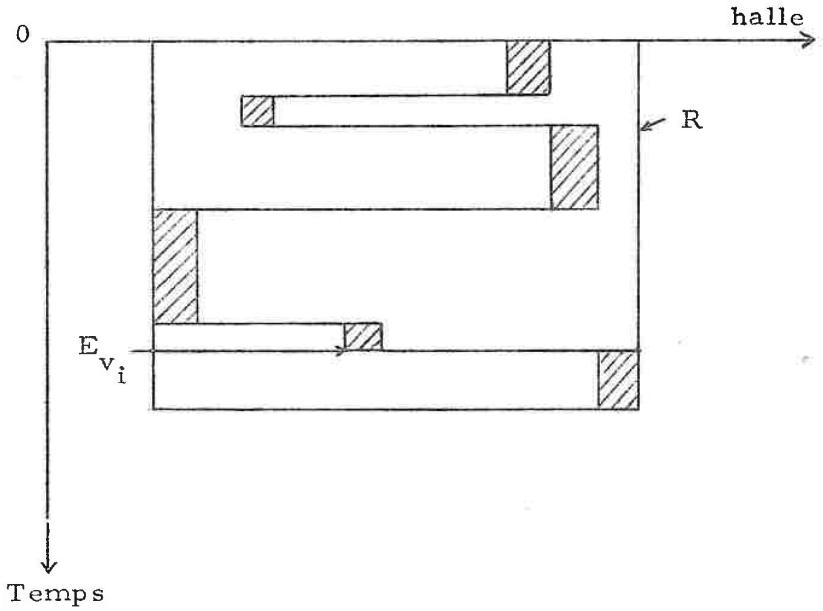


Figure n° V.1

. L'approximation de la tâche n'est pas très ≪fine≫ mais elle ne donne que plus d'importance à la règle de sécurité adoptée, elle permet de limiter les zones de rapprochement entre les ponts donc les risques de collisions et ceci est extrêmement important pour nous.

. Une des conséquences de cette approximation est le fait de considérer les cases (comprises entre les deux cases extrêmes) comme occupées toute la durée de la tâche et non plus seulement la durée effective du travail qui s'y déroule. Quand, dans la suite nous parlerons d'occupation, ce sera toujours de cette occupation ≪généralisée≫ que nous venons d'introduire et non pas d'occupation effective.

A-3-2) Enoncé du problème et des contraintes

On dispose d'un ensemble de tâches à réaliser: chacune de ces tâches est munie de ses caractéristiques. A partir de celles-ci on peut obtenir, par calcul, les deux éléments pertinents pour cette ≪approximation-sécurité≫ de la tâche, à savoir :

① Sa durée

② Ses deux positions géographiques extrêmes i-e les deux cases extrêmes - limites de la bande de travail - de cette tâche.

Ces deux éléments suffisent à définir clairement la tâche.

Chaque tâche peut être schématisée par un carton rectangulaire représentant le contour R de la tâche.

Un des cotés du carton est la donnée temporelle, l'autre étant la donnée spatiale.

ENONCE 1 :

Le problème d'ordonnancement est le suivant :

≪Déterminer pour chaque tâche son instant de déclenchement avec pour objectif de réaliser toutes les tâches dans le minimum de temps ≫

Compte tenu des CONTRAINTES 1 :

. R1 : La règle de Sécurité. Rappelons-là :

≪Lors de la réalisation d'une tâche i on interdit l'accès du secteur géographique relatif à cette tâche i à toute activité pendant toute la durée de la réalisation de cette tâche i ≫

. R2 : Le nombre de ponts disponibles.

Le nombre de ponts dont on dispose est de 4, donc, au plus, quatre tâches peuvent avoir lieu en même temps.

. R3 : La zone d'activité d'une tâche.

Chaque tâche occupe un secteur géographique bien défini et si son instant de déclenchement peut varier, il n'en est pas de même de la portion de la halle où se déroule l'activité relative à cette tâche.

En transposant l'énoncé 1 et les contraintes R1, R2, R3, en termes de ≪cartons≫, c'est à dire, en utilisant l'interprétation géométrique que l'on donne ainsi au problème, on obtient l'énoncé 2 et les contraintes R'_1, R'_2, R'_3.

ENONCE 2 :

Le problème de l'ordonnancement est le suivant :

≪Placer dans le système d'axes temps et halle tous les rectangles avec pour objectif de placer tous les rectangles en allant le moins loin possible sur l'axe des temps. Le placement se fait de la façon suivante :

- Compte tenu du fait que les axes ne sont pas de même nature, (l'un est spatial et l'autre temporel) les rectangles doivent toujours être placés de la même façon :

α) le côté temporel, parallèle à l'axe correspondant, donc vertical

β) le côté spatial, parallèle à l'axe correspondant, donc horizontal.

- Compte tenu des CONTRAINTES 2 :

. R'1 : La règle de sécurité :

Deux rectangles ne peuvent pas se chevaucher

. R'2 : Le nombre de ponts disponibles :

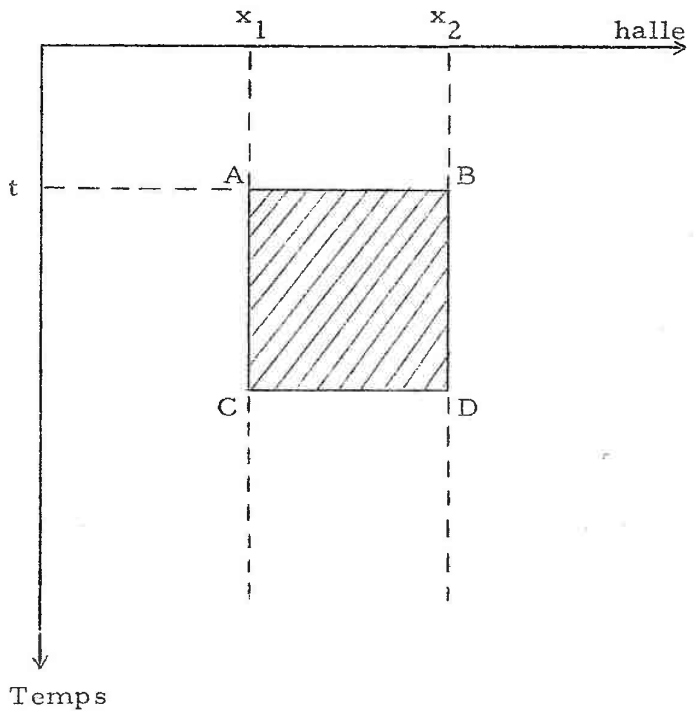
On ne peut pas mettre plus de quatre rectangles sur la même ligne horizontale.

. R'3 : La zone d'activité d'une tâche :

Sur le graphique V.2, nous représentons le rectangle dont les

sommets sont A, B, C, D. La position du rectangle dans le système d'axes est définie par les abscisses (x_1, x_2) des sommets A et B et par l'ordonnée t de ces deux sommets.

Les abscisses des sommets A et B sont invariantes, seule peut varier l'ordonnée t, c'est à dire que la position du rectangle peut varier mais seulement sur des ≪glissières≫ parallèles à l'axe temporel.



Graphique V.2

A-3-3) Genèse de la méthode manuelle et introduction des notions et règles nécessaires

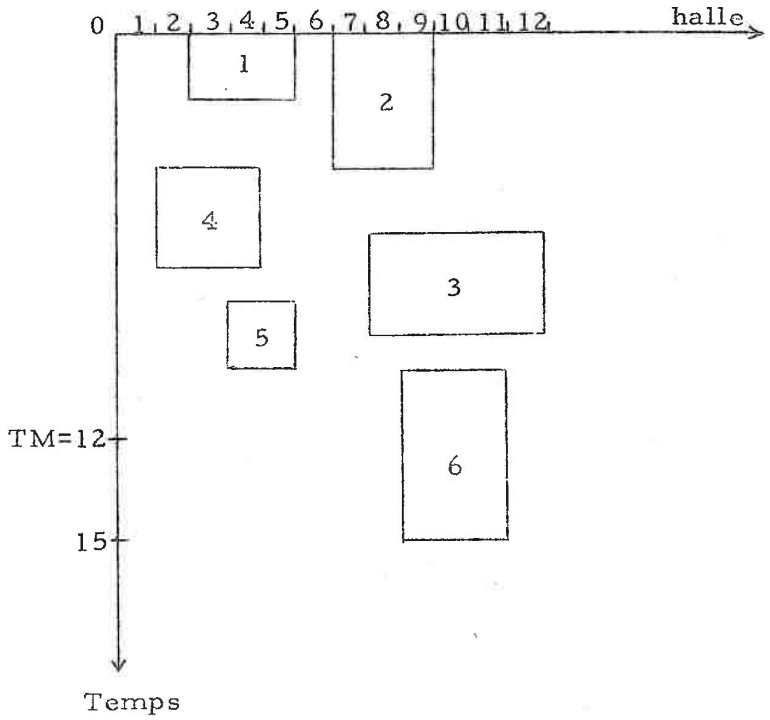
Nous explicitons maintenant, rapidement, la genèse de la méthode adoptée. Nous avons d'abord essayé de placer, manuellement, tous les rectangles.

C'est au cours de cette phase de manipulation, après divers essais, que nous nous sommes rendus compte que certaines règles se retrouvaient dans chaque essai et nous permettraient d'obtenir de bons résultats. La méthode que nous allons développer est donc une méthode heuristique.

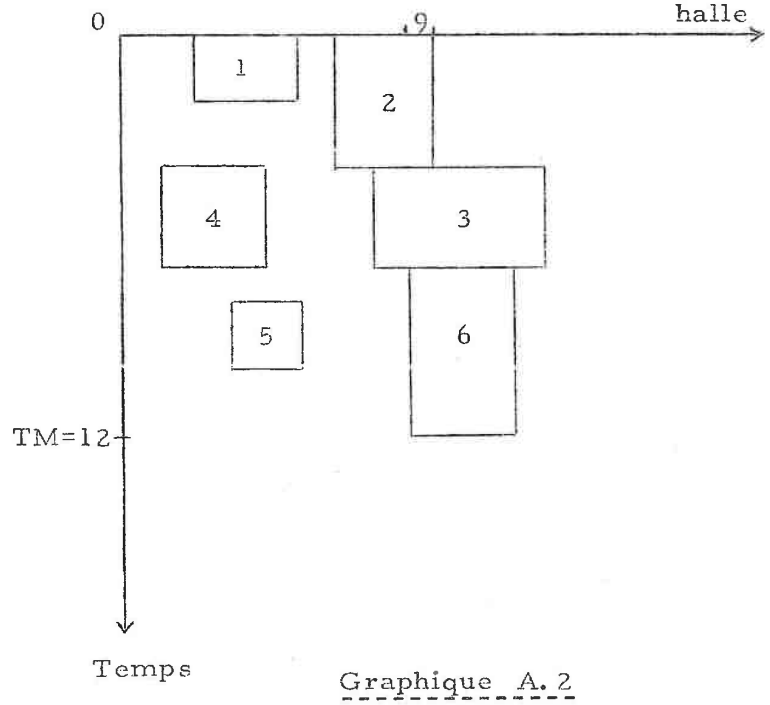
Nous allons introduire, sur des exemples simples, les notions et règles de bon sens qui nous ont facilité la pose des rectangles. Dans une phase ultérieure nous essaierons de formaliser ces règles afin de pouvoir les transposer sur ordinateur.

ⓐ Ensemble de base - Colonne de base et Ensemble critique

Sur le graphique A, 1 nous avons placé six rectangles.



Graphique A.1



Graphique A.2

Les rectangles ne se chevauchent pas, l'objectif n° 1 <<Conserver un certain niveau de sécurité au processus>> est satisfait. Mais il est évident que cette façon de placer les rectangles ne satisfait pas le second objectif <<Placer les rectangles dans le minimum de temps (axe vertical)>>. Sur le graphique A, 2 les six rectangles sont, cette fois, placés de telle sorte que l'objectif <<Réalisation des Tâcher dans le minimum de temps>> est atteint.

Il est même impossible de descendre en dessous de ce seuil (12) sans enfreindre la règle de sécurité. En effet, la case n° 9 (sur cet exemple) est tout le temps occupée et sa durée d'occupation fixe le seuil minimum du processus.

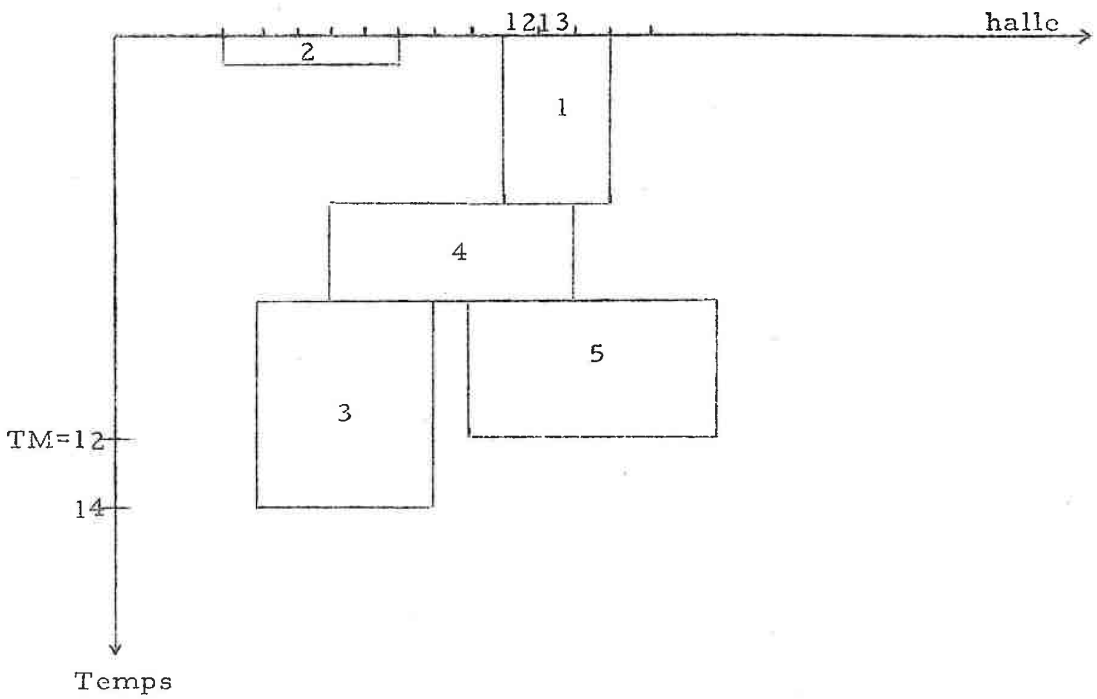
Nous mettons ainsi en évidence la case la plus occupée que nous appelons case de base.

Nous voyons également que nous sommes obligés de regrouper les rectangles recouvrant cette case de base. Ils formeront l'ensemble de base et une fois placés ils formeront la colonne de base. L'objectif à atteindre est de ne pas dépasser le temps d'occupation de cette case de base aussi placerons-nous en priorité les rectangles dits de base et nous les placerons les uns à la suite des autres sans laisser d'espace entre eux de telle sorte que la case de base soit toujours occupée. L'ensemble de base formera ce qu'on appellera un ensemble critique.

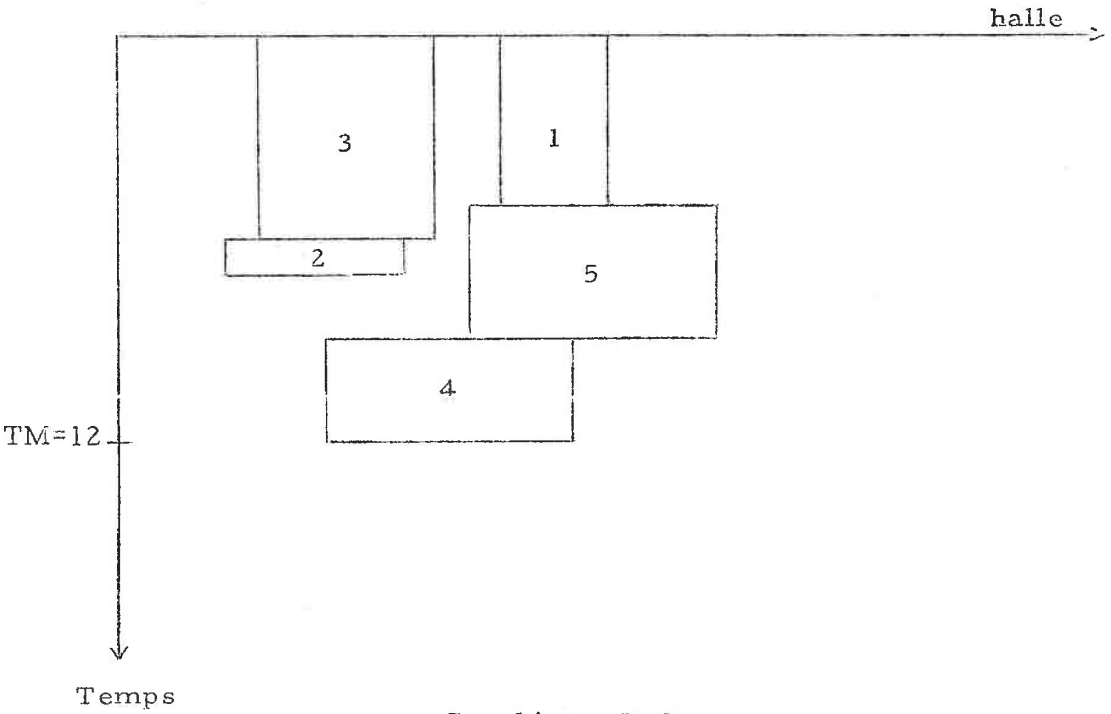
Définition : On appellera "Enveloppe" de la colonne de base le contour limite de cette colonne formé par les côtés verticaux des rectangles qui forment cette colonne. On parlera d'"Enveloppe gauche" si on ne s'intéresse qu'aux côtés verticaux gauches des rectangles (ces côtés correspondent aux cases extrêmes, bornes inférieures des bandes de travail). On parlera, de façon analogue, d'"Enveloppe droite".

Ⓑ Régularité de l'enveloppe de la colonne de base et sous-ensembles complémentaires.

Sur le graphique B, 1 nous avons placé cinq rectangles. Nous voyons, sur cet exemple, que la notion d'ensemble de base, d'ensemble critique n'est pas suffisante pour obtenir la pose de ces cinq rectangles dans le minimum de temps. Les rectangles 1, 4, 5 forment, en effet, un ensemble de base et sont placés sans perte de place mais le placement effectué nécessite 14 unités de temps alors que la case la plus occupée ne l'est que 12 unités de temps. Ceci est dû au fait que le rectangle 4 gène le placement du rectangle 3. Si on s'intéresse à l'enveloppe gauche de la colonne de base, la ≪protubérance≫ formée par le rectangle 4 empêche le placement du rectangle 3. Pour faciliter la pose de ce rectangle il faut que l'on regroupe les rectangles 1 et 5, puis placer seulement après le rectangle 4. On donne ainsi une certaine régularité à l'enveloppe gauche qui permet de placer les rectangles 2 et 3. Mais pour que l'objectif Temps minimum soit réalisé il faut que les rectangles 2 et 3 soient placés l'un à la suite de l'autre (avec au plus un retard de 2 unités de temps). Cela nous donne le graphique B, 2.



Graphique B.1



Graphique B.2

Indiquons tout de suite qu'en aucun cas, le côté gauche est privilégié par rapport au côté droit. L'étude aurait pu être faite à droite comme nous le verrons sur l'exemple suivant, on parlera alors de régularité de l'enveloppe droite de la colonne de base.

Si on revient à notre graphique B, 2. Les rectangles 1 et 5 forment un sous-ensemble A, les rectangles 3 et 2 forment un sous-ensemble B. Les éléments de chacun de ces sous-ensembles ne peuvent être posés que l'un derrière l'autre. Ces deux sous-ensembles peuvent être placés l'un à côté de l'autre. Nous dirons qu'ils sont complémentaires.

Ⓨ Régularité à droite. Rectangle ≪commun≫

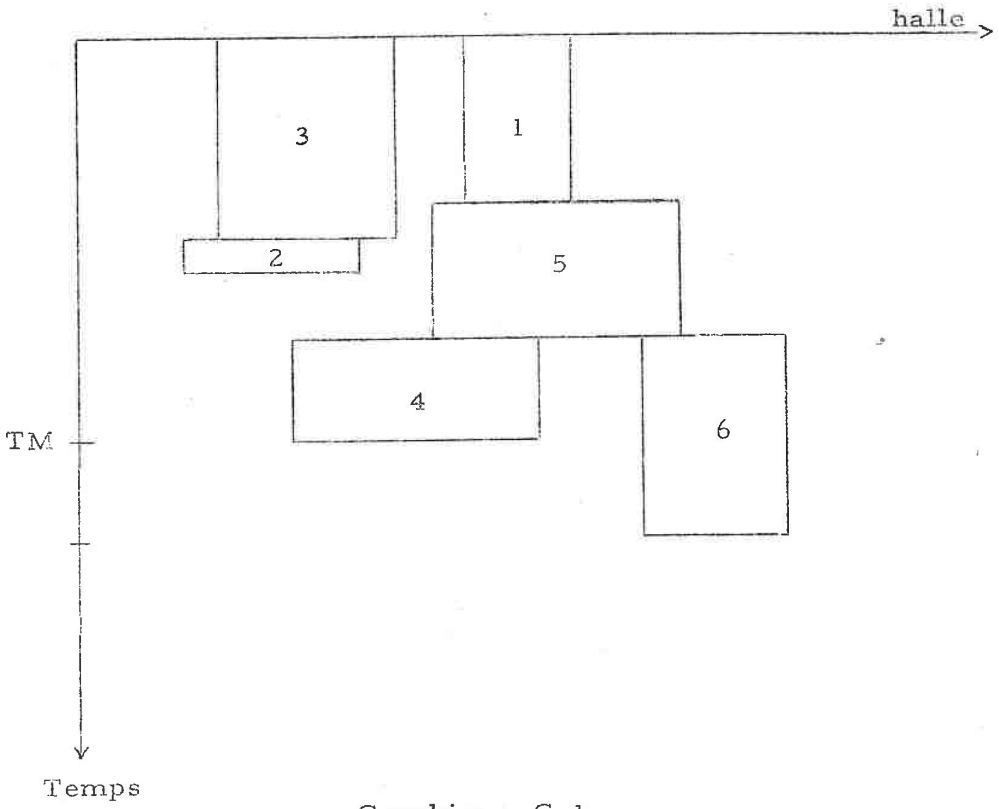
Sur le graphieque C, 1 nous avons placé six rectangles en tenant compte de l'existence de la colonne de base, de la régularité à gauche de sa courbe enveloppe. Nous voyons que, là encore, l'objectif Temps minimum n'est pas atteint.

Il manque, en effet, la régularité à droite, de l'enveloppe de la colonne de base. Le rectangle 5 forme ≪obstacle≫ à la pose du rectangle 6.

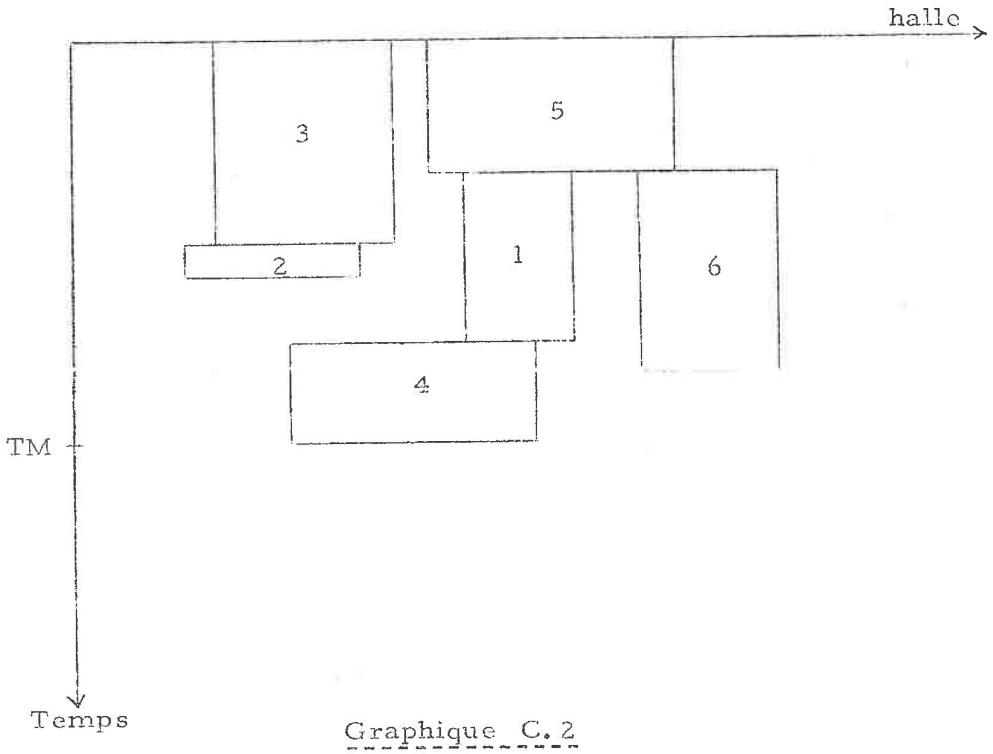
Pour obtenir cette régularité il faut permuter les rectangles 1 et 5 de telle sorte que les rectangles 1 et 4 placés l'un derrière l'autre ≪dégagent≫ suffisamment de place pour positionner le rectangle 6 (voir graphique C, 2).

Le rectangle 1 participe en même temps à la régularité à gauche et à la régularité à droite. Il est dit rectangle ≪commun≫. Afin de ne pas perdre cette propriété nous le placerons en fin du sous-ensemble intervenant dans la régularité d'un des côtés et en début du sous-ensemble intervenant dans la régularité de l'autre côté.

La notion de sous-ensemble complémentaire est valable, bien sûr, pour l'étude à droite.



Graphique C.1



Graphique C.2

⑧ Durée d'occupation de la case la plus occupée (Temps minimum = TM) et Minimum de temps pour le Processus

Nous avons cherché, sur chaque exemple présenté, à atteindre l'objectif visé, Temps minimum, et diverses règles nous ont permis de l'obtenir. Il existe cependant des cas où cet objectif ne peut pas être atteint (c'est le cas de l'exemple présenté sur le graphe D, 1), quelles que soient les règles employées. Mais l'application des règles que nous avons mis en évidence permet d'obtenir une bonne solution.

Montrons qu'il n'existe pas de processus qui puisse être réalisé en moins de temps :

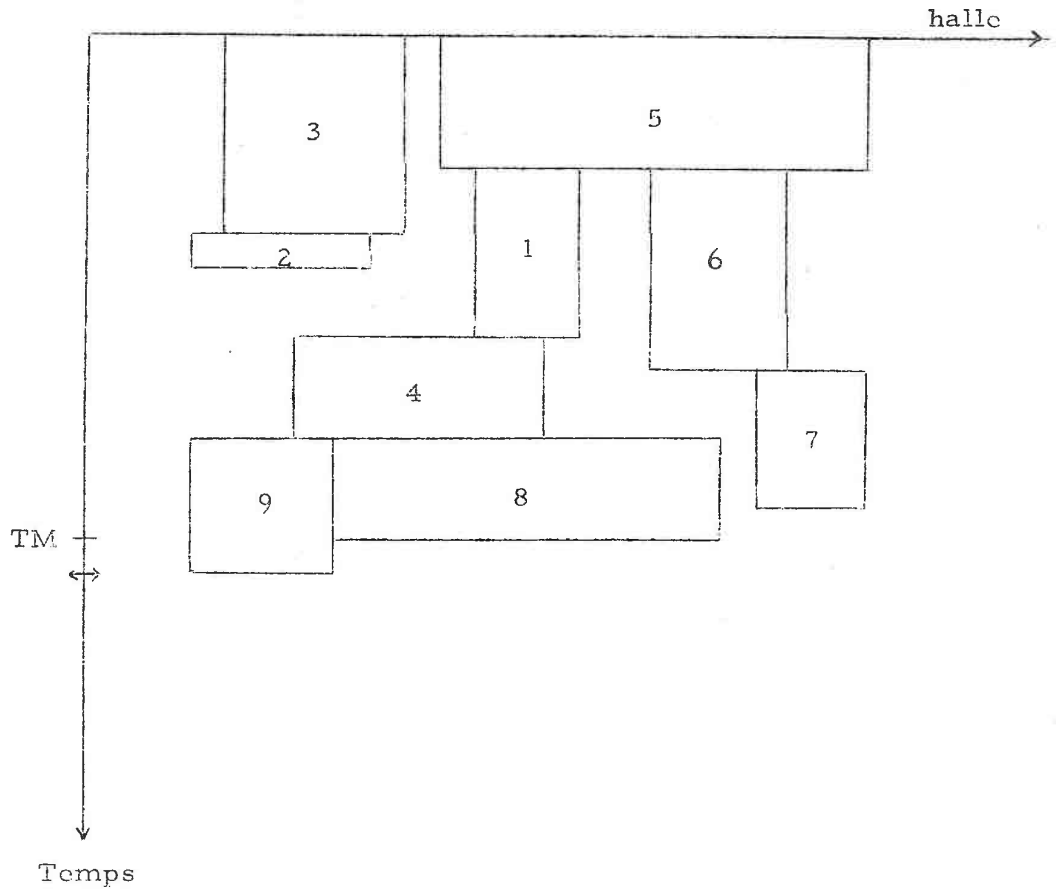
- Pour placer les rectangles 3 et 2, il faut que les rectangles 1 et 5 se suivent.

- Pour placer le rectangle 6, il faut que les rectangles 1 et 4 se suivent.

- On a ainsi l'ordre des rectangles (5, 1, 4 ou 4, 1, 5).

- Mais pour placer le rectangle 9, il faut que les rectangles 8 et 5 se suivent. Pour placer le rectangle 7, il faut que les rectangles 8 et 4 se suivent.

  Il est impossible au rectangle 8 de suivre à la fois 4 et 5.



Graphique D.1

Ayant introduit sur des exemples les diverses notions et règles qui nous ont servi dans nos manipulations, nous allons étudier la formalisation qui peut en être faite et la démarche que nous proposons pour traiter chaque exemple présenté.

Cette démarche est susceptible d'amélioration. Il est en effet possible qu'à la suite de manipulations sur de nombreux exemples de type analogue, apparaissent des règles de placement qui s'adaptent mieux au type de problème traité. Des changements devront, alors, être faits.

Dans ces manipulations, nous disposons d'un ≪outil≫ de travail extrêmement utile: La visualisation du processus, facilitée par la matérialisation des tâches. En effet la vue d'un processus représenté graphiquement donne l'idée de certaines modifications et de règles qui pourront être formalisées. La visualisation joue donc un rôle extrêmement important dans l'aide à la conception.

Dans le paragraphe suivant, nous allons formaliser les règles déjà énoncées afin de transposer la méthode d'ordonnancement sur ordinateur.

Nous obtiendrons, par ce programme, un processus. Afin de conserver l'outil très utile qu'est la visualisation, nous donnerons, par programme également, une représentation graphique du processus généré.

Nous demanderons, en plus, au programme de nous fournir certains renseignements qui permettront à l'utilisateur de mieux connaître la structure de l'ensemble des tâches à ordonnancer.

La représentation graphique complétée par ces renseignements forment les éléments de l'Aide à la Décision.

Grâce à ces documents l'Utilisateur pourra voir si le processus généré est susceptible d'amélioration ou pas. Son intervention peut être alors de deux types :

. Soit : il interviendra en modifiant certains paramètres et en testant sur le programme les nouvelles valeurs de ces paramètres.

. Soit : Il pourra concevoir de nouvelles règles qu'il devra formaliser et introduire dans le programme pour améliorer l'automatisation des traitements.

Il y a donc ≪dialogue≫ entre l'utilisateur et son programme. L'efficacité du programme dépend du choix des paramètres et des règles que l'utilisateur aura pu formaliser et introduire dans ce programme.

A-3-4) Formalisation des règles adoptées

A-3-4-1) Recherche de la case la plus occupée (dite case de base)

α) Calcul de la durée d'occupation de chaque case.

Pour chaque tâche i à exécuter, la durée d'occupation de chaque case située dans la bande de travail de cette tâche sera augmentée de la durée de cette tâche.

Soit Zo(j) la durée d'occupation de la case j

Soit TAMP(i) la durée de la tâche i (i variant de 1 à iMAX)

NIN(2i) et NAX(2i) les bornes de sa bande de travail

Pour toute case n (variant de 1 à NT nombre maximum de case) nous obtiendrons sa durée d'occupation grâce à la formule :

$$Zo(n) = \sum_{i=1}^{iMAX} TAMP(i) \qquad \text{avec} \quad NIN(2i) \leq n \leq NAX(2i)$$

β) La case dont la durée d'occupation est maximum est appelée case de base (cb).

$$\text{i-e} \qquad Zo(cb) = \max_{1 \leq n \leq NT} (Zo(n))$$

N. B. S'il existe plusieurs cases de durée d'occupation maximum, nous choisirons comme case de base la première case ≪maximum≫ obtenue.

Zo(cb) est la limite inférieure de temps que nous cherchons à atteindre.

A-3-4-2) Répartition de l'ensemble E des tâches en trois sous-ensembles

α) Construction de l'ensemble TACCH dit ensemble de base

Pour que l'on puisse effectuer tout le travail imposé dans le minimum de temps Zo(cb) il faut (mais ce n'est pas toujours suffisant) que la case de base "cb" soit occupée sans discontinuité, donc qu'il n'y ait pas de temps mort entre les réalisations des tâches qui utilisent cette case de base. Ce sont ces tâches qui sont placées en priorité. Tout retard imposé au déclenchement d'une de ces tâches décalera d'autant la date de fin de réalisation du travail. Soit TACCH le sous-ensemble de E (ensemble des tâches à réaliser) formé de ces tâches dont la bande de travail recouvre la case de base.

$$TACCH = \{ i \in E \mid NIN(2i) \leq cb \leq NAX(2i) \}$$

β) Construction des sous-ensembles GAUCHE et DROIT.

Le sous-ensemble des tâches restantes peut être réparti en deux sous-ensembles selon la position de la bande de travail de ces tâches par rapport à la case de base.

. Soit GAUCHE (ou G(1)) le sous-ensemble de E formé des tâches dont la borne supérieure de la bande de travail est strictement inférieure à la case de base (i-e dont la bande de travail est à gauche de la case de base).

. Soit DROIT (ou D(1)) le sous-ensemble de E formé des tâches dont la borne inférieure de la bande de travail est strictement supérieur à la

case de base (i-e dont la bande de travail est à droite de la case de base).

En formalisant :

$$GAUCHE = \{ i \in E \ / \ MAX(2i) < cb \}$$

$$DROIT = \{ i \in E \ / \ cb < MIN(2i) \}$$

$$E = TACCH \cup GAUCHE \cup DROIT$$

A-3-4-3) Ordonnancement(s) interdépendant(s) des trois sous-ensembles (gauche, de base, et droit)

La méthode des ordonnancements proposés repose sur un principe de "Partition itérative de ces trois ensembles en sous-ensembles complémentaires" puis de "Construction progressive d'ordonnancements réalisables localement optimisés de ces sous-ensembles".

La méthode procède donc en plusieurs étapes et utilise les concepts de "Complémentarité" (des bandes de travail des tâches) et d'"Ensemble critique".

A - DEFINITION DES CONCEPTS

A-1 - La Complémentarité

On dira que deux ensembles de tâches A et B sont complémentaires si :

α) Quelque soit x appartenant à A et quelque soit y appartenant à B, les bandes de travail de x et de y ne se chevauchent pas (c'est à dire que les deux tâches x et y peuvent avoir lieu en même temps).

β) Quelque soit x_i et x_j appartenant tous les deux à l'un des deux ensembles A ou B, les bandes de travail de x_i et de x_j se chevauchent (c'est à dire que les deux tâches x_i et x_j ne peuvent pas avoir lieu en même temps).

On peut formaliser ainsi :

α) $\forall i \in A$ et $\forall j \in B$ nous avons

- soit $NAX(2i) < NIN(2j)$
- soit $NAX(2j) < NIN(2i)$

β) Nous avons

$$\begin{cases} \text{- soit} & NIN(2i_2) \leq NAX(2i_1) < NAX(2i_2) \\ \text{- soit} & NIN(2i_1) \leq NAX(2i_2) \leq NAX(2i_1) \end{cases}$$

pour i_1 et i_2 quelconques appartenant à A

ou

pour i_1 et i_2 quelconques appartenant à B

A-2 - Ensemble critique

On définit l'ensemble critique comme étant un ensemble de tâches tel que lorsqu'on le munit d'un ordre total chronologique, le chemin hamiltonien ainsi créé soit chemin critique.

B - PARTITION ITERATIVE DES TROIS SOUS-ENSEMBLES TACCH, GAUCHE, DROIT

Au cours de chaque itération la méthode comprend les phases suivantes :

- obtention de deux couples de sous-ensembles complémentaires, satisfaisant à certaines propriétés
- création de relations inter-tâches suivant des règles que nous nous sommes fixées
- mise à jour des sous-ensembles pour passage à l'itération suivante.

Avant de passer à l'étude détaillée de l'itération initiale nous allons donner quelques définitions.

Définition 1 : On définit une fonction MAMABA qui à un ensemble X fait correspondre le MAximum des cases MAximum (i-e des bornes supérieures) des BAndes de travail des tâches appartenant à cet ensemble X.

Définition 2 : On définit, également, une fonction MIMIBA qui à un ensemble X fait correspondre le MInimum des cases MInimum (i-e des bornes inférieures) des BAndes de travail des tâches appartenant à cet ensemble X.

ITERATION INITIALE

a Obtention de couples de sous-ensembles complémentaires.

a-1) Etude à gauche

a-1-1) On construit, à partir de l'ensemble G(1) des tâches restantes à gauche, le sous-ensemble (CHOIRG(1)) formé par les tâches dont la borne supérieure de la bande de travail est la plus grande de cet ensemble G(1) (on dit qu'elles possèdent la propriété $P_{1,g}$). En d'autres termes, ce sont les tâches situées le plus près (à gauche) de la case de base.

. En formalisant on dira que le sous-ensemble CHOIRG(1) vérifie la propriété $P_{1,g}$ si

$$\forall i \in CHOIRG(1) \qquad NAX(2i) = MAMABA(G(1))$$

. Soit c cette borne supérieure MAMABA(G(1)). Nous avons

$$c < cb$$

a-1-2) On construit, à partir de l'ensemble TACCH de base, le sous-ensemble CHOITG(1) formé des tâches dont la borne inférieure de la bande de travail est telle qu'aucune de ces tâches ne chevauche les tâches de l'ensemble CHOIRG(1) construit précédemment. En d'autres termes, elles peuvent avoir lieu en même temps que celles du sous-ensemble précédent.

. En formalisant, on dira que le sous-ensemble CHOITG(1) vérifie la propriété $P_{1,Bg}$ si :

$$\forall i \in CHOITG(1) \qquad NIN(2i) > MAMABA(G(1))$$

a-1-3) On peut démontrer le lemme suivant
≪La somme dtg(1) des durées des tâches du sous-ensemble CHOITG(1) est supérieure ou égale à la somme drg(1) des durées des tâches du sous-ensemble CHOIRG(1)≫

En raisonnant par l'absurde :

- soit Zo(cb) la durée d'occupation de la case de base cb

- la durée Zo(c) d'occupation de la case c sera due

  . d'une part aux tâches de base ne faisant pas partie de CHOITG(1) donc pour une durée totale = Zo(cb)-dtg(1)

  . d'autre part aux tâches faisant partie de CHOIRG(1) donc pour une durée totale de drg(1)

D'où Zo(c) = Zo(cb) + drg(1) - dtg(1)

- si, par l'absurde, drg(1) est supérieur à dtg(1) nous aurions

Zo(c) > Zo(cb)

Ce qui est contraire à l'hypothèse selon laquelle cb est la case dont la durée d'occupation est la plus élevée.

a-1-4) Conséquence

Le fait de former ainsi le sous-ensemble CHOITG(1) permet de dégager un ≪emplacement≫ suffisant pour placer les tâches du sous-ensemble CHOIRG(1).

Pour pouvoir utiliser au mieux cet ≪emplacement≫ les réalisations de ces tâches seront regroupées, c'est à dire qu'elles seront déclenchées les unes à la suite des autres, sans interruption. Cependant l'ordre dans lequel ces tâches seront déclenchées n'est pas encore fixé. Le fait que CHOITG(1) soit sous-ensemble critique (comme sous-ensemble d'un ensemble critique) impose le déclenchement, sans temps mort, des tâches de CHOITG(1). Afin de perdre le moins de place possible, nous prenons comme règle de traitement la règle suivante (RT): Considérer et Traiter CHOIRG(1) comme sous-ensemble critique. En réalité la réalisation des tâches le composant peut accepter un retard cumulé au plus égal à dtg(1) - drg(1).

a-1-5) Création des relations inter-tâches

α) Pour indiquer que CHOITG(1) et CHOIRG(1) sont considérés comme sous-ensembles critiques, on crée entre tout couple de tâches appartenant au même sous-ensemble (que ce soit CHOITG(1) ou CHOIRG(1)) une relation indiquant, grâce à leurs appartenances communes au même sous-ensemble, que leurs réalisations pourront être successives. Cette relation sera appelée relation de ≪succession possible≫.

β) Pour indiquer que les ensembles CHOITG(1) et CHOIRG(1) sont complémentaires on crée entre tout couple de tâches (l'une d'elles appartenant à CHOIRG(1), l'autre appartenant à CHOITG(1)) une relation indiquant que leurs réalisations pourront être simultanées. Cette relation sera appelée relation de ≪voisinage possible≫

Remarque : Actuellement ces deux sous-ensembles ne sont pas ordonnés. Nous n'avons pas établi d'ordre chronologique pour chacun de ces sous-ensembles.

- Le nombre de façons possibles d'ordonner les tâches de ces sous-ensembles est respectivement de

(card CHOITG(1))! et (card CHOIRG(1))!

- Le choix de la solution proposée se fera suivant une règle que nous énoncerons plus loin.

Les autres solutions, non proposées, contribuent à former l'ensemble des ≪bonnes solutions≫ dont nous avons déjà parlé.

a-2) Etude à droite

On effectue le même travail de recherche de sous-ensembles complémentaires que lors de l'étude à gauche. On retrouve donc la même démarche.

a-2-1) Construction, à partir de l'ensemble D(1) des tâches restantes à droite, du sous-ensemble CHOIRD(1) formé des tâches dont la borne inférieure de la bande de travail est la plus petite de cet ensemble D(1). On dit qu'elles vérifient la propriété $P_{l,d}$. En d'autres termes ce sont les tâches situées le plus près, à droite, de la case de base.

. On aura : le sous-ensemble CHOIRD(1) vérifie $P_{l,d}$

si $\forall i \in$ CHOIRD(1) NIN(2i) = MIMIBA(D(1))

. Soit c' cette borne inférieure MIMIBA(D(1)). On a $c' > cb$

a-2-2) Construction, à partir de l'ensemble TACCH de base, du sous-ensemble CHOITD(1) formé des tâches dont la borne supérieure de la bande de travail est telle qu'aucune de ces tâches ne chevauche les tâches de l'ensemble CHOIRD(1) construit précédemment. Ces tâches peuvent avoir lieu en même temps que celles du sous-ensemble le précédent.

. En formalisant, on dit que le sous-ensemble CHOITD(1) vérifie la propriété $P_{1,Bd}$ si

$$\forall\, i \in \text{CHOITD(1)} \qquad \text{NAX(2i)} < \text{MIMIBA(D(1))}$$

a-2-3) On peut démontrer, en raisonnant par l'absurde comme plus haut, le lemme suivant :

≪La somme dtd(1) des durées des tâches du sous-ensemble CHOITD(1) est supérieure ou égale à la somme drd(1) des durées des tâches du sous-ensemble CHOIRD(1)≫

a-2-4) Conséquence

Contrairement à l'étude à gauche (faite en premier) le fait de former le sous-ensemble CHOITD(1) ne permet de dégager ≪l'emplacement≫ suffisant pour placer les tâches du sous-ensemble CHOIRD(1) qu'à la condition de respecter la règle suivante : (RC)

En effet, il peut arriver que certaines tâches (de base) soient communes aux deux sous-ensembles CHOITD(1) et CHOITG(1) que l'on vient de construire (exemple. une tâche n'occupant que la case de base est nécessairement commune aux deux sous-ensembles).

Soit ICOMMUN (1) le sous-ensemble formé de ces tâches communes.

$$ICOMMUN(1) = CHOITG(1) \cap CHOITD(1)$$

Règle RC : Il faut que les tâches formant ce sous-ensemble ICOMMUN(1) se trouvent regroupées et placées à la fin de l'ensemble CHOITG(1) et au début de l'ensemble CHOITD(1). Ce sous-ensemble ICOMMUN(1) est lui aussi sous-ensemble critique.

Nous sommes obligés d'appliquer la règle RC si l'on veut que les tâches "Communes" (i-e appartenant à ICOMMUN(1)) interviennent à la fois dans la régularité à gauche et dans la régularité à droite de l'enveloppe de l'ensemble de base.

Compte tenu de cette règle RC, nous nous retrouvons dans le même cadre que celui de l'étude à gauche, avec des résultats similaires :

- CHOITD(1) est sous-ensemble critique comme sous-ensemble d'un ensemble critique

- Suivant la règle de traitement RT adoptée pour l'étude à gauche, nous considérons et traitons CHOIRD(1) comme sous-ensemble critique (ici, le retard cumulé autorisé dans le déclenchement des tâches de ce sous-ensemble est égal au plus à dtd(1) - drd(1)).

a-2-5) Création de relations inter-tâches

α) Pour indiquer que CHOITD(1) et CHOIRD(1) sont considérés comme sous-ensembles critiques, on crée entre tout couple de tâches appartenant au même sous-ensemble (CHOITD(1) ou CHOIRD(1)) la relation dite de ≪ succession possible ≫

β) Pour indiquer que les ensembles CHOITD(1) et CHOIRD(1) sont complémentaires, on crée entre tout couple de tâches (l'une appartenant à CHOITD(1) et l'autre appartenant à CHOIRD(1)) la relation dite de ≪ voisinage possible ≫

Remarque : La remarque faite en a-1-5 reste, elle aussi, valable :

A ce niveau-ci de la méthode nous n'avons pas encore établi d'ordre chronologique sur les deux couples de sous-ensembles complémentaires.

Cependant compte tenu des deux règles énoncées, celle de traitement RT et celle RC relatif à l'ensemble des tâches communes, le nombre de façons possibles d'ordonner les tâches de base choisies au cours de cette première itération est le suivant :

$$(\text{Card}(\text{CHOITG}(1)/\text{ICOMMUN}(1)))! \times (\text{Card}(\text{ICOMMUN}(1)))! \times (\text{Card}(\text{CHOITD}(1)/\text{ICOMMUN}(1)))!$$

N.B. Si un ensemble est vide, son cardinal est nul, nous prendrons par convention $(\text{Card}(o))! = 1$.

a-3) Contribution à l'aide à la décision

Tout rectangle ne peut être placé dans le système d'axes temps et espace qu'après avoir été rangé dans un des sous-ensembles que nous construisons par programme (que ce soit CHOITG(1), CHOIRG(1), CHOITD(1), CHOIRD(1)).

Nous avons vu que le rangement d'un rectangle dans un des sous-ensembles était dû au fait que l'un de ses deux côtés verticaux ≪participait≫ à la régularité de l' "Enveloppe" (gauche ou droite).

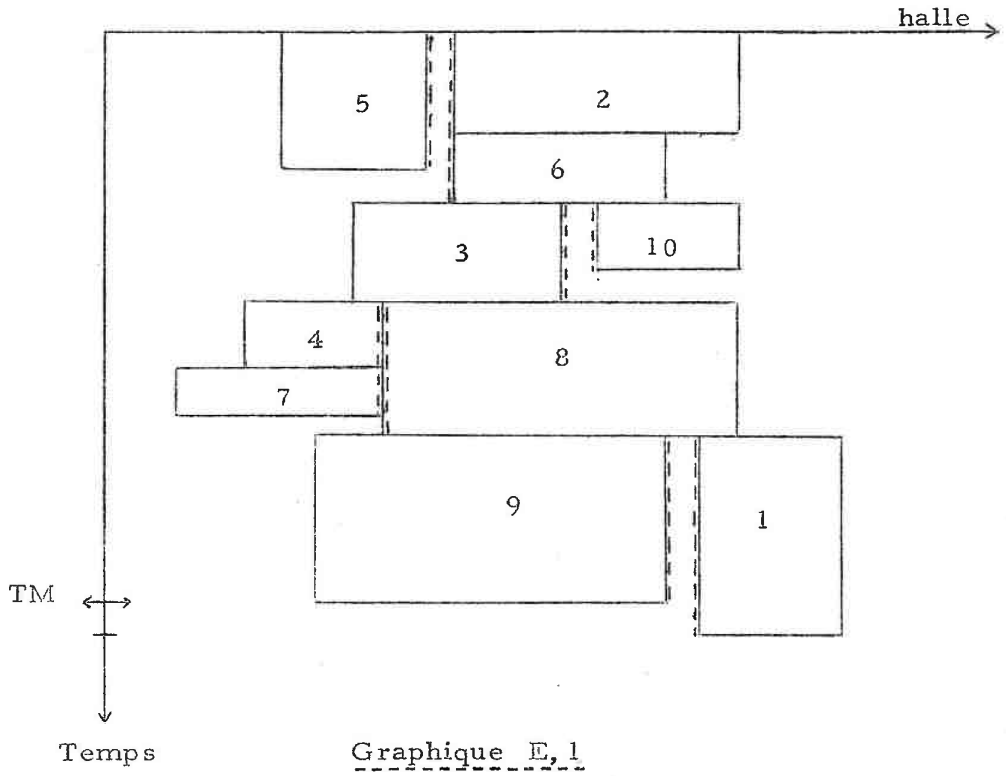
Mais il nous a semblé intéressant, à titre indicatif, de faire apparaître dans les documents d'aide à la décision d'autres possibilités de rangement de ce rectangle en tenant compte de l'autre côté vertical, de sa ≪participation≫ à la régularité de l' "Enveloppe (droite ou gauche).

Voyons sur les graphiques E, 1 et E, 2 quelle peut être l'exploitation de ces documents d'aide à la décision.

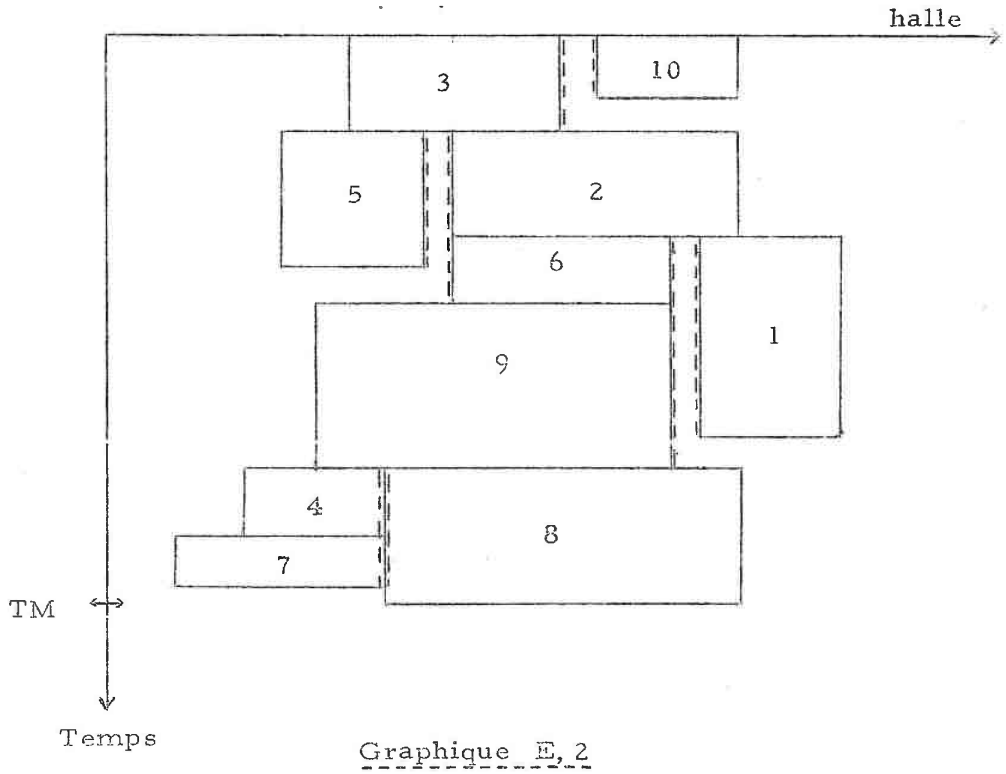
Sur le graphique E, 1 on ne tient compte que de l'utilisation d'un seul côté vertical pour chaque rectangle. Nous avons tracé en pointillé le côté vertical ≪participant≫ à la régularité de l'Enveloppe.

Ainsi le rectangle n° 6 intervient par son côté vertical gauche (en pointillé) dans la régularité de l' "Enveloppe" gauche : il doit en effet être regroupé avec le rectangle n° 2 pour faciliter la pose du rectangle n° 5.

Le rectangle n° 9 intervient par son côté vertical droit pour faciliter la pose du rectangle n° 1 mais il ne suffit pas à ≪acceuillir≫ seul le rectangle 1 (le processus représenté sur le graphique E, 1 dure 1 Δ t de plus que la durée d'occupation TM de la case de base).



Graphique E, 1



Graphique E, 2

Mais si on sait que les rectangles n° 6 et n° 9 ont même borne supérieure (côté droit) on va tenter de les rapprocher pour augmenter la régularité de l' "Enveloppe" droite. Nous voyons sur le graphique E, 2, que, pour cet exemple, le rapprochement est possible mais en bouleversant le processus établi.

En exploitant ainsi les deux côtés verticaux du rectangle n° 6, on permet un positionnement de tous les rectangles dans le minimum de temps TM. Le rectangle n° 1 peut, en effet, être placé sans problème à côté des rectangles 6 et 9.

La solution, ainsi obtenue, est meilleure et même ne peut pas être améliorée.

Mais si cette modification est relativement aisée à faire manuellement, sa formalisation et sa mise en oeuvre par programme est assez complexe. C'est pourquoi nous laissons à l'utilisateur le soin de la réaliser lui-même, le programme lui fournissant les éléments utiles à cette réalisation.

Pour permettre ce double rangement, lors de l'étude sur l'ensemble de base TACCH et afin de différencier l'étude à gauche de celle à droite, nous dirons que l'étude est faite respectivement sur TABG(1) et sur TABD(1)

Nous avons TACCH = TABG(1) = TABD(1)

Après la construction de ces couples de sous-ensembles complémentaires et la création de relations inter-tâches que nous avons explicitées lors de cette itération initiale, nous allons mettre à jour les ensembles de tâches sur lesquels nous travaillons. Nous pourrons alors passer aux itérations suivantes. La démarche de chacune des itérations sera la même que celle de l'itération initiale. Le travail étant terminé lorsqu'il n'y a plus de tâches à placer.

a-4 - Mise à jour des ensembles de tâches

a-4-1) Les ensembles des tâches restantes à gauche et à droite sont diminuées des tâches choisies lors de cette itération

$$G(2) = G(1) \;/\; CHOIRG(1)$$

$$D(2) = D(1) \;/\; CHOIRD(1)$$

a-4-2)

α) L'ensemble des tâches qui pourront intervenir (par la borne inférieure de leur bande de travail) dans la régularité à gauche contiendra

1) l'ensemble des tâches de base n'étant pas encore intervenues par leur borne inférieure

2) l'ensemble des tâches choisies parmi les restantes à gauche (i-e celles dont la borne supérieure de leur bande de travail a déjà provoqué une première apparition).

$$TABG(2) = (TABG(1) \;/\; CHOITG(1)) \cup CHOIRG(1)$$

β) De même pour l'ensemble droit

$$TABD(2) = (TABD(1) \;/\; CHOITD(1)) \cup CHOIRD(1)$$

En autant d'itérations qu'il faudra pour placer toutes les tâches restantes, nous renouvellerons la manipulation dans sa forme la plus générale. L'étude de l'itération i est refaite en détail : l'idée générale de construction de sous-ensembles complémentaires et la démarche restent celles de l'itération initiale, mais les énoncés des lemmes, la création des relations inter-tâches sont modifiés légèrement.

ETUDE DE L'ITERATION n° i

a) Obtention de couples de sous-ensembles complémentaires

a-1) Etude à gauche

a-1-1) Construction, à partir de G(i), du sous-ensemble CHOIRG(i) formé des tâches vérifiant la propriété $P_{i,g}$.

On dit que la tâche j vérifie la propriété $P_{i,g}$

si NAX(2j) = MAMABA(G(i))

a-1-2) Construction, à partir de TABG(i), du sous-ensemble CHOITG(i) formé des tâches vérifiant la propriété $P_{i,Bg}$.

On dit que la tâche j vérifié la propriété $P_{i,Bg}$

si NIN(2j) > MAMABA(G(i))

a-1-3) Enoncé du lemme

L'énoncé du lemme n'est pas identique à celui de l'itération initiale. En effet, lors des itérations précédentes, on a pu dégager plus d'"emplacement" qu'il n'était nécessaire. Soit RESG(i) le surplus de place inemployé à gauche.

On peut énoncer le lemme dans sa forme générale.

≪La somme (dtg(i)) des durées des tâches du sous-ensemble CHOITG(i) augmentée du surplus (RESG(i)) déjà existant est supérieure à la somme (drg(i)) des durées des tâches du sous-ensemble CHOIRG(i)≫

La démonstration reste du même type que celle que nous avons déjà présentée.

a-1-4) Conséquence

Compte tenu de l'importance et de la structure de RESG(i), nous ne pouvons pas, contrairement à l'itération initiale, assurer que le fait de former CHOITG(i) permet de dégager, chaque fois, un ≪emplacement≫ suffisant pour placer les tâches du sous-ensemble CHOIRG(i). Nous plaçons le plus de tâches possible appartenant à CHOIRG(i).

Les deux règles de traitement sont donc :

1) Traiter CHOITG(i) en sous-ensemble critique (il l'est comme sous-ensemble d'ensemble critique)

2) Considérer et traiter, dans la mesure du possible, CHOIRG(i) comme un sous-ensemble critique.

a-1-5) Création des relations inter-tâches

α) On crée pour tout couple de tâches appartenant au même sous-ensemble (que ce soit CHOITG(i) ou CHOIRG(i)) une relation indiquant leur appartenance commune à ce sous-ensemble. C'est la relation de ≪succession possible≫.

Cependant, si c'est la seconde apparition pour un couple déterminée, c'est à dire qu'il existe déjà une relation de succession possible de rang inférieur : on ne crée pas de nouvelle relation.

β) On crée entre tout couple de tâches (l'une appartenant à CHOIRG(i) et l'autre à CHOITG(i)) une relation de ≪voisinage possible≫

a-2 - Etude à droite

a-2-1) Construction, à partir de D(i), du sous-ensemble CHOIRD(i) formé des tâches vérifiant la propriété $P_{i,d}$.

On dit que la tâche j vérifie la propriété $P_{i,d}$

si NIN(2j) = MIMIBA(D(i))

a-2-2) Construction, à partir de TABD(i), du sous-ensemble CHOITD(i) formé des tâches vérifiant la propriété $P_{i,Bd}$.

On dit que la tâche j vérifie la propriété $P_{i,Bd}$

si NAX(2j) < MIMIBA(D(i))

a-2-3) Enoncé du lemme

La somme (dtd(i)) des durées des tâches du sous-ensemble CHOITD(i) augmentée du surplus (RESD(i)) déjà existant est supérieure à la somme (drd(i)) des durées des tâches du sous-ensemble CHOIRD(i).

a-2-4) Conséquence

Comme pour l'étude à gauche, la possibilité de considérer et de traiter CHOIRD(i) comme un sous-ensemble critique est subordonnée à la structure et à l'importance du surplus de ≪place≫ déjà dégagé mais inemployé.

Cependant nous maintenons la règle de traitement RC selon laquelle les tâches communes aux deux sous-ensembles CHOITG(i) et CHOITD(i) doivent former un sous-ensemble critique ICOMMUN(i).

Nous conservons également les règles de traitement RT selon lesquelles

. CHOITD(i) est un sous-ensemble critique comme sous-ensemble d'un ensemble critique

. CHOIRD(i) est considéré et traité, dans la mesure du possible, comme un sous-ensemble critique.

a-2-5) Création de relations inter-tâches

$_{\alpha}$) Création de la relation dite de ≪succession possible≫ sur tout couple de tâches appartenant au même sous-ensemble (CHOITD(i) ou CHOIRD(i)).

On conserve la même restriction que dans l'étude gauche s'il existe déjà une relation de ≪succession possible≫ de rang inférieur.

β) Création de la relation dite de ≪voisinage possible≫ sur tout couple de tâches (l'une appartenant à CHOITD(i) et l'autre à CHOIRD(i)).

a-3 - Mise à jour des ensembles de tâches

α) On remet à jour d'abord les deux ensembles de tâches situées à gauche et à droite en enlevant celles qui viennent d'être choisies (pour constituer un des sous-ensembles complémentaires)

$$G(i) = G(i-1) \;/\; CHOIRD(i)$$

$$D(i) = D(i-1) \;/\; CHOIRD(i)$$

β) 1 - On remet également à jour l'ensemble TABG(i-1) dont les éléments sont susceptibles d'intervenir pour la régularité à gauche.

Il est constitué par

1) les tâches qui en faisaient déjà partie lors de l'itération précédente mais qui n'ont pas été utilisées dans CHOITG(i) en raison de la position géographique trop à gauche de leur bande de travail.

2) les tâches qui viennent d'être choisies dans l'ensemble des tâches restantes à gauche c'est à dire celles dont c'est la première apparition, apparition dûe à la position de la borne supérieure de leur bande de travail).

$$TABG(i) = (TABG(i-1) \;/\; CHOITG(i)) \cup CHOIRG(i)$$

2 - On obtient de même pour l'étude à droite

$$TABD(i) = (TABD(i-1) \;/\; CHOITD(i)) \cup CHOIRD(i)$$

B - CONTRUCTION PROGRESSIVE D'ORDONNANCEMENT

Toutes les relations inter-tâches que nous venons de créer sont regroupées dans une matrice que nous nommerons ≪ Matrice des Relations Inter-tâches ≫ (M. R. I)

A l'intersection de la $i^{ème}$ ligne et de la $j^{ème}$ colonne nous indiquerons, si elle existe, la relation liant les deux tâches i et j. Pour des raisons de commodité d'exploitation, nous indiquerons également cette relation à l'intersection de la $j^{ème}$ ligne et de la $i^{ème}$ colonne. La matrice sera donc une matrice symétrique.

Initialement toutes les tâches ont été dans notre exemple, considérées comme étant libres : aucune contrainte, aucune relation n'existe entre les tâches. Donc à l'origine tous les éléments de la Matrice MRI sont nuls.

Pour chaque relation créée entre les tâches i et j lors de la phase de "Partition en sous-ensembles complémentaires" nous allons placer un élément différent de zéro à l'intersection des $i^{ème}$ ligne et $j^{ème}$ colonne (respectivement $j^{ème}$ ligne et $i^{ème}$ colonne).

Cependant, nous voulons, avec cet élément situé en MRI(i, j), spécifier deux types de renseignements différents :

α) Le type de la relation liant les deux tâches. Rappelons qu'elle peut être

$\alpha 1$: de succession possible (s.p.)

$\alpha 2$: de voisinage possible (v.p.)

β) Le numéro de l'itération au cours de laquelle cette relation a été créée.

Nous avons fixé, par convention, la valeur de l'élément MRI(i, j) de la façon suivante :

- La relation s.p. obtenue à la $k^{ème}$ itération sera représentée par le nombre k+1
- La relation v.p. obtenue à la $k^{ème}$ itération sera représentée par le nombre 10(k+1)

Ce décalage d'une unité est dû au fait que nous utilisons le nombre 1 pour spécifier l'appartenance d'une tâche à l'ensemble de base.

. Lorsqu'une tâche i appartient à l'ensemble de base, nous donnons à l'élément MRI(i, i) la valeur 1, en d'autres termes la présence d'un nombre 1 sur la diagonale indique que la tâche correspondante appartient à l'ensemble de base (celui dont les tâches doivent être placées en priorité)

Ce mode de fixation peut être modifié légèrement si le cas traité est plus complexe. Ce mode-ci reste valable lorsque le nombre d'itérations nécessaires à la réalisation de la phase de partition en sous-ensembles complémentaires n'excède pas 18.

En effet, si le nombre d'itérations excède ce nb limite et si MRI(i, j) = 20 , on ne saura pas dire si la relation existante entre la tâche i et la tâche j est

. Soit une relation s.p. créée lors de la $19^{ème}$ itération

. Soit une relation v.p. créée lors de la $1^{ère}$ itération.

Les exemples que nous avons traité sont restés dans la limite acceptable de ce mode de fixation.

Cas particulier :

. Si une tâche de l'ensemble gauche ou droit constitue à elle seule un sous-ensemble complémentaire, on indiquera sur la diagonale le numéro (augmenté d'une unité) de l'itération pour laquelle cette tâche est obtenue. On considère que cette tâche est reliée à elle-même par la relation s.p.

Exploration de la Matrice et Ordonnancement

Nous allons explorer la matrice afin d'obtenir toutes les tâches et de les placer (i-e fixer leur instant de déclenchement).

Nous donnons seulement le canevas général.

Pour chaque itération n° K

nous procédons de la façon suivante :

- K partant de 1
- valeur de s.p. = K+1
- valeur de v.p. = 10(K+1)

I - ETUDE DU PREMIER COUPLE DE SUITES DE TACHES

A) Obtention de la première suite (formée de tâches de base)

1) Recherche d'une tâche de base (présence du nombre un sur la diagonale). Soit i cette tâche.

2) Recherche, pour cette tâche de base i, d'une autre tâche de base j qui lui soit liée par la relation s.p. relative à l'itération (présence dans la colonne i d'un élément de valeur K+1).

3) Vérification que i et j ne sont pas tâches "Communes"

- Si i est tâche "Commune", recommencer en 1)
- Si j est tâche "Commune", recommencer en 2)

4) a) On place alors la tâche i à la suite de la dernière tâche de base placée lors de l'itération précédente.

b) Comme on a déjà le suivant possible j , on supprime les éléments de la colonne (et de la ligne) i qui sont relatifs à des relations de succession possible.

Puis on continue la construction de cette suite en faisant jouer à la tâche j le rôle tenu par la tâche i , c'est à dire

- Recherche dans la colonne j d'un élément de valeur i+1. Soit k' la ligne pour laquelle cet élément est obtenu.
- Vérification que k' n'est pas tâche "Commune" sinon recherche d'une autre tâche k'.
- Placement de la tâche j.
- Suppression des autres relations de succession possible concernant j.

Puis on fait jouer à k' le rôle de j et ainsi de suite jusqu'à ce que l'on ne trouve plus d'élément de valeur K+1. A ce moment là on passe à l'obtention de la suite complémentaire.

B) Obtention de la suite complémentaire

1) Recherche d'une tâche-tête de série : on cherche la première tâche - possible voisinne - de la tâche de base i (la première obtenue dans la phase A). Pour cela on cherche dans la colonne i un élément de valeur 10(K+1). Soit j la ligne pour laquelle cet élément est obtenu.

2) Puis on obtient la série à partir de cette tâche-tête de série en appliquant le procédé vu plus haut (mais ici on ne s'attache pas

à la vérification de tâches "Communes" car ces cas ne se produisent qu'avec les tâches de base).

- On cherche donc un successeur possible pour la tâche j (recherche dans la colonne j d'un élément de valeur K+1). Soit ℓ la ligne pour laquelle cette valeur est obtenue.

- On place la tâche j : on place cette tâche à côté de la tâche i obtenue plus haut (i-e on la déclenche en même temps). Cependant si lors de l'itération précédente on a dégagé plus de temps que nécessaire et que ce temps soit utilisable à cette itération, on placera la tâche j dès que possible.

  On supprime tous les éléments relatifs aux relations de succession possible.

- Puis on fait jouer à ℓ le rôle tenu par j et ainsi de suite, jusqu'à ce que la série soit achevée.

II - ETUDE DU DEUXIEME COUPLE DE SUITES DE TACHES

2 cas peuvent se produire :

Cas 1 : Il existe des tâches de base "Communes"

Dans ce cas : C-1 placement des tâches de base "Communes" les unes à la suite des autres à partir de la dernière tâche de base placée.

C-2 Suppression, au fur et à mesure, des éléments de relations ≪successions possibles≫ dans les colonnes (et lignes)correspondantes.

C-3 Conservation de la dernière tâche "Commune" qui servira de tête de série pourl'obtention de la deuxième série de tâche de base.

C-4 On effectue alors la même méthode de recherche et d'exploration mais on ne fait pas l'étude (A-3) de vérification (en effet les tâches "Communes" ont été explorées lors de C-1.

On réalise donc les points A-2 ; A-4 ; B-1 ; B-2 ;

Cas 2 : Il n'existe pas de tâche de base "Communes".

Dans ce cas, l'étude des 2 suites complémentaires reste la même, hormis le point A-3 ici aussi inutile.
On réalise donc A-1 ; A-2 ; A-4 ; B-1 ; B-2.

S'il reste après cette itération K des tâches à placer, on refait une itération supplémentaire.

ETUDE D'UN EXEMPLE SIMPLE

α) Nous présentons dans le tableau V.EX1 la liste des tâches avec pour chaque tâche sa bande de travail et sa durée (ce seront les seules données fournies au programme).

N° des tâches	Bande de Travail		Durée (en Δt)
	Borne inférieure	Borne supérieure	
1	17	20	5
2	13	17	3
3	9	11	4
4	7	10	2
5	13	15	3
6	14	20	5
7	7	11	3
8	8	13	3
9	11	17	3
10	10	13	4
11	14	17	1
12	13	16	2
13	9	14	5
14	11	14	1

Tableau V.EX1

β) Dans le tableau V.EX2 , nous indiquons la durée d'occupation de chaque case et mettons en évidence la case de base.

Case de base ↓

N° de case	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Durée générale d'occupation	5	8	17	21	19	20	24	23	17	14	17	10	10	10

Tableau V.EX2

γ) Dans le tableau V.EX3, se trouvent regroupés les résultats de la partition de l'ensemble des tâches en 3 sous-ensembles : ensemble de base, gauche et droit.

	N° des tâches							
Ensemble de base	2	5	8	9	10	12	13	14
Gauche	3	4	7					
Droit	1	6	11					

Tableau V.EX3

δ) Puis l'étude itérative et construction de la matrice MRI des relations inter-tâches.

Itération 1

α) Etude à gauche

. GMAX = MAMABA(G(1)) = 11

CHOIRG(1) = 3, 7

CHOITG(1) = 2, 5, 12

β) Etude à droite

. DMIN = MIMIBA(D(1)) = 14

CHOIRD(1) = 6, 11

CHOITD(1) = 8, 10

Itération 2

α) Etude à gauche

. GMAX = MAMABA(G(2)) = 10

CHOIRG(2) = 4

CHOITG(2) = 9, 14

β) Etude à droite

. DMIN = MIMBA(D(2)) = 17

CHOIRD(2) = 1

CHOITD(2) = 5, 12, 13, 14

γ) Etude des Tâches-communes

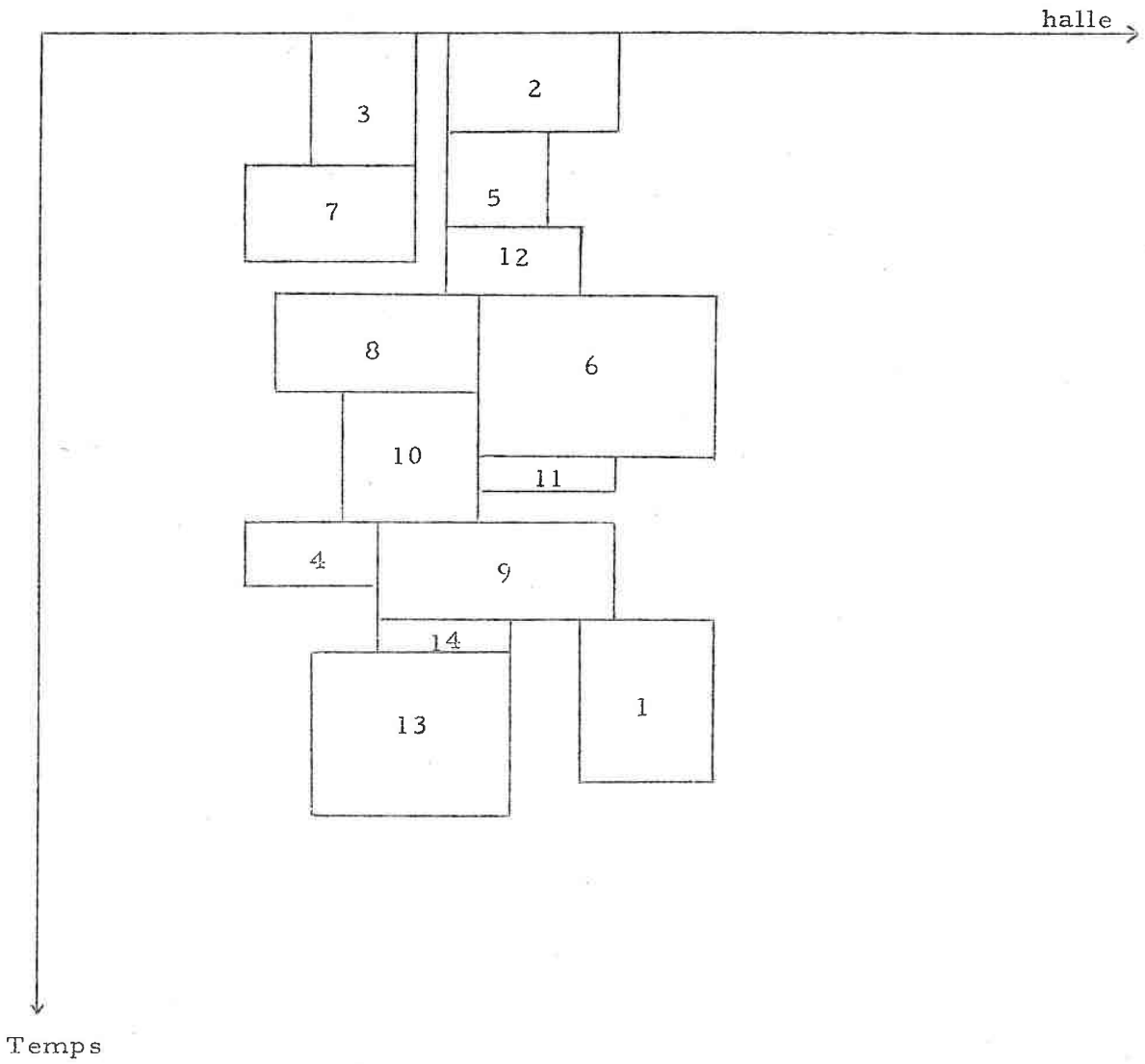
. COMMUN = 14

ϵ) Le résultat obtenu : la représentation graphique du processus généré par exploration de la Matrice.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	3				30							30	30	30	
2		1	20		2		20					2			
3		20			20		2					20			
4				3					30					30	
5	30	2	20		1		20					2	3	3	
6								20		20	2				
7		20	2		20							20			
8						20		1		2	20				
9				30					1					3	
10						20		2		1					
11						2		20							
12	30	2	20		2		20					1			
13	30				3								1	3	
14	30			30	3				3				3	1	
15															

COMMUNE(1) = Néant
COMMUNE(2) = 14

MATRICE MRI


halle
3
2
7
5
12
8
6
10
11
4
9
14
13
1
Temps

Etude de l'introduction de Tâches Prioritaires

Nous venons d'expliciter une méthode d'ordonnancement d'un ensemble de tâches libres. Rappelons que nous avons pensé et résolu, en termes de rectangles, ce problème de génération d'un processus.

Il paraît intéressant de voir comment se modifierait le processus ainsi généré si on introduit des tâches prioritaires.

Rappelons que par tâches prioritaires (cf. chapitre III, p. 8) nous entendons les tâches dont la réalisation doit avoir lieu à une date ficée par avance.

Le problème d'ordonnancement que l'on a à résoudre peut s'énoncer de la façon suivante :

Nous disposons :

. d'une part, d'un ensemble E de tâches libres (sans contrainte entre elles) dont nous avons déjà explicité les caractéristiques

. d'autre part, d'un ensemble F de tâches prioritaires munies chacune de

α) son instant de déclenchement obligatoire

β) sa bande de travail, c'est à dire les cases extrêmes de la halle entre lesquelles se déroulera la tâche.

Nous désirons : obtenir l'ordonnancement de l'ensemble $E \cup F$ de toutes les tâches à réaliser compte tenu du fait que les tâches prioritaires doivent avoir lieu à des instants déterminés à l'avance.

Nous conservons : les contraintes R1, R2, R3 déjà vues (règles de sécurité, de nombre de ponts, de zone d'activité).

Pour résoudre ce problème, nous utiliserons à nouveau une interprétation géométrique et pour cela transposons le problème en termes de ≪contours≫.

Nous désirons : placer dans le système d'axes temps et halle tous les rectangles avec pour objectif de tous les placer en allant le moins loin possible sur l'axe des temps, compte tenu du fait que les rectangles ≪prioritaires≫ ont une position déterminée à l'avance dans ce système d'axe.

Nous conservons : les contraintes R'1, R'2, R'3 transposées des contraintes R1, R2, R3 (cf. chapitre V, page

La méthode employée :

Nous commençons par placer tous les rectangles ≪libres≫ en appliquant la méthode décrite plus haut.

Nous disposerons, alors, d'un processus composé uniquement de tâches libres. Nous placerons ensuite les rectangles ≪prioritaires≫ les uns après les autres.

Nous conservons : les contraintes R1, R2, R3 déjà vues (règles de sécurité, de nombre de ponts, de zone d'activité).

Pour résoudre ce problème, nous utiliserons à nouveau une interprétation géométrique et pour cela transposons le problème en termes de ≪contours≫.

Nous désirons : placer dans le système d'axes temps et halle tous les rectangles avec pour objectif de tous les placer en allant le moins loin possible sur l'axe des temps, compte tenu du fait que les rectangles ≪prioritaires≫ ont une position déterminée à l'avance dans ce système d'axe.

Nous conservons : les contraintes R'1, R'2, R'3 transposées des contraintes R1, R2, R3 (cf. chapitre V, page

La méthode employée :

Nous commençons par placer tous les rectangles ≪libres≫ en appliquant la méthode décrite plus haut.

Nous disposerons, alors, d'un processus composé uniquement de tâches libres. Nous placerons ensuite les rectangles ≪prioritaires≫ les uns après les autres.

Pour chaque rectangle ≪prioritaire≫ l'opération se fera comme suit :

(α) La position d'un rectangle ≪prioritaire≫ étant imposée aussi bien sur l'axe temporel que sur l'axe spatial, nous plaçons ce rectangle dans le système d'axes. Mais ce placement peut entrainer des chevauchements de ce rectangle ≪prioritaire ≫ avec des rectangles ≪libres≫.

(β) Pour respecter la contrainte R'_1 de sécurité qui interdit les chevauchements, nous appliquerons la règle de décalage qui consiste à décaler tous les rectangles ≪libres≫ à partir du premier rectangle (compris) chevauché par le rectangle ≪prioritaire≫.

(γ) Cependant cette règle de décalage est trop ≪grossière≫ aussi nous avons, dans un but d'amélioration du processus obtenu, adopté une règle d'affinage appelée règle de remontée. Cette règle consiste à étudier chaque rectangle ≪libre≫ décalé et à essayer de le remonter dans le temps tout en respectant la contrainte R'_1 de sécurité.

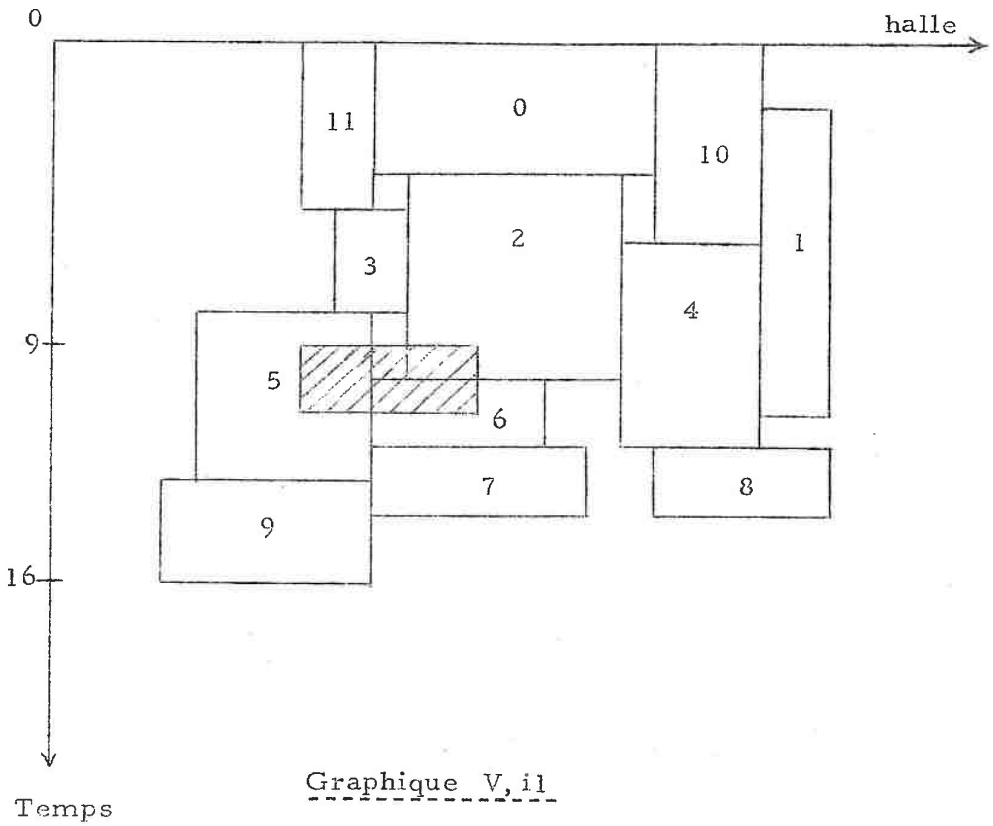
Application à un exemple simple :

(α) Sur le graphique V, i1 , nous représentons le processus composé de tâches libres et nous avons fait figurer en grisé la tâche prioritaire, imposant son instant de déclenchement à l'instant 9. Le premier rectangle chevauché par ce rectangle ≪prioritaire≫ est le rectangle n° 2.

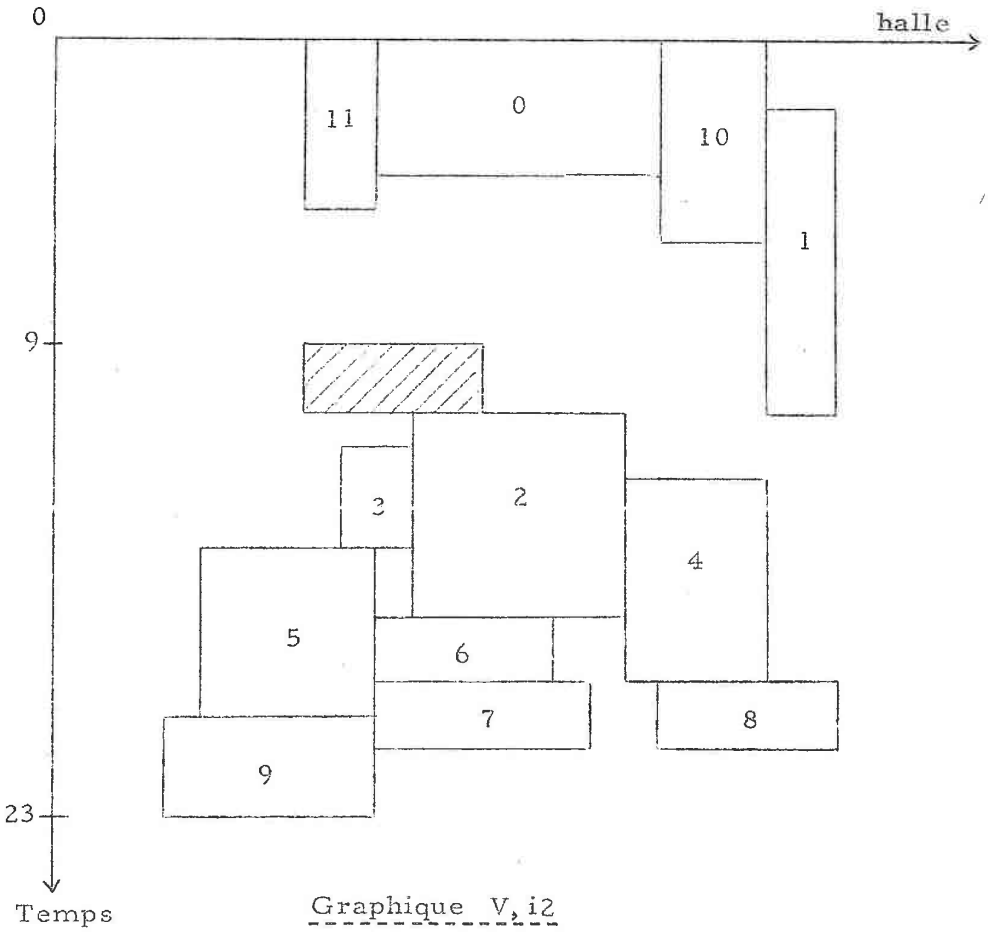
(β) Nous avons présenté sur le graphique V, i2 , la situation après application de la règle de décalage. Le temps nécessaire à la réalisation du processus est de $23\,\Delta t$, alors que le processus composé de tâches ≪libres≫ durait $16\,\Delta t$ et que la durée de la tâche prioritaire n'est que de $2\,\Delta t$.

(γ) Nous avons, sur le graphique V, i3 figuré la situation après application de la règle de remontée. Le temps nécessaire à la réalisation du processus n'est plus que de $21\,\Delta t$.

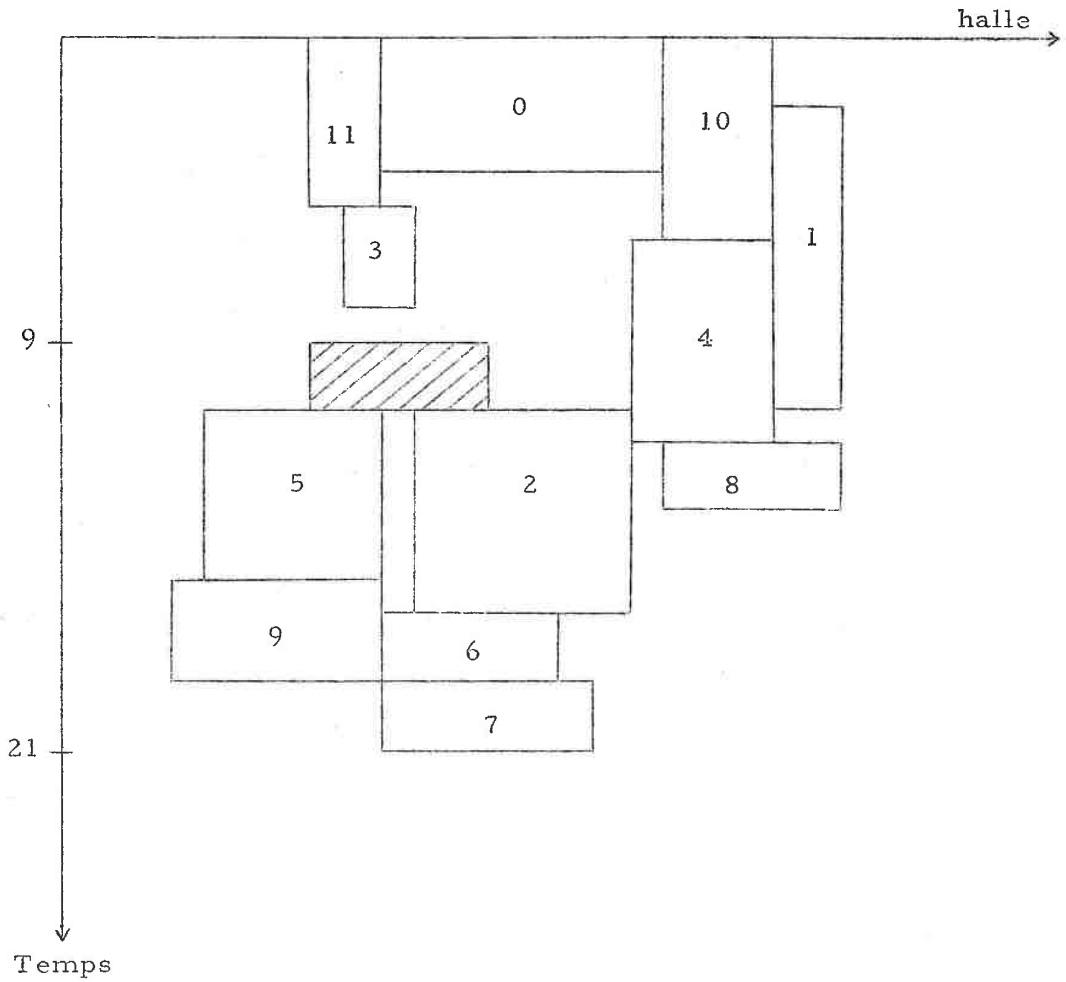
Conclusion : Il n'en demeure pas moins un retard de $3\,\Delta t$ sur la durée "optimum" du processus obtenue en ajoutant à la durée du processus composé de tâches libres la durée de la tâche prioritaire. Nous avons formalisé ces deux règles de décalage et de remontée et rédigé le programme correspondant.



Graphique V, i1



Graphique V, i2



Graphique V, i3

Les résultats obtenus par programme ont montré que l'introduction de tâches prioritaires entrainait une légère altération de l' "optimalité" du processus.

Cependant, de façon générale, le retard engendré varie selon la structure de l'activité de production. Par exemple, si la tâche prioritaire a lieu dans une partie peu fréquentée de la halle, il peut ne pas y avoir d'altération du tout. Mais si la tâche a lieu dans la zone la plus occupée, le retard peut être important.

Nous avons regroupé dans le tableau V, i4 certains des résultats obtenus. Dans la colonne 2 nous avons fait figurer la durée du Processus obtenu par ROSEC . Dans la colonne 3 nous avons indiqué le nombre de tâches prioritaires à introduire mais nous avons voulu différencier les tâches prioritaires recouvrant la case de base (donc à placer dans la zone la plus occupée) et les tâches prioritaires ne la recouvrant pas.

Nous avons rapproché ensuite (col. 8 et col. 9)

α) la durée ≪optimum≫ c'est à dire celle obtenue en ajoutant à la durée du processus (col. 2) la durée des tâches prioritaires recouvrant la case de base.

γ) la durée réelle du processus généré par programme.

Nous avons indiqué en dernière colonne l'écart entre ces deux durée.

nb. de tâches libres	durée du processus «libre»	nb. de tâches prioritaires	nb. de tâches prioritaires occupant la case de base	durée de celles-ci	nb. de tâches prioritaires n'occupant pas la case de base	durée de celles-ci	durée «optimum»	durée réelle	retard
62	4h31'15"	3	2	2'30	1	2'	4h33'45"	4h36'45"	3'
62	4h31'15"	10	4	19'	6	24'	4h50'15"	5h34'15"	44'
62	4h31'15"	15	4	19'	11	43'	4h50'15"	5h34'15"	44'
62	4h31'15"	17	7	40'	10	42'	5h11'15"	5h28'45"	17'30

Tableau V,i4

Que tirer de ces résultats (dépendant du jeu d'essai, rappelons-le)

- La comparaison des lignes 2 et 3 montrent que l'introduction de 5 tâches prioritaires supplémentaires dans des zones relativement peu occupées de la halle n'altère en rien le processus.

- La comparaison des lignes 3 et 4 montrent que l'altération d'un processus (i-e l'importance du retard) ne dépend pas seulement du nombre de tâches prioritaires mais surtout de l'endroit où elles doivent être implantées.

B - L'ANALYSE FONCTIONNELLE

(I) Fichier et Table permanents concernant l'activité de production.

① Table "TTACH" fichier des tâches

Revoir la définition donnée au chapitre II, page 21.

② Fichier "TCOMP" fichier des composantes

Revoir la description donnée au chapitre II, page 22.

(II) Les traitements Le programme définit

A - Pour chaque case de la halle sa durée d'occupation généralisée.

B - La case de base et les ensembles BASE, GAUCHE et DROIT.

C - Pour chaque itération i

C-1 - Les ensembles G(i), D(i), TABD(i), TABG(i)

C-2 - Les ensembles CHOITG(i), CHOIRG(i), CHOITD(i), CHOIRD(i), ICOMMUN(i)

D - La matrice MRI des relations inter-tâches

E - L'exploration de la Matrice

F - Le Processus ainsi généré

G - Le Processus après aménagement éventuel et contrôle de validité

H - La sortie du Processus "Matérialisée" par les "cartons".

DESCRIPTION MODULAIRE

NIVEAU I

I - TRAITEMENT

1 - Lecture des caractéristiques générales du problème

2 - Lecture des fichiers tâches et composantes relatifs à ce problème et exploitation

3 - Recherche de la case de base

4 - Partition de l'ensemble des tâches en trois sous-ensembles (base, gauche et droit)

5 - Pour chaque itération :

> 6 - Impression des renseignements dont on dispose (A-D) *
>
> 7 - Obtention des deux couples de sous-ensembles complémentaires et par création des relations (de succession possible et de voisinage possible) contribution à la construction de la matrice des relations inter-tâches.

8 - Sortie de cette matrice des relations inter-tâches (A-D)

9 - Génération d'un processus par exploration de la matrice

10 - Aménagement éventuel si la durée du processus est supérieure à la durée d'occupation de la case de base (1er objectif recherché)

11 - Contrôle de validité du processus

12 - Visualisation du processus par la "Sortie" des cartons (A-D)

Déclaration : fichiers TTACH, TCOMP, TPROC

---

* A-D Les modules suivis de ces abréviations A-D (Aide à la Décision) sont des modules utiles à l'utilisateur dans sa recherche de renseignements lui permettant d'affiner la solution que proposera la "Maquette".

NIVEAU II

I-1 - Lecture des caractéristiques générales du problème

1 - Lecture du nombre de tâches à effectuer

2 - Lecture du nombre de ponts dont on dispose

NIVEAU II

I-2 - Lecture des fichiers relatifs à ce problème et exploitation

1 - Lecture du nombre de composantes de chaque tâche dans le fichier TTACH

2 - Pour chaque tâche

| 3 - Pour chacune de ses composantes, lecture de ses caractéristiques dans le fichier TCOMP
| 4 - Calcul de sa durée
| 5 - Calcul des déplacements effectués
| 6 - Calcul de sa bande de travail

Déclaration : fichiers TTACH et TCOMP

NIVEAU II

I-3 - Recherche de la case de base

1 - Pour chaque case de la halle : calcul de sa durée d'occupation

2 - Rangement des cases par ordre de durée décroissante

3 - Obtention de la case de base

4 - Impression des résultats (A-D)

NIVEAU II

I-4 - Partition de l'ensemble des tâches en trois sous-ensembles

1 - Pour chaque tâche

| 2 - Si sa bande de travail recouvre la case de base
| | 3 - Mise de la tâche dans l'ensemble de base
| | 4 - Indication dans la matrice des relations inter-tâches de l'appartenance de cette tâche à l'ensemble de base
| 5 - Sinon, selon sa position, mise dans l'ensemble gauche ou droit
| 6 - Impression des résultats

NIVEAU II

I-6 - Impression des renseignements dont on dispose

1 - Ensemble des tâches restantes à gauche

2 - Ensemble des tâches restantes à droite

3 - Ensemble des tâches n'étant pas intervenues pour la régularité de l'enveloppe gauche

4 - Ensemble des tâches n'étant pas intervenues pour la régularité de l'enveloppe droite

5 - Surplus du temps dégagé à gauche

6 - Surplus du temps dégagé à droite

NIVEAU II

I-7 - Obtention des deux couples de sous-ensembles complémentaires et etc...

1 - Pour l'étude à gauche puis pour l'étude à droite

2 - Obtention du sous-ensemble des tâches restantes vérifiant la propriété relative à l'itération

3 - Création de relation de "succession possible" entre elles et indication dans les éléments correspondants de la matrice des relations inter-tâches (M. R.I)

4 - Obtention du sous-ensemble complémentaire

5 - Création de relation de "succession possible" entre les tâches de ce sous-ensemble et contribution à la construction de la M. R. I

6 - Création entre les tâches des deux sous-ensembles de relation de "voisinage possible" et contribution à la construction de la M. R. I

7 - Mise à jour des renseignements à partir desquels l'itération suivante sera construite

8 - Obtention de l'ensemble des tâches "communes"

NIVEAU II

I-9 - Génération d'un processus par exploration de la matrice M. R. i

1 - Pour chaque itération

2 - Obtention et placement de la première suite formée par les tâches de base reliées entre elles par la relation de "succession possible" (relative à cette itération), à l'exclusion des tâches "communes" et de celles déjà placées

3 - Obtention et placement de la suite formée par les tâches reliées à celles de cette première suite par la relation de "voisinage possible"

4 - Obtention et placement des tâches de bases "communes" de cette itération

5 - Obtention et placement de la deuxième suite formée par les tâches de base reliées entre elles par la relation de "succession possible" (relative à cette itération) à l'exclusion des tâches déjà placées

6 - Obtention et placement de la suite formée des tâches reliées à celles de cette deuxième suite par la relation de "voisinage possible"

Déclaration : fichier TPROC

NIVEAU II

I-10 - Aménagement éventuel

1 - Vérifier que la date de fin de processus est égale à l'heure de fin d'occupation de la case de base (on ne peut pas diminuer davantage la durée du processus).

2 - Si ce n'est pas le cas

3 - Pour chaque tâche dont l'instant de déclenchement est postérieur à cette heure de fin d'occupation de la case de base

4 - Recherche d'un emplacement (horaire) susceptible d'acceuillir cette tâche

5 - Placement de cette tâche

6 - Remise en ordre chronologique du fichier processus

Déclaration : fichier Processus

NIVEAU II

I-11 - Contrôle de validité

1 - Pour tout couple de tâches

2 - Vérifier que leurs bandes de travail ne se chevauchent pas

3 - Si ce n'est pas le cas

4 - Sortie de renseignements sur ce "conflit"

5 - Replacement de l'une des deux tâches

6 - Remise en ordre du fichier processus

Déclaration : fichier Processus

NIVEAU II

I-12 - Visualisation du processus par "sortie" des cartons

1 - Impression du fichier processus

2 - Pour chaque Δt

3 - Recherche des tâches du processus qui débutent à cet instant

4 - Vérifier que le nombre de ponts disponibles est au moins égal au nombre de tâches déclenchées

5 - Si ce n'est pas le cas, décalage des tâches en surnombre

6 - Pour chacune des tâches effectivement réalisables

7 - Obtention des renseignements relatifs à cette tâche

8 - Rangement de ces renseignements dans une colonne du tableau à partir duquel est préparé l'impression

9 - Pour chaque tâche en cours d'exécution (i-e colonne du tableau)

10 - Mise à jour des renseignements relatifs à cette tâche

11 - Constitution à partir de ceux-ci de la ligne partielle d'impression

12 - Constitution de la ligne complète à partir des lignes partielles et impression

Déclaration : fichier Processus

CONCLUSION

Un certain nombre de conclusions semblent s'imposer à la suite de l'étude qui vient d'être faite.

Le problème ne se prêtant pas à un formalisme mathématique, seule une "Maquette" construisant une image aussi fidèle que possible de la réalité pouvait nous donner les éléments qui permettent d'agir sur le système.

Les principales difficultés dans la construction de la "Maquette", outre les problèmes particuliers de programmation*, résident dans l'analyse du système, dans la mise en évidence des paramètres qui le définissent et dans la formalisation des règles le caractérisant.

Nous nous sommes attachés à ce que tous les résultats obtenus soient exploitables pour une interprétation par des psychologues. Les indicateurs dont nous avons muni la "Maquette" aident à juger le niveau de sécurité des processus testés. La "Maquette" elle-même, permet de vérifier, à peu de frais et sans perturber l'exploitation réelle, la connaissance du système qu'ont les opérateurs et la façon dont ils pensent leur travail. Les méthodes heuristiques d'affectation et d'ordonnancement proposées introduisent en priorité des règles de sécurité.

Afin de rendre immédiatement exploitable ces divers éléments et méthodes, nous envisageons la rédaction du dossier d'exploitation des programmes déjà rédigés.

---

* Par exemple, le Format ajustable nous fut d'une aide très précieuse pour la visualisation du processus.

Nous y ferons figurer les renseignements utiles à celui qui, confronté à ce type de problème, désirerait le résoudre, au moins, partiellement. Ce dossier comportera deux parties :

α) un court exposé de la méthode d'analyse et des méthodes de résolution.

β) des renseignements purement techniques pour l'utilisation proprement dite des programmes.

L'Ampleur du sujet sur lequel porte cette étude ne nous a pas permis de l'explorer entièrement mais nous pensons que notre contribution, outre les résultats déjà acquis, rend possible d'éventuels prolongements.

Parmi les prolongements possibles citons en quelques uns que nous avons abordés, pour lesquels nous proposons quelques idées mais qui devront être développées.

Une extension possible peut être l'aide à la prévision d'implantation de ponts-roulants. Pour ce faire, on conserve l'ordonnancement des tâches tel qu'il est généré par programme. Mais on fait varier le paramètre ≪nombre de ponts disponibles≫. Par programme on recherche les moments où le nombre de tâches à effectuer en même temps est supérieur au nombre de ponts disponibles. On déplace les tâches en surnombre pour les positionner à des instants où le nombre des tâches réalisées sera au plus égal au nombre de ponts disponibles. On obtient une solution bonne mais approchée que l'on peut affiner manuellement, si nécessaire.

Dans cette optique nous avons utilisé quatre jeux d'essais et nous avons donné trois valeurs différentes au paramètre "nombre de ponts disponibles". Nous avons regroupé dans le tableau suivant C, 1 les résultats obtenus (dépendant de la structure des jeux d'essais)

n° de l'essai	Durée du processus avec		
	4 ponts	3 ponts	2 ponts
1	4 h 31	4 h 31	4 h 48
2	4 h 13	4 h 13	4 h 36
3	2 h 45	2 h 45	3 h 06
4	2 h 28	2 h 28	3 h 07

Tableau C, 1

Remarque : La durée du processus est la même que l'on travaille à 4 ou 3 ponts. Ceci s'explique quand on sait que l'activité de production n'est pas également répartie dans la halle. A priori, la solution à 3 ponts semble la plus économique. Cependant la décision ne peut pas être prise à la simple vue de ces résultats, il est nécessaire de faire une étude approfondie des conditions et de la charge de travail de chaque pontonnier dans les 2 systèmes.

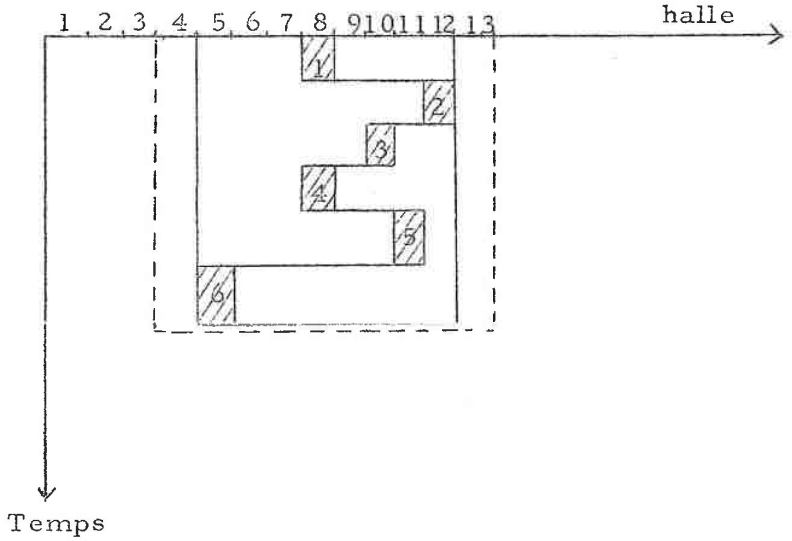
Il peut être intéressant aussi de traiter le problème de la panne d'un ou plusieurs ponts. Il faudra alors envisager deux cas possibles :

α) C'est un pont extrême qui est en panne. S'il est possible de le pousser en bout de la halle, à un endroit où il ne génera pas le travail des autres ponts : le problème redevient du même type que celui de la prévision d'une implantation, c'est à dire faire faire le travail par les n-1 ponts restants.

β) C'est un pont médian qui est en panne.

β1 - S'il ne peut pas être déplacé, on se trouve confronté à une partition de la halle en deux ≪ semi-halles≫ indépendantes. Les tâches devant se dérouler à la fois dans les deux ≪ semi-halles≫ ne peuvent pas être réalisées.

β2 - S'il peut être déplacé, on pourra (sous réserve de possibilité technique) l'accoler à un pont voisin et considérer ce pont voisin comme ayant une longueur double. On doit pouvoir résoudre le problème en conservant la méthode adoptée si on prend la précaution d'augmenter la dimension spatiale des cartons (matérialisant les tâches) de deux cases (une à droite et une à gauche) de façon à permettre la position du pont en panne à côté du pont qui réalise effectivement la tâche. Voyons à l'aide du graphique C,2 la raison de cet ≪élargissement≫



Graphique C,2

1er cas : Soit i-1 le numéro du pont en panne accolé au pont i . Lors de la réalisation de la 6ème composante de la tâche (borne inférieure de la bande de travail), le pont i occupera la case n° 5 ce qui aura pour effet de faire occuper la case n° 4 par le pont i-1 et donc d'empêcher toute réalisation de tâche utilisant cette case n° 4. On élargit donc d'une case à gauche le "carton" matérialisant la tâche.

2ème cas : Soit i+1 le numéro du pont en panne accolé au pont i . Lors de la réalisation de la 2ème composante de la tâche (borne supérieure de la bande de travail), le pont i occupera la case n° 12 , donc le pont i+1, accolé, occupera alors la case n° 13 interdisant de ce fait toute tâche utilisant cette case. On élargit donc d'une case à droite le "carton" matérialisant la tâche.

Signalons aussi, sans vouloir être exhaustif, d'autres extensions intéressantes à réaliser .

- L'introduction de contraintes entre les tâches. Ces contraintes pourront être des contraintes d'antériorité ou de simultanéité entre les tâches, c'est à dire que la matrice des relations inter-tâches (MRI) ne sera pas nulle dans son état initial, l'exploration de la matrice devra être adaptée pour traiter en priorité les contraintes imposées.

- L'introduction d'une fonction d'exclusion, en d'autres termes que certaines tâches ne pourraient être réalisées que par certains ponts aux caractéristiques techniques adéquates.

De façon générale, nous pensons que toutes les méthodes que l'on pourrait élaborer sur ce sujet devraient l'être avec comme objectif principal : l'Aide à la décision en développant l'aide à la conception et surtout l'aspect ≪dialogue≫ entre l'utilisateur et son programme une fois que ce programme aura fourni une bonne solution.

ANNEXE II

II-1 - Quelques définitions utiles à la compréhension de la description modulaire.

II-2 - Liste des tâches

II-1 - Quelques définitions utiles à la compréhension de la description modulaire

La description du programme se fait en plusieurs niveaux. Certains éléments d'un niveau étant redétaillés au niveau suivant, la description part d'un niveau global (7 grandes parties, ici) pour arriver à un niveau assez détaillé (parfois celui de l'instruction du langage utilisé).

Ce mode de description permet d'atténuer les inconvénients de l'organigramme qui sont surtout la difficulté de lecture par un non spécialiste, la difficulté de modification. En effet, si l'on veut modifier une partie de l'organigramme, on est obligé de refaire entièrement l'organigramme, alors qu'avec cette description il suffira de modifier le module correspondant (voir définition d'un module ci-après).

Donnons alors les définitions utiles à l'établissement de cette description :

- un programme sera constitué d'une suite de modules
- un module est formé :

a) d'un titre (un indicateur) et éventuellement de commentaires
b) d'un corps (une séquence d'opérations)

chaque opération est :
- soit un module
- soit une table de décision
- soit une opération de répétition
- soit un retour à un indicateur de module
- soit une action
- soit un organigramme, notion qui semble s'introduire au niveau organique.

c) d'une déclaration (la liste des fichiers nécessaires et des documents à établir, les fichiers seront en entrée (les données), en sortie (les résultats), en entrée-sortie (les fichiers mis à jour), en sortie-entrée (les fichiers de travail). Ceci est fort utile lors d'un programme nécessitant beaucoup de place en mémoire. Lors du traitement d'un niveau, seuls seront gardés en mémoire centrale les fichiers apparaissant en déclaration.

- Définition des opérations :

1) Opération de répétition :

Pour chaque index [faire]

A_1
A_2
:
A_4

} séquence d'opérations à réaliser

Index : unité d'un compteur, exemples :

α) : Δt, (augmentation du temps de Δt)
β) : enregistrement d'un fichier.

2) Table de décisions : (ici table de décision à entrée limitée)

La table de décision est composée de deux souches :

a) Une souche des conditions :

- chaque condition occupe une ligne
- toutes les conditions élémentaires doivent y apparaître
- les conditions n'ont que deux réalisations possibles : vrai ou faux.

b) Une souche des actions :

- chaque action occupe une ligne
- doivent apparaître toutes les actions possibles dans le cadre de ce problème
- les actions ont deux réalisations possibles ? Exécution ou non de l'action représentée respectivement par un nombre ou rien.

Le nombre indique : ⟶ l'exécution de l'action a lieu

⟶ le numéro d'ordre dans la séquence "ensemble-action" ce numéro est très important car certaines actions ne peuvent être exécutables qu'avec le résultat de certaines autres.

Remarque : Si on a n conditions, on pourra avoir jusqu'à 2^n réalisations possibles de l'ensemble des conditions, donc 2^n régles.

Une règle : expression logique qui, à une réalisation simultanée des conditions, fait correspondre une réalisation simultanée des actions.
Cette propriété permet de se rendre compte, lors de l'analyse d'un problème, que toutes les éventualités ont été prises en considération.

La séquence ordonnée des actions déterminées par une règle R_j est appelée "Ensemble-action" d'une règle R_j.

Cependant, dans de très nombreux cas, une partie des règles définies n'a aucun intérêt pour le problème, on a alors une table incomplète, les éventualités non prévues n'étant pas prises en considération.

Conclusion :
---

* On dispose d'un mode de représentation accessible et utilisable par tous. Outil commun pour le psychologue et l'informaticien.

* On dispose d'un mode de description qui facilite l'explication (on peut se placer à des niveaux différents, niveau global jusqu'au niveau de détail).

* Autre avantage : il existe des formes de table de décision acceptable par un programme traducteur (d'où gain de place en mémoire. gain de temps).

1-2 - Liste des tâches dans l'exemple traité

N° 1 - Changement de benne à crasse au four 3
N° 2 - Changement de benne à crasse au four 2
N° 3 - Enlever les plateaux de la fosse à coulée au four 3 par le pont 5
N° 4 - Remettre les plateaux de la fosse à coulée au four 3 par le pont 5
N° 5 - Enlever les plateaux de la fosse à coulée au four 3 par le pont 7
N° 6 - Remettre les plateaux de la fosse à coulée au four 3 par le pont 7
N° 7 - Enlever les plateaux de la fosse à coulée au four 2
N° 8 - Remettre les plateaux de la fosse à coulée au four 2
N° 9 - Alimentation du four 3 en panier de ferrailles, quartier d'orange
N° 10 - Alimentation du four 3 en panier de ferrailles coquilles
N° 11 - Alimentation du four 2 en panier de ferrailles, quartier d'orange
N° 12 - Alimentation du four 2 en panier de ferrailles coquilles
N° 13 - Utilisation de l'appareil block au four 2
N° 14 - Utilisation de l'appareil block au four 3
N° 15 - Changement d'électrode au four 2
N° 16 - Changement d'électrode au four 3
N° 17 - Raboutage près du four 2
N° 18 - Raboutage près du four 3
N° 19 - Alimentation du four 3 en benne à coke
N° 20 - Alimentation du four 2 en benne à coke
N° 21 - Alimentation du four 2 en benne à chaux
N° 22 - Alimentation du four 3 en benne à chaux
N° 23 - Démoulage des bennes à crasses par le pont 4
N° 24 - Démoulage des bennes à crasses par le pont 5
N° 25 - Coulée au four 3 par le pont 4
N° 26 - Coulée au four 2
N° 27 - Reprendre la poche au skip
N° 28 - Coulée au four 3 avec incident éventuel au skip
N° 29 - Préparation de la poche
N° 30 - Evacuation des lingotières pleines

N° 31 - Sortie du plateau du fond de la fosse
N° 32 - Mise du plateau au fond de la fosse
N° 33 - Sablage
N° 34 - Transfert de briques
N° 35 - Décharges des lingotières vides
N° 36 - Sortie de mère
N° 37 - Remise de mère
N° 38 - Accroche le fond de lingot
N° 39 - Mise des lingotières dans la fosse
N° 40 - Transfert de passerelles
N° 41 - Transport des bennes à laitier pleines
N° 42 - Transport des bennes à laitier vides
N° 43 - Descendre les lingots dans la fosse
N° 44 - Change le container à Dolomie vide sur le wagon
N° 45 - Amène le container à Dolomie plein
N° 46 - Caisse de sécurité au four 2
N° 47 - Caisse de sécurité au four 3
N° 48 - Utilisation de la masse à damer au four 2
N° 49 - Utilisation de la masse à damer au four 3
N° 50 - Travail dans les cases 1 - 2 - 3 - 4
N° 51 - Prise du palonnier de panier de ferrailles
N° 52 - Prise du palonnier pour la poche de coulée par le pont 4
N° 53 - Prise du palonnier benne marrel
N° 54 - Prise de la chaîne à 4 brins
N° 55 - Prise des élingues
N° 56 - Descend le fond de lingot
N° 57 - Utilisation de la poire
N° 58 - Benne cail au four 2
N° 59 - Benne cail au four 3
N° 60 - Utilisation de l'électro-aimant
N° 61 - Transfert de benne marrel vide
N° 62 - Transfert de benne marrel pleine

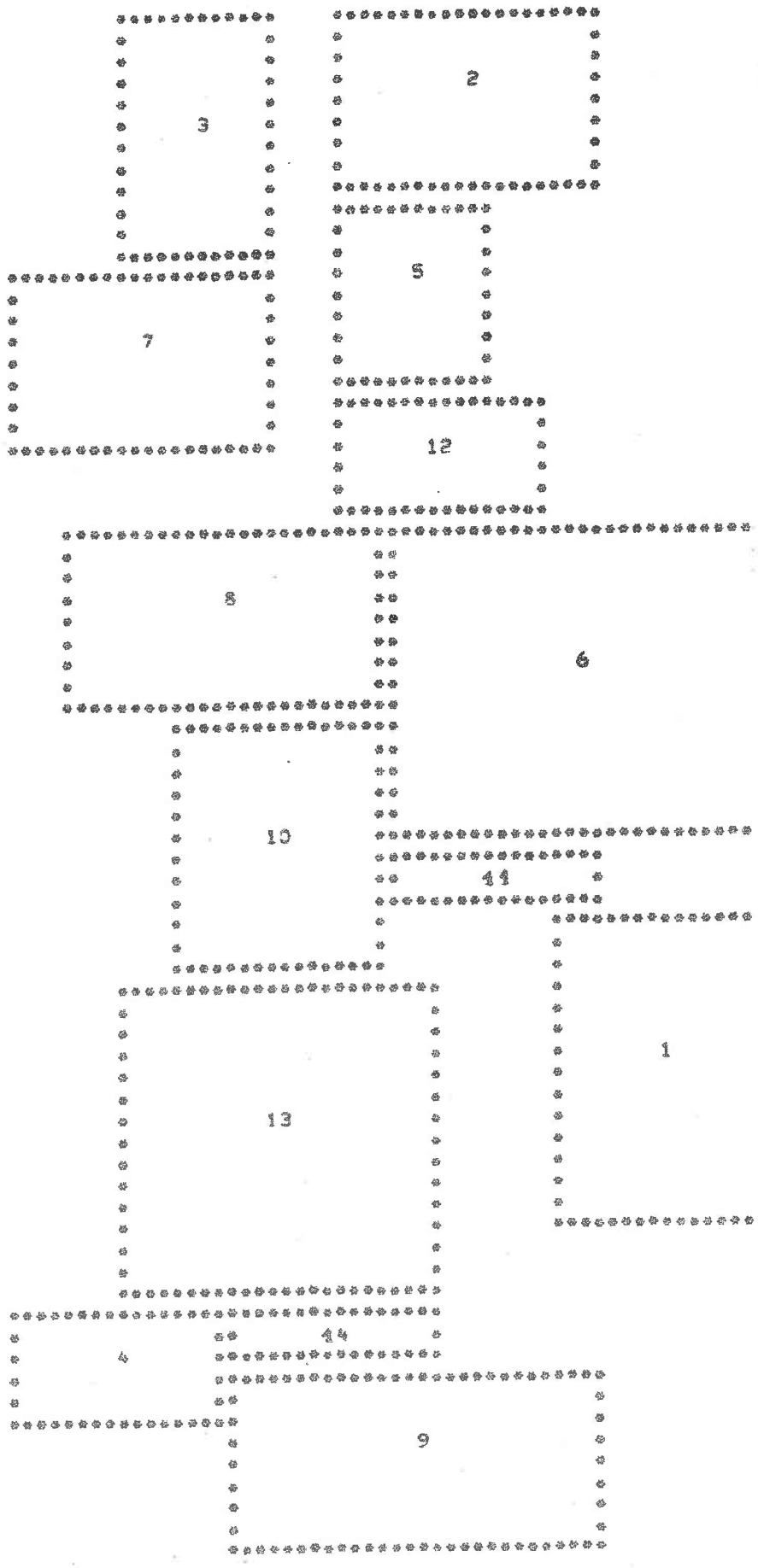
ANNEXE V

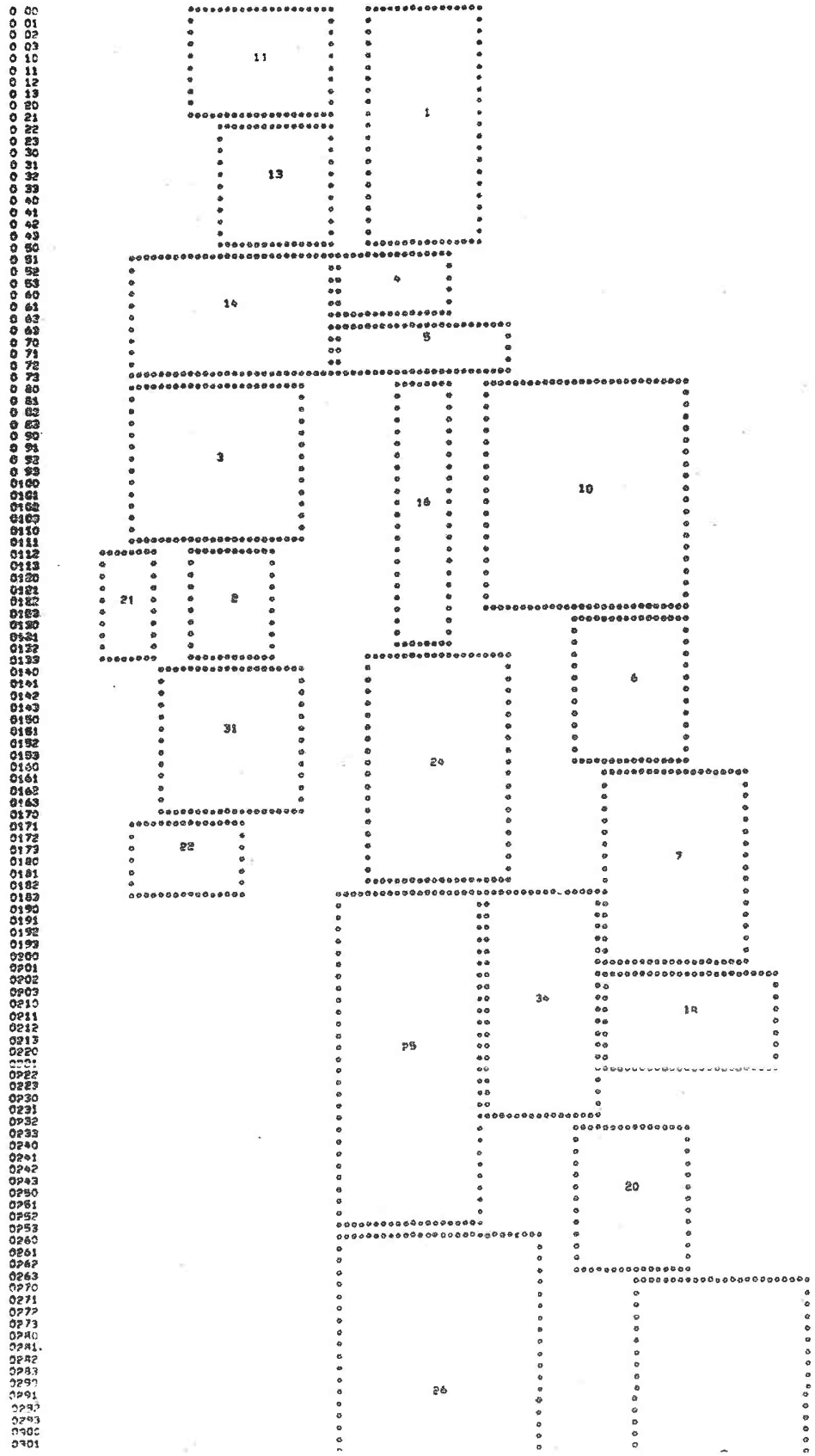
V-1 - "Visualisation" de deux processus générés par programme.

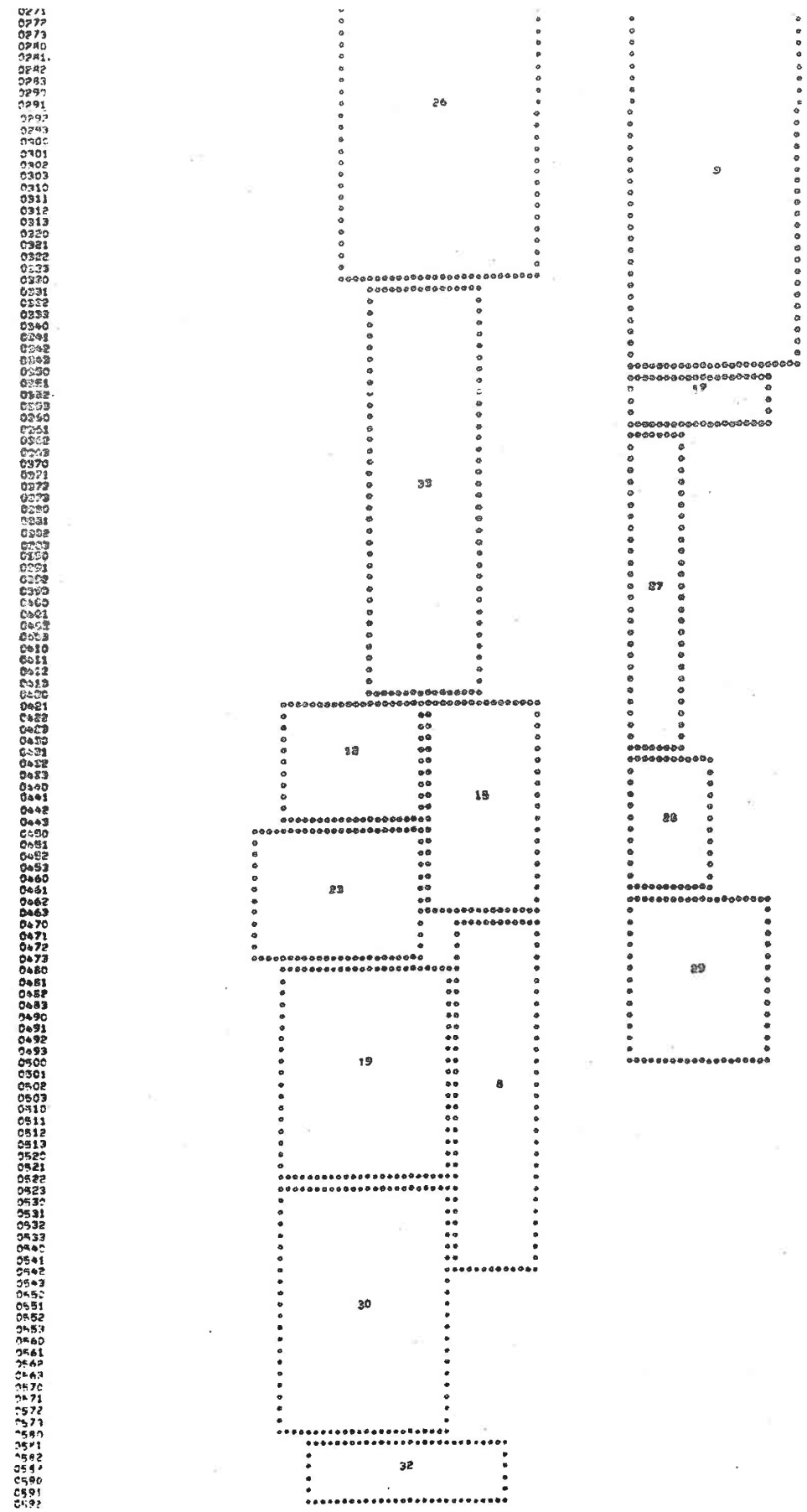
V-1 A) Nous avons fait généré par programme un processus à partir de l'ensemble des tâches qui avait servi d'exemple (chapitre V, page 50). Le processus obtenu réalise les deux objectifs "Règle de Sécurité" et "Temps minimum". Ce processus , légèrement différent de celui obtenu manuellement, prouve que l'optimum n'est pas unique.

N.B. Les chiffres qui se trouvent à gauche représente le temps suivant le code adopté 0143 = 0h 14' 45".

B) Nous présentons ensuite un exemple plus complexe. Le nombre de tâches à réaliser est de 34.


3
2
7
5
12
8
6
10
11
1
13
4
14
9


11
1
13
4
14
5
3
16
10
21
2
6
31
24
7
22
34
18
25
20
26


26
9
17
33
27
18
15
28
23
29
19
8
30
32

BIBLIOGRAPHIE

On aura intérêt à consulter au sujet de méthodes développées pour une aide à la Décision :

1 - ARMOUR G.C. and E.S. BUFFA

≪A Heuristic Algorithm and Simulation Approach to the Relative Location of Facilities≫

Management Science , Janvier 1963, page 294-309.

2 - BUFFA E.S. , G.C. ARMOUR and T.E. VOLLMAN

≪Allocating facilities with CRAFT≫

Harvard Business Review, Mars - Avril 1964.

3 - PROTH J.M.

≪Recherche de règles d'ordonnancement par traitement sur ordinateur≫

Thèse de Spécialité, Institut Universitaire de Calcul Automatique - Nancy Janvier 1971.

NOM DE L'ETUDIANT : LADURE Pierre

Nature de la thèse : Doctorat de Spécialité en Mathématiques Appliquées

Vu, Approuvé

et permis d'imprimer

NANCY, le 9 Mars 1979

Le Président du Conseil de l'Université de NANCY I

J.R. HELLUY