

UNIVERSITÉ DE NANCY I
FACULTÉ DE SCIENCES MATHÉMATIQUES

CENTRE DE RECHERCHE EN
INFORMATIQUE DE NANCY

Sc N 80 / 36 B

CONCEPTS ET LOGICIEL

D'ANALYSE STATISTIQUE

DANS LE PROJET MAESTRO

THÈSE

PRÉSENTÉE POUR L'OBTENTION DU
DOCTORAT DE 3ÈME CYCLE EN
INFORMATIQUE

SOUTENUE LE 13 JUIN 1980

PAR MICHÈLE THAUPEL

JURY :

PRÉSIDENT

EXAMINATEURS



D 095 097739 5

J. P. DUFORD
C. ROLLAND



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

CENTRE DE RECHERCHE EN
INFORMATIQUE DE NANCY

CONCEPTS ET LOGICIEL

D'ANALYSE STATISTIQUE

DANS LE PROJET MAESTRO



THÈSE

PRÉSENTÉE POUR L'OBTENTION DU
DOCTORAT DE 3ÈME CYCLE EN
INFORMATIQUE

SOUTENUE LE 13 JUIN 1980

PAR MICHÈLE THAUREL

JURY :

PRÉSIDENT C. PAIR
EXAMINATEURS : M. CREHANGE
J.F. DUFOURD
C. ROLLAND

La démarche de remercier avec des paroles ou des mots m'est toujours apparue difficile et embarrassante, tant il est facile d'être hypocrite et calculateur avec des mots et tant l'omission et l'injustice peuvent avoir un effet contraire à celui désiré.

Cependant, il m'est agréable de reconnaître que ce travail n'aurait jamais vu le jour sans tous ceux qui m'ont aidée, conseillée, critiquée jusqu'à présent et à qui je voudrais exprimer ma gratitude :

- ma famille pour son soutien de toujours et mes amis pour m'avoir soutenue et supportée patiemment,
- l'équipe MAESTRO sans qui ce travail n'aurait jamais existé,
- mes professeurs à qui je dois ma formation intellectuelle,
- les membres du jury qui acceptent aujourd'hui de juger ce travail,
- tous ceux qui ont participé à la réalisation matérielle de cette thèse.

SOMMAIRE

CHAPITRE I : INTRODUCTION

| | |
|--|-----|
| I - Objectifs du projet MAESTRO | I-1 |
| I-1. Définitions générales | I-1 |
| I-1.1. Système | I-1 |
| I-1.2. Système d'information | I-2 |
| I-1.3. Modèle | I-3 |
| I-1.4. Simulation de système d'information .. | I-3 |
| I-2. Objectifs de la modélisation de système d'in- formation dans le projet MAESTRO | I-4 |
| I-2.1. Description et évaluation structurale .. | I-4 |
| I-2.2. Evaluation fonctionnelle qualitative .. | I-5 |
| I-2.3. Evaluation quantitative du comportement dynamique | I-6 |
| I-2.4. Prédiction | I-6 |
| II - Notre étude | |

CHAPITRE II : CONCEPTS ET OUTILS UTILISES DANS LE PROJET MAESTRO

| | |
|--|-------|
| I - Architecture générale d'une maquette de système d'in- formation | |
| I-1. Point de vue logique | II-1 |
| I-1.1. Aspects asynchrones | II-1 |
| I-1.2. Aspects synchrones | II-6 |
| I-2. Point de vue physique | II-7 |
| I-2.1. Aspects asynchrones | II-7 |
| I-2.2. Aspects synchrones | II-9 |
| II - Réduction et simplification dans une maquette .. | II-10 |
| II-1. Point de vue logique | II-10 |
| II-1.1. Données | II-10 |
| II-1.2. Processus | II-11 |
| II-2. Point de vue physique | II-12 |
| II-2.1. Implantation des données | II-12 |
| II-2.2. Implantation des processus | II-12 |
| II-2.3. Ressources | II-12 |

| | |
|--|-------|
| III - Exemple de maquette | II-13 |
| III-1. Architecture générale | II-13 |
| III-2. Fonctionnement des composants | II-14 |

CHAPITRE III : LES MESURES ET LEUR ANALYSE STATISTIQUE

| | |
|---|--------|
| I - Processus stochastiques | III-2 |
| I-1. Stationnarité stricte | III-5 |
| I-2. Faible stationnarité | III-6 |
| I-3. Non stationnarité | III-6 |
| II - Estimateurs des moments des processus stochastiques | III-7 |
| II-1. Stationnarité stricte | III-7 |
| II-1.1. Moyenne | III-11 |
| II-1.2. Variance | III-11 |
| II-1.3. Autocovariance | III-11 |
| II-1.4. Autocorrélation | III-12 |
| II-1.5. Densité spectrale de puissance | III-13 |
| II-2. Stationnarité faible | III-16 |
| II-3. Non stationnarité | III-17 |
| III - Construction d'intervalles de confiance | III-19 |
| III-1. Plusieurs répliques indépendantes du processus | III-19 |
| III-2. Une seule réplique du processus | III-21 |
| III-2.1. Méthode utilisant les coefficients d'autocorrélation | III-21 |
| III-2.2. Méthode des blocs | III-22 |
| III-2.3. Méthode utilisant des cycles indépendants | III-25 |
| IV - Analyse des séries chronologiques | III-26 |
| IV-1. Définition | III-27 |
| IV-2. Généralités | III-27 |
| IV-3. Démarche d'analyse | III-29 |
| IV-4. Visualisation | III-30 |
| IV-4.1. Le graphe des données | III-30 |

| | |
|---|--------|
| IV-4.2. Le corrélogramme | III-31 |
| IV-4.3. Le périodogramme de Schuster | III-32 |
| IV-4.4. Le filtre de Buys-Ballot | III-34 |
| IV-4.5. Le spectrogramme | III-35 |
| IV-5. Recherche de caractéristiques sur le processus stochastique sous-jacent | III-37 |
| IV-5.1. Stationnarité | III-37 |
| IV-5.2. Existence de périodicité | III-41 |
| IV-5.3. Corrélacion sériale | III-42 |
| IV-6. Identification du processus stochastique sous-jacent | III-43 |
| IV-6.1. Variables aléatoires mutuellement indépendantes | III-44 |
| IV-6.2. Variables aléatoires dépendantes .. | III-48 |
| IV-7. Conclusion | III-55 |

CHAPITRE IV : METHODOLOGIE D'EVALUATION QUANTITATIVE DU COMPORTEMENT DYNAMIQUE

| | |
|--|-------|
| I - Objectifs | IV-1 |
| II - Méthodologie | IV-2 |
| II-1. Formulation du problème | IV-3 |
| II-2. Etablissement d'un modèle | IV-3 |
| II-2.1. Clôture du modèle | IV-4 |
| II-2.2. Paramétrage du modèle | IV-4 |
| II-2.3. Collecte des résultats | IV-6 |
| II-3. Collecte et analyse des mesures | IV-7 |
| II-3.1. Recueil des mesures sur le système réel | IV-7 |
| II-3.2. Synthèse des mesures | IV-8 |
| II-4. Analyse et génération des entrées | IV-9 |
| II-4.1. Entrée du modèle de simulation | IV-9 |
| II-4.2. Détermination de modèles statistiques de l'environnement | IV-11 |
| II-4.3. Génération de nombres suivant une loi donnée | IV-11 |

| | |
|--|-------|
| II-5. Analyse des sorties | IV-15 |
| II-5.1. Résultats souhaités | IV-15 |
| II-5.2. Mesures | IV-18 |
| II-5.3. Précision et signification des résultats | IV-29 |
| II-6. Validation | IV-33 |
| II-6.1. Validation par rapport à la réalité ou un autre modèle déjà validé | IV-34 |
| II-6.2. Validation par rapport à un modèle mathématique | IV-37 |
| II-7. Robustesse | IV-47 |
| II-8. Utilisation d'une maquette en prédiction .. | IV-48 |
| II-9. Conclusion | IV-49 |

CHAPITRE V : OUTILS DE MESURE ET D'ANALYSE DES ENTREES

| | |
|---|------|
| I - <u>Spécification fonctionnelles</u> | V-1 |
| I-1. Objectifs | V-1 |
| I-2. Démarche | V-1 |
| I-3. Caractéristiques générales de l'outil ... | V-1 |
| II - <u>Spécifications techniques</u> | V-2 |
| II-1. Visualisation | V-2 |
| II-2. Recherche de caractéristiques sur le processus stochastique sous-jacent | V-8 |
| II-2.1. Stationnarité | V-8 |
| II-2.2. Existence de composantes cycliques .. | V-8 |
| II-2.3. Dépendance temporelle | V-8 |
| II-3. Recherche de modèle probabiliste | V-14 |
| II-3.1. Choix d'un modèle | V-14 |
| II-3.2. Estimation des paramètres | V-15 |
| II-3.3. Vérification | V-16 |
| III - <u>Conclusion</u> | V-19 |

CHAPITRE VI : OUTILS DE MESURE ET D'ANALYSE DES SORTIES

| | |
|---|-------|
| I - <u>Spécifications fonctionnelles</u> | VI-1 |
| I-1. Objectifs | VI-1 |
| I-2. Démarche | VI-1 |
| I-3. Caractéristiques générales | VI-2 |
| I-3.1. Description des objets utilisés | VI-2 |
| I-3.2. Description et utilisation des outils . | VI-4 |
| II - <u>Spécifications techniques</u> | VI-5 |
| II-1. Caractéristiques techniques générales .. | VI-5 |
| II-1.1. Langage | VI-5 |
| II-1.2. Utilisation de fichiers | VI-7 |
| II-2. Description des objets spécifiques aux mesures | VI-8 |
| II-2.1. Les transactions | VI-8 |
| II-2.2. Les régions | VI-8 |
| II-2.3. Les structures de données | VI-10 |
| II-2.4. Les ressources | VI-15 |
| II-3. Processus prise de mesure | VI-17 |
| II-4. Exécution d'une simulation | VI-18 |
| II-5. Procédures relatives à l'obtention de résultats détaillés | VI-20 |
| II-6.- Obtention des résultats | VI-22 |
| II-7. Conclusion | VI-24 |

CONCLUSION

BIBLIOGRAPHIE

ANNEXE

CHAPITRE I

INTRODUCTION

Notre travail s'inscrit dans le cadre du projet MAESTRO (Maquettes pour évaluer les systèmes d'information des organisations).

Aussi, nous rappelons tout d'abord les objectifs de ce projet avant de présenter notre étude.

I - OBJECTIFS DU PROJET MAESTRO

L'objectif essentiel de ce projet (DUF 80a) est d'aider à décrire et analyser les systèmes d'information des organisations pour mieux comprendre leur structure, leur fonctionnement logique et leur comportement dynamique.

Pour avoir une description globale de tels systèmes, le projet MAESTRO propose de construire des modèles généraux programmés ou maquettes (logicielles).

Ces maquettes sont des modèles réduits de systèmes d'information et intègrent à un certain niveau plusieurs aspects :

- automatique et non-automatique,
- logique et physique,
- synchrone et asynchrone .

Suivant les différents niveaux de détail, fonctions des objectifs fixés, ces maquettes peuvent être évaluées du point de vue de leur (s) :

- structure,
- fonctionnement logique,
- performances.

Cette évaluation est obtenue en exécutant les maquettes selon une technique de simulation à événements discrets.

Avant de développer les différents objectifs de ce projet, il nous faut rappeler quelques définitions simples et suffisantes pour une bonne compréhension du travail en précisant ce que nous entendons par système, système d'information, modèle et simulation de système d'information.

I-1.- Définitions générales

I-1.1.- Systeme

Parmi les nombreuses définitions existantes, nous adoptons la suivante tirée de (DUF 80a) :

Un système est un ensemble d'objets identifiables (ou entités) qui sont reliés entre eux.

Chaque entité est caractérisée par des attributs ou des propriétés.

Il est possible de classer les liaisons existant entre entités selon deux catégories :

- statiques (indépendantes du temps),
- dynamiques (dépendant explicitement du temps),

et d'une certaine façon, les considérer comme des objets (abstraits).

L'espace d'états du système est l'ensemble des valeurs que peuvent prendre les entités et les liaisons. Cet espace d'états qui est fini (c'est le cas pour toute application réelle) peut être très complexe à étudier, on choisit alors certaines entités ou liaisons caractéristiques, en fonction des objectifs définis dans l'étude. Lorsque les changements d'états se produisent de manière discrète dans le temps, ils sont appelés événements et les instants où ils ont lieu sont appelés temps d'occurrence d'événement ou dates (d'événement).

L'environnement du système se définit comme tout ce qui est extérieur au système. Lorsqu'il peut influencer l'état du système, celui-ci est dit ouvert ; dans le cas contraire il est dit clos.

I-1.2.- Système d'information

Dans l'optique du projet MAESTRO, nous retenons la définition suivante (DUF 80a) :

Un système d'information d'organisation est un ensemble complexe d'informations et de moyens de traitement incluant : collecte, mémorisation, transformation, restitution, communication d'information.

L'étude du système d'information se fait à partir du "réel perçu". Cette vision subjective du système d'information permet de mettre en évidence un certain nombre d'entités et de liaisons entre ces entités. En fonction d'objectifs précis, il est possible de se restreindre à un nombre limité d'entités et de liaisons et de considérer par la suite leurs différents changements d'états.

Suivant le niveau de description, on est amené à distinguer les niveaux logique et physique en appréhendant chaque fois les aspects asynchrones (indépendants des dates et des traitements) et synchrones (liés explicitement au temps).

Au niveau logique, on s'intéresse aux données et aux transformations effectuées sur ces données. Au niveau physique, on considère les ressources physiques nécessaires à l'enregistrement des données et à leurs transformations. Ce sont des éléments qui influencent le fonctionnement dynamique du système d'information en introduisant la notion de durée d'exécution.

I-1.3.- Modèle

Pour étudier un système, il est intéressant d'en construire un modèle. On peut adopter la définition suivante citée dans (DUF 80a) :

Un modèle est la représentation simplifiée d'un système. Le modèle réduit doit fonctionner sur les mêmes principes que l'original en vraie grandeur.

Un modèle est donc une simplification et une réduction de la réalité, qui conserve les propriétés "les plus importantes", propriétés qui dépendent des objectifs poursuivis dans la modélisation.

La représentation d'un système par un modèle peut s'obtenir en prenant en compte un nombre plus ou moins important d'entités et de liaisons entre ces entités suivant le niveau de détail de la description et ce en fonction d'objectifs précis. Ainsi, à partir d'un même système plusieurs modèles peuvent être construits à différents niveaux de détail et selon des objectifs divers.

I-1.4.- Simulation de système d'information

La simulation d'un modèle consiste à remplacer son étude par celle d'un de ces modèles.

Si on suppose que le modèle construit reflète "correctement" la réalité, on peut alors faire confiance aux résultats obtenus lors de la simulation et les transposer au système étudié.

Le projet MAESTRO s'intéresse à la simulation de modèles programmés écrits dans un langage de simulation : SIMULA (DAH 70). Ceux-ci sont exécutés selon une technique de simulation à événements discrets (FIS 73, LER 77) par opposition à la simulation continue (FOR 61).

Au niveau de description où nous nous placerons, la simulation à événements discrets apparaît mieux appropriée : elle permet d'appréhender les phénomènes essentiellement discrets du traitement des données.

Après avoir exposé brièvement comment sont utilisés les concepts de système, modèle, et simulation dans le projet MAESTRO, nous précisons les objectifs de ce projet.

I-2.- Objectifs de la modélisation de systèmes d'information dans le projet MAESTRO (DUF 80a)

A partir du même système d'information, il est possible de construire de nombreuses maquettes, de différents niveaux et avec des caractéristiques diverses pour satisfaire des objectifs précis.

Comme une maquette est un modèle orienté vers l'analyse de systèmes d'information, elle peut être exploitée à différentes fins.

I-2.1.- Description et évaluation structurale

Une maquette logicielle constitue une description formelle de système d'information. Le modèle écrit dans un langage précis donne une description globale du système incluant les parties automatique et non-automatique.

Outre l'aspect statique d'une description (architecture des composants), il est possible d'appréhender l'aspect dynamique en décrivant les modifications des données ordonnancées dans le temps en tenant compte des phénomènes de parallélisme et de synchronisation.

Par ailleurs, une maquette reproduit les aspects logiques (description des données et des processus logiques de modifications de ces données) et physiques (en incluant dans la description l'utilisation des ressources physiques nécessaires pour une bonne exécution des processus réels) d'un système.

Cette description est modulaire : elle s'obtient par assemblage de composants standard diversifiés (KLE 79) et spécifiques au système décrit.

Ainsi, la conception et l'écriture de maquettes logicielles sont une aide dans la description d'un système d'information :

- existant en fournissant une documentation claire permettant de le comparer à d'autres systèmes.
- futur pour l'évaluer structurellement, fonctionnellement et guider ensuite sa conception détaillée, sa réalisation et son exploitation.

Après avoir écrit une maquette logicielle, il est important de savoir si la forme du système d'information décrit (existant ou futur) est "satisfaisante".

Comme ces maquettes sont écrites dans un langage de programmation, l'analyse syntaxique est déjà une forme simple de test formel de l'architecture du système, si le modèle est assez proche de la réalité.

I-2.2.- Evaluation fonctionnelle qualitative

L'écriture structurée d'une maquette avec des composants standard aide à détecter les vices de fonctionnement logique du système.

De façon très empirique, l'édition d'une trace au cours de l'exécution du programme maquette en quasi-parallélisme permet de porter une appréciation sur le comportement logique du système décrit (détection de blocages imprévus par exemple).

I-2.3.- Evaluation quantitative du comportement dynamique

L'exécution d'une maquette selon une technique de simulation à événements discrets peut fournir des renseignements quantitatifs sur l'évolution d'un système au cours du temps.

L'évaluation des performances du système décrit peut s'obtenir en effectuant des mesures sur certains composants du modèle en fonctionnement, lors de son exécution. Ces mesures sont ensuite analysées statistiquement et servent à déterminer certaines caractéristiques du comportement.

Si on estime le modèle décrit comme suffisamment "proche" de la réalité, on peut accorder une certaine confiance aux résultats obtenus, en les transposant au système décrit.

I-2.4.- Prédiction

Après avoir validé le modèle en s'assurant qu'il est capable de prendre en compte un ensemble de situations connues appelé domaine de validation, il peut être utilisé à des fins de prédiction. Pour cela, il faut supposer la robustesse du modèle en faisant l'hypothèse qu'il est capable de fournir des résultats en dehors du domaine de validation.

Suivant les hypothèses envisagées, on peut effectuer plusieurs types d'expériences en modifiant divers paramètres concernant :

- l'environnement
- le modèle (architecture et fonctionnement)

et ce avant ou pendant l'exécution de la maquette.

Les techniques utilisées sont identiques à celles requises en évaluation quantitative.

Dans (DUF 80a), les problèmes soulevés par ces divers types d'évaluation et les solutions apportées par les maquettes sont bien exposés et développés.

C'est dans ce contexte général que se situe notre travail.

II - NOTRE ETUDE

Dans le cadre du projet MAESTRO, nous nous sommes plus particulièrement intéressés à l'évaluation des performances d'un système d'information.

Suivant les hypothèses de fonctionnement, on est conduit à envisager plusieurs types d'analyses :

- en régime stationnaire. On vérifie que le système est "stable", "en équilibre" ("steady-state"). Dans (KLE 74), KLEIJNEN donne la définition suivante pour un tel état :

Le système est dit en régime stationnaire si la probabilité d'être dans cet état est fixe et indépendante du temps.

- en régime transitoire (par opposition à stationnaire).

En régime stationnaire, il est intéressant d'obtenir des grandeurs caractéristiques du comportement du système à l'équilibre. Ce sont géné-

ralement des valeurs moyennes permettant d'apprécier :

- le volume des données traitées,
- l'occupation des ressources,
- les temps de réponse de certaines parties.

Par contre en régime transitoire, on peut chercher à étudier le comportement du système lorsque se produisent des événements qui perturbent le fonctionnement "normal" du système.

Pour satisfaire cet objectif d'évaluation de performances, nous avons conçu divers outils de mesure et d'analyse qui sont intégrés dans le système logiciel proposé dans le projet MAESTRO (KLE 79, DUF 80a).

Après avoir rappelé les objectifs du projet MAESTRO, nous présentons dans le chapitre II les concepts et les notions qui guident la conception et l'écriture d'une maquette. Nous y donnons aussi un exemple de maquette auquel nous nous référerons par la suite pour illustrer nos propos et montrer comment fonctionnent nos outils.

Le chapitre III nous sert à exposer différentes techniques statistiques d'analyse de mesures (séries chronologiques, intervalles de confiance ...). Nous y rappelons définitions et résultats indispensables (BOX 70, AND 71, COX 69, KLE 74) à la bonne compréhension du reste de l'exposé.

Dans le chapitre IV, nous développons l'aspect méthodologie en appliquant les principes généraux concernant une étude de simulation à événements discrets (FIS 73, LER 77, KOB 78) à l'évaluation des systèmes d'information. Nous y abordons les problèmes soulevés par ce type d'approche et proposons quelques solutions qui permettent d'une part d'obtenir un modèle de simulation suffisamment "proche" du système étudié et efficace, et d'autre part d'en tirer des résultats significatifs.

Après avoir développé les aspects théoriques que comporte l'évaluation des performances par la simulation d'un modèle, nous présentons les aspects plus pratiques de réalisation d'outils mettant en oeuvre des méthodes exposées.

Le chapitre V est consacré à la présentation d'outils d'analyse statistique concernant l'étude de séries chronologiques et la recherche de

modèles probabilistes adéquats. Les outils conçus et réalisés mettent en oeuvre des techniques rappelées au chapitre III. Ils ont été utilisés lors du paramétrage de la maquette exemple. Celui-ci a été réalisé à partir de mesures réelles recueillies dans l'entreprise étudiée.

Dans le chapitre VI, nous donnons les caractéristiques générales des outils de mesure et d'analyse servant à interpréter les mesures et fournir des résultats sur les performances du système modélisé.

CHAPITRE II

CONCEPTS ET OUTILS UTILISÉS
DANS LE PROJET MAESTRO

I - ARCHITECTURE GENERALE D'UNE MAQUETTE DE SYSTEME D'INFORMATION

Une maquette est un modèle de système d'information (DUF 80a) :

Du point de vue structural, une maquette de système d'information peut être définie comme un système complètement formalisé de processus coopérants par l'intermédiaire de structures de données partageables, et concurrents pour l'obtention de ressources physiques .

Nous distinguons les deux points de vue : logique et physique.

I-1.- Point de vue logique

Au niveau logique, on considère les données et les transformations effectuées sur ces données.

Dans un modèle, on retrouve les aspects :

- asynchrones (indépendants des dates et des durées de traitement),
- synchrones (liés explicitement ou implicitement au temps).

I-1.1.- Aspects asynchrones

Une maquette est un système de processus coopérants avec données globales partageables.

Nous précisons successivement les différentes notions intervenant dans cette définition.

a) Structures de données globales

Pour concevoir une maquette, on peut raisonner statiquement en termes de types abstraits (GUT 78, DER 79, DUF 80b).

Un type abstrait peut être considéré comme un ensemble d'objets et d'opérations sur cet ensemble.

La spécification des types abstraits ne permet pas de décrire toutes les caractéristiques de la dynamique. C'est pourquoi, après le point de vue statique, il est nécessaire d'adopter un point de vue dynamique en considérant des structures de données (DUF 80b) :

Une structure de données \mathcal{S} est un couple $(\mathcal{D}, \mathcal{M})$ où \mathcal{D} est un ensemble de données et \mathcal{M} un ensemble de modifications sur ces données.

Dans une maquette, ces structures de données sont utilisées pour modifier les données communes partageables (fichiers manuels ou informatiques, base de données) du système réel.

Les modifications sur ces structures sont exécutées par des processus qui sont autorisés à appeler certaines fonctions ou procédures locales à ces structures.

Exemple

Considérons la structure de données carnet de COMMANDES avec un ensemble de fonctions qui sont divisées en deux catégories :

- Accès

- *premier* : délivre la première commande, si elle existe.
- *suisant (c)* : délivre la commande suivante de *c*, si elle existe.

- Modifications

- *ajouter (c)* : modifie le carnet en ajoutant la commande *c*.
- *supprimer (c)* : supprime la commande *c* du carnet.
- *modifier (c,v)* : modifie la commande *c* en lui donnant une nouvelle valeur d'attribut *v*.
- *enlever (c)* : délivre la première commande du carnet et la supprime, si elle existe.
- *créer* : crée un nouveau carnet de commandes.

b) Protection et synchronisation

Diverses solutions ont été proposées pour décrire la protection des données et la synchronisation des processus. On peut citer, entre autres, les sémaphores, les réseaux de Pétri, les moniteurs ... C'est ce dernier mécanisme de moniteur au sens de HOARE (HOA 74) qui a été retenu dans ce projet.

Un moniteur se compose :

- d'une structure de données locale.

- de (fonctions -) procédures d'accès ou de modification de ces données, qui sont appelées de l'extérieur par des processus.

Par ailleurs, ce mécanisme de protection des données est bien adapté à la synchronisation des processus. Suivant l'état des données, il permet de bloquer ou débloquer des processus, grâce aux conditions. Une condition est une file d'attente déclarée à l'intérieur du moniteur dans laquelle peuvent être bloqués des processus. Les modifications sur une condition se font par l'intermédiaire de procédures qui sont appelées dans le corps de (fonctions -) procédures externes au moniteur :

- l'appel de wait bloque le processus appelant dans la file d'attente,
- l'appel de signal libère immédiatement un des processus bloqués dans la file.

Le contrôle des accès à une structure de données peut se faire de différentes façons :

- si tous les accès ou modifications sur la structure doivent se faire en exclusion mutuelle totale, le mécanisme d'exclusion des moniteurs peut être utilisé pour "englober" la structure de données. On considère alors que la structure de données est incluse dans le moniteur.

Exemple

On peut vouloir que tous les appels des fonctions-procédures concernant la structure carnet se fassent en exclusion mutuelle. Dans ce cas, la structure est alors confondue avec un moniteur dont une condition "remplie" permet d'assurer le blocage ou la libération de processus appelant la fonction-procédure enlever :

- l'appel de wait provoque un blocage de ce processus sur la condition "remplie" lorsque le carnet est vide.
- l'appel de signal, précédé par un appel de ajouter entraîne la libération d'un des processus bloqués sur "remplie".

- si parmi les accès ou modifications, certains doivent se faire en exclusion mutuelle, on distingue alors bien nettement structure de données et moniteur protecteur. Toute tentative d'accès ou de modification doit alors être précédée par la consultation du moniteur qui donne l'au-

torisation ou bloque le processus appelant. Lorsque le processus a fini de travailler sur la structure, le moniteur est de nouveau consulté, pour libérer d'éventuels processus en attente.

Exemple

On peut considérer le problème classique des lecteurs-écrivains : on a un ensemble de processus lecteurs et écrivains qui se partagent un tampon. Comme les lecteurs ne font que lire dans le tampon, ils peuvent y accéder simultanément. Par contre, les écrivains modifiant le contenu ne peuvent y avoir accès qu'en exclusion mutuelle. Dans (HOA 74), HOARE propose une solution en utilisant le mécanisme de moniteur en considérant le tampon en dehors du moniteur.

c) Processus

Un processus est composé d'une structure de données propre à laquelle il est le seul à pouvoir accéder directement et d'un programme séquentiel.

Il peut opérer sur ses données propres et sur les données communes (éventuellement protégées par des moniteurs), mais jamais sur les données privées d'un autre processus.

Dans une maquette, les processus sont utilisés pour modéliser tous les traitements de données automatiques, manuels ou décisionnels. Très fréquemment, les traitements rencontrés en organisation sont répétitifs. Pour des raisons de simplicité, on les modélise par des processus cycliques : ils répètent indéfiniment la même séquence d'instructions et se bloquent de temps à autre dans les moniteurs. Cependant, il n'est pas interdit de générer et détruire dynamiquement des processus dans une maquette.

A un instant donné, l'état d'un processus est soit actif, soit passif (ou bloqué).

Un processus bloqué peut être non terminé : le programme séquentiel du processus est interruptible, avec reprise au point d'interruption.

d) Communication entre processus

La communication entre processus s'effectue par la transmission de messages qui peuvent être porteurs d'information utile ou n'être que des signaux de synchronisation.

Parmi les différents modes de communication, on peut distinguer (QUE 79) :

- la communication directe ou indirecte avec ou sans mémorisation possible de messages,
- la diffusion avec ou sans mémorisation.

Dans une maquette, la communication entre processus est assurée par un interface standard "le moniteur d'activation".

Jusqu'à présent, trois types de moniteurs d'activation ont été définis. Ils permettent la communication indirecte :

- avec mémorisation d'au plus un message s'il existe des processus en attente,
- avec mémorisation d'au plus un message, indépendamment de l'existence ou non de processus en attente de message,
- avec mémorisation de tous les messages.

Lorsque deux processus communiquent directement par un canal particulier, ce dernier est modélisé par un moniteur d'activation.

Exemple

Pour illustrer ce qui vient d'être dit, nous montrons sur la figure 2.1 une structure de données bons qui est partagée en exclusion mutuelle par deux processus : le processus décision y inscrit des ordres de livraison qui sont utilisés par le processus préparation selon un schéma classique de producteur-consommateur cyclique sur un tampon de taille infinie.

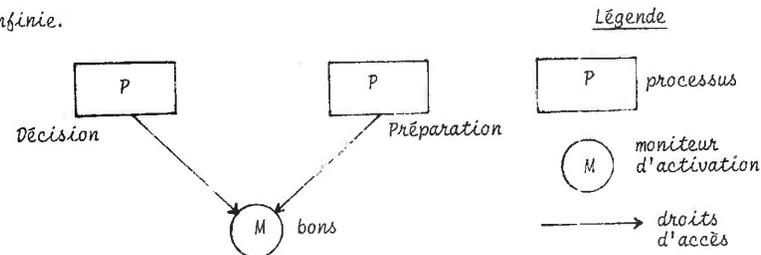


Fig. 2.1 : Exemple de moniteur d'activation
Partage de structure

On peut considérer que la structure bons est confondue avec un moniteur d'activation qui enregistre tous les messages et bloque les consommateurs si le tampon est vide.

I-1.2.- Aspects synchrones

Dans une maquette, le temps intervient explicitement, mais de manière simulée.

Suivant le type de processus (déterministe ou stochastique), on insère dans le programme séquentiel du processus des suspensions dont la durée (aléatoire ou fixe) simulée est fonction des traitements modélisés.

En gestion, il est courant qu'un processus soit activé à certaines dates. Dans une maquette, le concept d'échancier secondaire (par rapport à l'échancier standard du langage SIMULA (DAH 68)) permet de prendre en compte cette notion (DUF 80a) :

Conceptuellement, un échancier secondaire peut se représenter par une liste de couples (dates d'activation, nom du moniteur d'activation) rangés par dates d'activation croissantes et un processus scrutateur de cette liste.

Lorsque le temps courant simulé est égal à une date d'activation, le processus scrutateur envoie un message vers le moniteur associé, ce qui permet de libérer un processus en attente (s'il en existe).

Exemple

La figure 2.2 illustre ce mécanisme d'échancier : le processus décision peut être libéré par l'intermédiaire du moniteur activation à certaines dates précisées dans l'échancier ech.

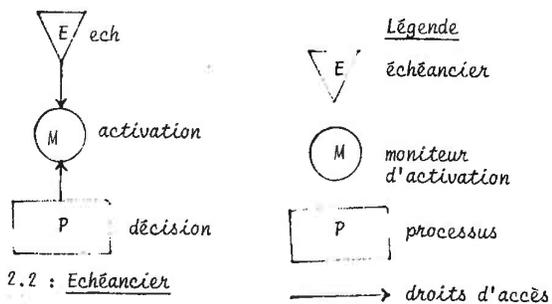


Fig. 2.2 : Echancier

I-2.- Point de vue physique

Maintenant, nous nous plaçons au niveau physique pour montrer comment la prise en compte des ressources physiques peut occasionner de nouveaux blocages. A ce niveau, on introduit la notion de durée d'exécution d'un processus. Du fait de l'existence en nombre limité de certaines ressources physiques, il y a possibilité de concurrence entre les divers processus requérants.

De nouveau, nous distinguons les aspects asynchrones et synchrones.

I-2.1.- Aspects asynchrones

Dans une maquette, les processus sont concurrents pour obtenir les ressources.

Nous considérons successivement les différents types de ressources physiques existant dans une maquette, leur modélisation et la description qui en est faite.

a) Types de ressources

Dans une maquette, il est possible de représenter les différentes ressources du système réel, qu'elles soient matérielles, logicielles ou humaines.

On peut les classer en deux catégories (BOD 79) :

- réutilisables : le processus requérant se bloque si la ressource est absente ou indisponible.
- consommables : elles modélisent des moyens auxiliaires requis par les processus pour s'exécuter ; elles figurent uniquement pour établir une sorte de comptabilité sur ces moyens (on fait alors l'hypothèse qu'ils sont toujours disponibles).

Dans une maquette, on ne tient compte que des ressources réutilisables dont l'influence est plus forte sur la dynamique du système que celle des ressources consommables. Ainsi, il est possible de considérer globalement un ordinateur comme une ressource réutilisable non bloquante.

b) Modélisation

Dans une maquette, les ressources sont toujours gérées indé-

pendamment les unes des autres de manière décentralisée (il n'y a pas d'allocateur unique).

Elles sont toujours modélisées par des stations de service (par analogie à la terminologie des files d'attente : les ressources sont des serveurs requis par les processus associés aux clients). Les stations peuvent être de différents types :

- file d'attente à un serveur (ou ressource critique), avec ou sans priorité, préemptible ou non ;
- file d'attente à plusieurs serveurs (ou ressource à accès multiples) avec ou sans priorité, préemptible ou non.

....

Différentes politiques de gestion sont envisageables : Premier-Arrivé-Premier-Servi, Dernier-Arrivé-Premier-Servi, processeur partagé, selon un "tourniquet"...

c) Description

Etant donné qu'il est possible de considérer les données partageables comme des ressources, ceci nous amène à utiliser le même mécanisme pour l'allocation et la protection : le moniteur.

Remarquons que la notion classique de moniteur ne permet pas de prendre en compte des politiques de gestion préemptives. Pour pallier cette difficulté, des conditions particulières "avec priorité et préemption" ont été définies (KLE 79).

Cette extension des moniteurs assure la gestion des différents types de ressources présentées ci-avant. Par ailleurs, l'utilisation de ce seul mécanisme (le moniteur) préserve une unité des concepts de description.

Exemple

La figure 2.3 illustre ce mécanisme : pour s'exécuter le processus saisie requiert deux ressources :

- l'ordinateur (ord) modélisé par une ressource à nombre infini d'accès, et représenté uniquement à des fins de comptabilité ;
- une personne (y) modélisée par une ressource critique préemptible (le processus saisie possède une priorité).

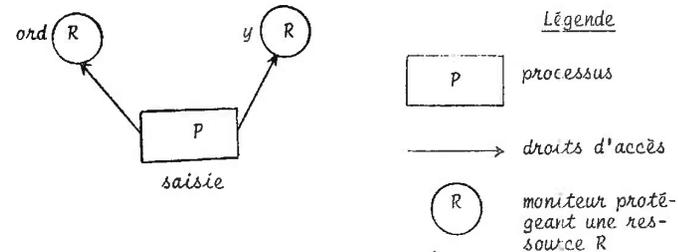


Fig. 2.3 : Utilisation de stations de service

I-2.2.- Aspects synchrones

Dans une maquette, les différentes ressources représentées ne sont pas toujours disponibles, elles peuvent l'être uniquement pendant certaines périodes. En dehors de ces intervalles, un processus requérant ces ressources est bloqué.

Pour tenir compte de la disponibilité d'une station pendant certains intervalles de temps, on utilise alors un calendrier (DUF 80a) :

Un calendrier attaché à une station se constitue d'une liste de couples (dates de début et fin de périodes de disponibilité) rangés par ordre croissant des dates de début et de deux processus scrutateurs de cette liste.

A chaque début de période, un processus "activateur" rend la ressource disponible, tandis qu'en fin, un processus "désactiveur" bloque l'usage de cette ressource.

Exemple

Si on veut associer à la ressource y de l'exemple précédent un calendrier c qui contient ses périodes de disponibilité, on utilise le graphisme suivant (fig 2.4).

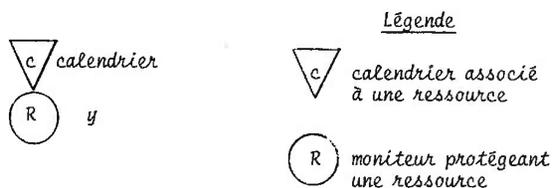


Fig. 2.4 : Calendrier associé à une station

En fin de période de disponibilité, l'utilisateur peut laisser terminer ou non le processus en cours d'exécution.

Une maquette étant un modèle réduit et simplifié de la réalité, il nous paraît important d'exposer plus précisément comment interviennent ces notions de réduction et simplification dans une maquette.

II - REDUCTION ET SIMPLIFICATION DANS UNE MAQUETTE

Nous examinons successivement les points de vue logique et physique

II-1.- Point de vue logique

Au niveau logique, nous considérons uniquement les données et les processus.

II-1.1.- Données

La réduction sémantique des données porte sur le choix de certaines données parmi celles mises en évidence dans le système réel en fonction d'objectifs précis parce qu'elles caractérisent "le mieux" certains aspects du réel qu'on désire étudier ou prendre en compte dans l'étude du système.

Entre la structure des objets réels et la structure de données d'une maquette, il n'y a généralement pas correspondance bijective.

Dans une maquette, la réduction sémantique des données se traduit par :

- le choix exclusif d'un certain nombre de structures de données et de fonctions d'accès et de modifications sur ces structures.
- la diminution du nombre d'attributs servant à caractériser les

données, (conservation des attributs les plus significatifs par rapport aux objectifs fixés).

- l'agrégation ou l'éclatement de certaines structures de données correspondant à des ensembles d'objets (les données) mis en évidence dans le système réel et vérifiant certaines propriétés.

La réduction sémantique des données diminue généralement le volume des données à traiter. Lorsque ces dernières sont en nombre encore trop important, il est nécessaire de procéder à une réduction quantitative supplémentaire. Pour respecter la dynamique du système, il faut alors en tenir compte lors de l'appréhension des traitements.

II-1.2.- Processus

La réduction sémantique des données induit une réduction et une simplification des traitements, qu'on peut appeler réduction sémantique des traitements. Elle peut réduire le nombre de processus, notamment lorsque les données sur lesquelles ils travaillent ont disparues, ce qui risque de supprimer certaines causes de blocage (suppression de processus consommateurs de ressources).

La réduction sémantique des traitements se traduit aussi en considérant uniquement leurs durées d'exécution. Très souvent, les instances de programmes informatiques sont représentées par des processus déterministes, tandis que les traitements non-automatiques peuvent l'être par des processus déterministes ou stochastiques.

Dans une maquette la réduction sémantique des traitements se traduit par :

- la considération unique des durées d'exécution,
- l'existence de processus "parasites", consommateurs de ressources uniquement (parce que les données sur lesquelles ils travaillaient ont disparu).
- l'agrégation de certains processus modélisant des traitements élémentaires (ce qui est fonction du niveau de détail de la description) dont le comportement dynamique est modifié en augmentant les consommations de ressources.

II-2.- Point de vue physique

Au niveau physique, les ressources physiques nécessaires à la collecte, à l'enregistrement et à la transformation des données sont prises en compte. Il en est de même pour les durées d'exécution des processus.

II-2.1.- Implantation des données

Vu l'échelle de temps généralement considérée nous ne nous préoccupons pas de l'implantation physique des données. Par contre, il est important que soient traduits en termes physiques l'organisation logique des données et l'évolution de ces dernières dans le temps, éléments qui sont très souvent indépendants des structures physiques réelles.

II-2.2.- Implantation des processus

Comme pour les données, les préoccupations essentielles concernent la traduction en termes physiques de leur logique et de leur comportement dynamique, en considérant les possibilités de blocage dues au niveau logique et à l'utilisation des ressources physiques.

II-2.3.- Ressources

Dans une maquette, nous considérons uniquement les ressources physiques nécessaires pour rendre compte du fonctionnement dynamique. Ceci se traduit par :

- l'ignorance de la plupart des ressources passives qui sont des moyens auxiliaires et la prise en compte des ressources réutilisables,
- le choix d'un certain nombre de types de ressources dans la modélisation,
- l'agrégation de ressources élémentaires simples (ce qui est fonction du niveau de détail de la description).

Même si ce n'est pas le cas dans la réalité, les ressources sont gérées indépendamment les unes des autres de manière décentralisée. Ce type d'allocation peut provoquer un inter-blocage. Pour éviter ce problème,

est recommandé de ne faire utiliser par un processus qu'une seule ressource réutilisable bloquante (DUF 80a).

III - EXEMPLE DE MAQUETTE

Pour illustrer les notions développées précédemment, nous présentons la structure d'une petite maquette de système d'information (fig. 2.5). Cette maquette est une des modélisations possibles d'un système de gestion de commandes. Elle correspond à la gestion des commandes-clients d'une entreprise commerciale que nous avons étudiée. Dans la thèse de P. KLEIN (KLE 79), on trouve une maquette plus complète qui l'englobe.

III-1.- Architecture générale

Le système décrit comprend :

- la prise de commandes écrites et téléphonées,
- le traitement administratif,
- le traitement physique.

Il se compose de quatre processus :

Enregistrement : processus exécuté par une personne x (pour la frappe) et l'ordinateur ord (partie automatique du traitement). Il codifie et saisit les bons de commande issus de la structure courrier pour mettre à jour la structure commandes. Il est déclenché tous les jours à une heure précisée dans l'échéancier ech, par l'intermédiaire du moniteur d'activation activation 1.

Saisie : processus de saisie des appels téléphoniques (demandes de renseignements et commandes) qui arrivent dans la structure boîte. Il est exécuté par une personne y qui saisit ces commandes en mode conversationnel (utilisation de ord). Lorsqu'il y a commande, celle-ci est codifiée, saisie, puis envoyée dans la structure commandes. Ce processus est déclenché aléatoirement lors de chaque appel téléphonique.

Décision : processus d'obtention des bons de livraison. Il est déclenché journalièrement à une heure précisée dans l'échéancier ech grâce au moniteur d'activation activation 2. Après une édition automatique des propositions de livraison par l'ordinateur ord, la personne y examine les propositions

éditées, fait des modifications que z saisit après en mode conversationnel. Ensuite, c'est l'édition définitive des bons de livraison. Ce processus alimente la structure ordres à partir de la structure commandes qu'il met à jour.

Préparation : processus de préparation des bons de livraison issus de la structure ordres. Il est exécuté par une personne p qui prépare les produits, complète les bons avant de les scinder en deux parties, l'une pour le client, l'autre pour la facturation ultérieure. Il alimente les structures bons 1 et bons 2. Il est activé par l'intermédiaire de la structure de données ordres qui est englobée par un moniteur.

Si on veut utiliser cette maquette en simulation, il est nécessaire de la clôre. Pour modéliser les relations du modèle avec son environnement, quatre processus ont été adjoints, deux en entrée et deux en sortie.

Générateur : processus générant des bons de commande. Il alimente la structure courrier. Il est déclenché tous les jours à une heure précisée dans l'échéancier ech, par l'intermédiaire du moniteur d'activation activation 5.

Généte] : processus générateur des appels téléphoniques. Il alimente la structure boîte. Il est déclenché aléatoirement.

Videur_1 : processus de vidage de la structure bons 1. Il modélise l'envoi des bons de livraison aux clients. Il est déclenché tous les jours à une heure précisée dans l'échéancier ech par l'intermédiaire du moniteur d'activation activation 3.

Videur_2 : processus analogue à videur 1. Il permet de vider tous les jours la structure bons 2 et correspond à l'envoi des bons de livraison à la comptabilité. Il est déclenché à une heure contenue dans l'échéancier ech par l'intermédiaire du moniteur d'activation activation 4.

III-2.- Fonctionnement des composants

Tous les processus décrits sont cycliques.

Ils sont activés de deux façons :

- soit à une date fixe, précisée dans l'échéancier ech par l'intermédiaire d'un moniteur d'activation,

- soit par l'intermédiaire d'une structure de données protégée par un moniteur, c'est le cas pour le processus préparation qui se bloque uniquement lorsque la structure de données ordres est vide (schéma classique de producteur-consommateur).

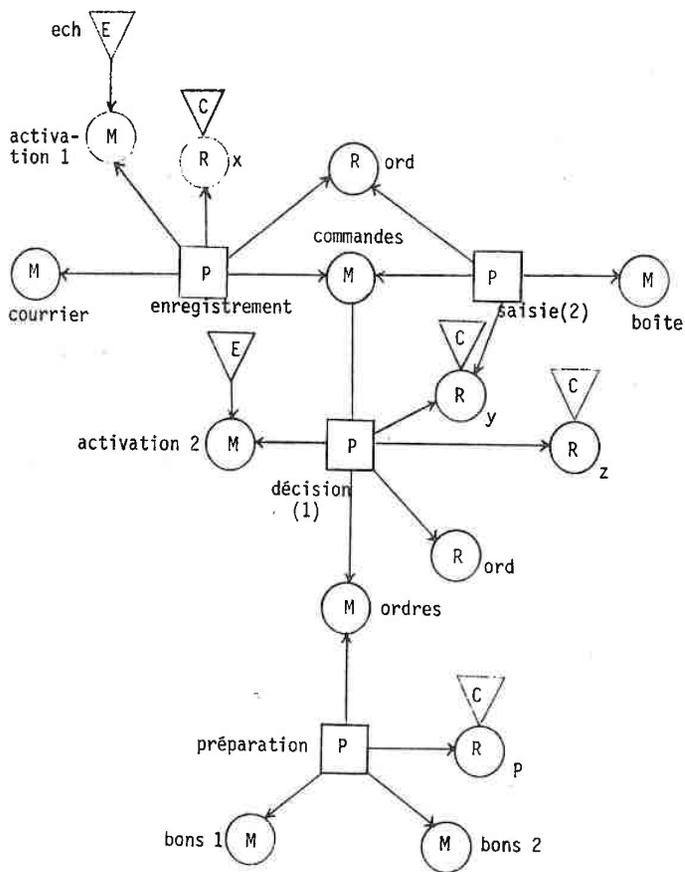
Les diverses structures de données présentes dans la maquette sont des files d'attente dont la politique de gestion peut être complexe. Elles sont incluses dans un moniteur. Ainsi, tout accès ou modification sur les données de la structure s'effectue en exclusion mutuelle.

Pour éviter les risques de blocage, l'ordinateur ord est considéré comme une ressource à nombre infini d'accès, il est représenté uniquement à des fins de comptabilité.

Les personnes x, z, p sont modélisées par des ressources critiques non préemptibles ayant chacune un calendrier contenant leurs périodes de disponibilité. La personne y est modélisée par une ressource critique préemptible avec priorité. Elle peut être requise par deux processus : saisie et décision dont les priorités sont respectivement 2 et 1. Pour s'exécuter, le processus saisie peut préempter la ressource y lorsqu'elle est utilisée par le processus décision : ceci correspond à la réponse immédiate à un appel téléphonique.

Les principales caractéristiques concernant ces composants sont résumées dans deux tableaux (fig. 2.6 et fig. 2.7).

Dans (DUF 80a), on trouvera une présentation complète.



LEGENDE

- S Structure de données
- M moniteur protégeant une structure de données
- P processus
- droit d'accès d'un processus à une structure de données ou à un moniteur
- E échéancier (secondaire)
- R moniteur protégeant une ressource physique
- C calendrier associé à une ressource physique

Fig. 2.5 : Architecture de maquette exemple

| PROCESSUS | Moniteur associé | Mode d'activation | Ressources utilisées | Priorités |
|----------------|------------------|---|----------------------|-----------|
| générateur | activation 5 | échéancier à 9 h 30 | | 1 |
| genetel | | processus de Poisson de moyenne λ quand y travaille | (y) | 1 |
| enregistrement | activation 1 | échéancier à 10 h | x, ord | 1 |
| saisie | boîte | genetel | y, ord | 2 |
| décision | activation 2 | échéancier à 15 h | y,z, ord | 1 |
| préparation | ordres | décision | P | 1 |
| videur 1 | activation 3 | échéancier à 20 h | | 1 |
| videur 2 | activation 4 | échéancier à 20 h | | 1 |

Fig. 2.6 : Tableau donnant les caractéristiques des processus

| NOM | TYPE de ressource | Nombre d'accès | Option laissant terminer le travail en cours | Heures de disponibilité |
|-----|-------------------------------------|----------------|--|-------------------------|
| x | critique | 1 | NON | 10h à 12h ; 14h à 15h |
| y | critique, préemptible avec priorité | 1 | NON | 8h à 12h ; 14h à 18h |
| z | critique | 1 | OUI | 15h30 à 17h |
| P | critique | 1 | NON | 8h à 12h ; 14h à 18h |
| ord | à nombre infini d'accès | ∞ | - | 0h à 24h |

Fig. 2.7 : Tableau donnant les caractéristiques des ressources

CHAPITRE III

LES MESURES ET LEUR
ANALYSE STATISTIQUE

L'évaluation du comportement dynamique d'un système d'information par simulation comporte différentes étapes qui sont rappelées dans la figure 4.1 et qui font souvent appel aux techniques statistiques notamment pour :

- faire la synthèse des mesures effectuées sur le système réel (paramétrage du modèle),
- obtenir des résultats de la simulation proprement dite (estimation de grandeurs caractéristiques, calcul d'intervalles de confiance),
- comparer les sorties du système réel avec celles du modèle simulé, en fonction d'un environnement donné (validation).

Pour résoudre ces questions, on peut classer les techniques statistiques utilisées en deux catégories :

- statistique descriptive

Elle est utilisée pour faire la synthèse des données brutes issues des mesures. Les techniques employées peuvent être élémentaires (construction d'histogramme, estimation de moyenne et d'écart-type ...) ou complexes (analyse de données multidimensionnelles, classification automatique ...).

- statistique explicative

L'interprétation des données est basée sur une représentation en termes probabilistes de la réalité, ce qui nécessite le choix d'un modèle et la détermination des valeurs de paramètres (estimation, ajustement).

Dans ce chapitre, nous rappelons définitions et résultats indispensables à la compréhension de ce travail. Les mesures que nous avons à analyser, que ce soit en entrée ou en sortie constituent habituellement des séries chronologiques (suites d'observations généralement ordonnées dans le temps), qu'on peut étudier en utilisant les propriétés des processus stochastiques sous-jacents.

La littérature est très abondante sur les processus stochastiques. Parmi les différents travaux, on peut citer ceux de BARTLETT (BAR 56), HANNAN (HAN 60), COX et LEWIS (COX 69), BOX et JENKINS (BOX 70). La présentation que nous faisons maintenant s'appuie sur ces ouvrages.

I - PROCESSUS STOCHASTIQUES

Un processus stochastique X est une famille de variables aléatoires

$$\{X(t), t \in T\}$$

T étant l'espace temps, T peut être un intervalle de \mathbb{R} ou un ensemble de points (sous-ensemble de \mathbb{N}).

On peut aussi définir X par une application réelle :

$$X : T \times \Omega \longrightarrow \mathbb{R}$$

$$(t, \omega) \longmapsto X(t, \omega)$$

Ainsi $X(t, \omega)$ est une réalisation du processus stochastique où t est un élément de l'espace temps T et ω un élément de l'espace des épreuves Ω .

- si $t \in \mathbb{N}$, le processus stochastique est discret,
- si $t \in \mathbb{R}$, le processus stochastique est continu.

En supposant que X est défini pour tout couple (t, ω) , il est possible d'échantillonner le processus de deux manières :

- t fixé : on fait varier ω en faisant un échantillonnage sur l'espace des épreuves Ω ,
- ω fixé : on fait varier t en réalisant un échantillonnage dans le temps, $X(t, \omega)$ représente alors une trajectoire (réplique) du processus.

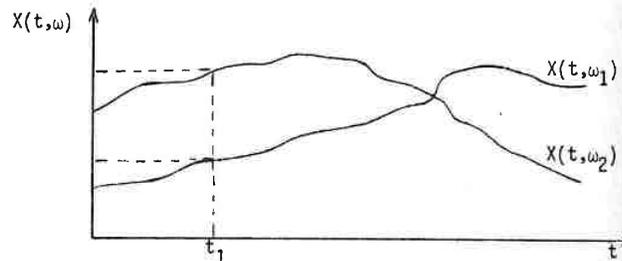


Fig. 3.1 : Trajectoire d'un processus stochastique continu

Après avoir défini les processus stochastiques, nous donnons quelques propriétés en nous référant essentiellement aux travaux de (HAN 60, AND 71).

Nous notons indifféremment l'espace temps T ou \mathbb{R} en nous plaçant dans le cas où nous disposons d'une seule trajectoire.

Le processus stochastique X est parfaitement défini si pour tout m -uplet d'instants t de T^m , (t_1, t_2, \dots, t_m) , la fonction de répartition conjointe est connue :

$$F(t_1, t_2, \dots, t_m; x_1, x_2, \dots, x_m) = P[X(t_1) < x_1, X(t_2) < x_2, \dots, X(t_m) < x_m]$$

Comme il est difficile de connaître cette fonction de répartition F , dans la pratique on se contente généralement de la fonction de répartition absolue :

$$F(t; x) = P[X(t) < x]$$

Très souvent, dans l'analyse des séries chronologiques, il est difficile d'identifier les processus stochastiques sous-jacents. On s'intéresse alors à différentes propriétés statistiques telles que :

- moyenne : $E[X(t)] = u(t)$
- variance : $\text{Var}[X(t)] = E[(X(t) - u(t))^2] = \sigma^2(t)$
- autocovariance : $\text{Cov}[X(t), X(t+\tau)] = E[(X(t) - u(t)) \cdot (X(t+\tau) - u(t+\tau))]$
 $= \gamma(t, \tau)$
- coefficient d'autocorrélation : $\rho(t, \tau) = \frac{\gamma(t, \tau)}{\sigma(t)\sigma(t+\tau)}$

Ces deux dernières propriétés ne peuvent s'étudier facilement que lorsque le processus est supposé faiblement stationnaire, c'est-à-dire qu'il vérifie les deux propriétés suivantes :

$$\begin{cases} - u(t) = u & \forall t \in T \\ - \gamma(t_1, \tau) = \gamma(t_2, \tau) = \gamma(\tau) & \forall (t_1, t_2) \in T^2 \quad \forall \tau \in T \end{cases}$$

Dans certaines conditions, il est possible de définir la transformée

de Fourier de la fonction d'autocovariance γ (cas continu) ou de la suite des autocovariances $\{\gamma(k), k \in \mathbb{N}\}$, appelée densité spectrale f .

Nous distinguons deux cas : la paramètre temps t varie de manière discrète ou non.

Si le processus stochastique X est discret, la transformée de Fourier de la suite d'autocovariance $\{\gamma(k), k \in \mathbb{N}\}$ s'écrit :

$$f(\lambda) = \frac{1}{2\pi} [\gamma(0) + 2 \sum_{k=1}^{\infty} \gamma(k) \cos \lambda k] \quad \lambda \in [-\pi, +\pi] \quad (1)$$

$$= \frac{1}{2\pi} \sum_{k=-\infty}^{+\infty} \gamma(k) \cos \lambda k \quad \text{car } \gamma \text{ est une fonction paire}$$

Cette expression donnant $f(\lambda)$ a un sens si la série $\sum_{k=-\infty}^{+\infty} |\gamma(k)|$ converge :

$$\sum_{k=-\infty}^{+\infty} |\gamma(k)| < +\infty \quad (2)$$

Si le processus stochastique X est continu, la transformée de Fourier de la fonction d'autocovariance γ s'écrit :

$$f(\lambda) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \gamma(\tau) \cos \lambda \tau \, d\tau \quad \lambda \in [-\pi, +\pi] \quad (3)$$

La condition :

$$\text{valeur absolue de la fonction } \gamma \text{ intégrable sur } \mathbb{R} \quad (4)$$

assure l'existence de f , partout définie.

Lorsque les conditions (2) ou (4) sont vérifiées, la fonction f converge uniformément, et on a la notation inverse :

$$\gamma(k) = \int_{-\pi}^{+\pi} \cos \lambda k f(\lambda) d\lambda \quad \forall k \quad (5)$$

Ainsi l'information contenue dans la suite des autocovariances $\{\gamma(k)\}$ ou dans γ est la même que celle apportée par la connaissance de f , (lorsque f existe). f peut être utilisée pour décrire le processus stochastique X : elle permet d'obtenir :

- les harmoniques utiles dans la recherche des cycles,
- les variations du processus autour de sa moyenne.

Dans la pratique (FIS 73), en supposant que X est un processus faiblement stationnaire, La condition suivante portant sur la fonction d'autocorrélation ρ

$$\lim_{k \rightarrow \infty} \rho(k) = 0 \quad (6)$$

assure l'existence de f . Dans le cas discret, il suffit que la suite des autocorrélations $\{\rho(k), k \in \mathbb{N}\}$ soit décroissante en valeur absolue.

L'étude des processus stochastiques pose des problèmes plus ou moins complexes suivant leurs propriétés. Nous allons explorer plus particulièrement ces difficultés par rapport à la propriété de stationnarité.

I-1.- Stationnarité stricte (AND 71)

X est un processus (fortement ou strictement) stationnaire si ses propriétés statistiques sont indépendantes de l'origine des temps.

La définition mathématique d'un processus strictement stationnaire est :

$$\forall (t_1, t_2, \dots, t_m) \in T^m : \forall h \in \mathbb{R}^+ \quad \forall m \in \mathbb{N}^+$$

$$F_X(t_1, t_2, \dots, t_m) = F_X(t_1+h, t_2+h, \dots, t_m+h) \quad (7)$$

F étant la fonction de répartition conjointe du processus stochastique X .

De cette définition, on peut déduire les propriétés suivantes :

- la fonction de répartition absolue $F(t; x)$ est indépendante de t
- $E(X(t)) = u \quad \forall t \in T$
- $\gamma(t_1, \tau) = \gamma(t_2, \tau) = \gamma(\tau) \quad \forall (t_1, t_2) \in T^2 \quad \forall \tau \in \mathbb{N}$
- $\text{Var}(X(t)) = \gamma(0) = \sigma^2 \quad \forall t \in T$
- $\rho(t, \tau) = \gamma(\tau)/\gamma(0) = \rho(\tau) \quad \forall t \in T \quad \forall \tau \in \mathbb{N}$

En outre, il est possible d'estimer la fonction de répartition F à partir d'une seule trajectoire, si le processus est ergodique.

I-2.- Faible stationnarité (COX 69, AND 71)

X est un processus faiblement stationnaire (quasi-stationnaire ou stationnaire du deuxième ordre) si les propriétés statistiques du premier et deuxième ordre sont indépendantes de l'origine des temps

- moyenne constante
- autocovariance indépendante de t mais dépendant seulement de l'intervalle considéré.

En d'autres termes, X est un processus faiblement stationnaire s'il vérifie les deux propriétés :

$$\left\{ \begin{array}{l} - E(X(t)) = \mu \quad \forall t \in T \\ - \gamma(t_1, \tau) = \gamma(t_2, \tau) = \gamma(\tau) \quad \forall (t_1, t_2) \in T^2, \forall \tau \in \mathbb{N} \end{array} \right. \quad (8)$$

De cette définition, on peut déduire les propriétés suivantes :

$$\left\{ \begin{array}{l} - \rho(t_1, \tau) = \rho(t_2, \tau) \quad \forall (t_1, t_2) \in T^2 \quad \forall \tau \in \mathbb{N} \\ - \text{Var}(X(t)) = \gamma(0) = \sigma^2 \quad \forall t \in T \end{array} \right.$$

Lorsque le processus est faiblement ou strictement stationnaire, il est possible de définir la fonction de densité spectrale f (si les conditions (2) ou (4) sont vérifiées).

Un processus X strictement stationnaire à variance finie est donc faiblement stationnaire. La réciproque est fautive, sauf pour les processus gaussiens (COX 69) qui sont complètement déterminés par la connaissance des deux premiers moments.

I-3.- Non-stationnarité (COX 69, LER 77)

Les différentes propriétés statistiques (moyenne, variance, ...) dépendent toutes du temps. Pour les estimer, il faut disposer de plusieurs trajectoires du processus et procéder à une estimation point par point. Si les différentes trajectoires sont indépendantes, on peut utiliser la théorie classique de l'estimation.

En présence de tels processus, on essaie généralement de se ramener à une étude de processus stationnaires de deux façons :

- découpage de la durée des observations en plusieurs intervalles de manière à obtenir un ou plusieurs processus stationnaires dont les caractéristiques changent avec l'intervalle considéré.

- décomposition du processus en deux processus $X = Y + \Gamma$

$$X(t) = Y(t) + h(t) \quad \forall t \in T$$

- avec Y processus faiblement stationnaire,
- h processus déterministe.

Cette deuxième approche est surtout utilisée par les économistes, la recherche de tendance consiste souvent à identifier le processus déterministe h . Par contre, la première approche met en évidence la possibilité de distinguer différentes échelles de temps en découpant la durée d'observation en intervalles de temps pendant lesquels l'hypothèse de stationnarité peut se justifier. Il n'existe aucune réponse théorique pour faire un bon découpage. Habituellement, la durée d'observation est découpée arbitrairement. Ensuite, on utilise des tests de faible stationnarité sur chaque intervalle considéré pour valider les hypothèses faites.

II - ESTIMATEURS DES MOMENTS DES PROCESSUS STOCHASTIQUES (HAN 60, COX 69, AND 71)

Suivant les propriétés vérifiées par le processus stochastique, il est plus ou moins facile de déterminer des estimateurs de grandeurs caractéristiques. Nous examinons cette question par rapport à la propriété de stationnarité en nous limitant aux propriétés du premier et deuxième ordre (définies à partir des moments d'ordre 1 ou 2).

II-1.- Stationnarité stricte

En vertu du théorème de DOOB-BIRKOFF, si le processus X est stationnaire et ergodique, il est alors possible d'estimer les différentes grandeurs caractéristiques (lorsqu'elles existent) en considérant une seule réplique du processus.

THEOREME DE DOOB-BIRKOFF (théorème 1)

Si le processus X est stationnaire et ergodique, les relations suivantes sont vérifiées :

$$E[X^n(t)] = E \left[\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T X^n(t) dt \right]$$

$$\begin{aligned} \bullet E[X(t) \cdot X(t+\tau)] &= E\left[\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T X(t) \cdot X(t+\tau) dt\right] \\ &= \gamma(\tau) + u^2 \end{aligned}$$

avec $E[X(t)] = u$ et $\gamma(\tau) = \text{Cov}[X(t), X(t+\tau)]$

Pour étudier un processus stochastique, il est donc intéressant qu'il soit ergodique, propriété que nous définissons maintenant.

Considérons un processus stochastique supposé stationnaire et variant discrètement aux instants t_1, t_2, \dots, t_n , une estimation de son $p^{\text{ième}}$ moment (lorsqu'il existe), est :

$$\widehat{E[X^p(t)]} = \frac{1}{t_n - t_0} \sum_{i=1}^n X^p(t_i - t_{i-1}) \quad (9)$$

L'estimation est obtenue à partir d'une trajectoire unique. Pour qu'elle soit "significative", il faut qu'elle soit "indépendante" de la trajectoire considérée.

Dire qu'un processus stochastique est ergodique par rapport au moment d'ordre p signifie que ce moment est indépendant du temps d'une part et que les moments temporels d'ordre $\leq p$ existent et ont même valeur quelle que soit la trajectoire du processus d'autre part.

Intuitivement, si on observe un processus stochastique stationnaire ayant une suite de covariances décroissantes dans le temps sur une durée suffisamment longue, on espère obtenir un grand nombre de variations possibles de ce processus. Ainsi, on peut penser qu'une seule trajectoire suffit pour résumer "l'information" contenue dans les autres trajectoires.

Cette propriété d'ergodicité peut se définir mathématiquement (DOO 53, BAR 56, HAN 60).

Le processus stochastique stationnaire est ergodique :

• par rapport à la moyenne si :

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T (X(t) - u) dt = 0 \quad (10)$$

• et plus généralement par rapport au moment d'ordre p si :

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T (X^p(t) - E[X^p(t)]) dt = 0 \quad (11)$$

Les conditions que doit satisfaire un processus stochastique stationnaire pour être ergodique sont très complexes en général. Ainsi, un processus gaussien stationnaire est ergodique si la fonction de densité spectrale f est absolument continue (DOO 53). Dans le cas de processus gaussiens discrets, cette condition d'ergodicité peut s'exprimer en fonction de la suite des coefficients d'autocorrélation $\{\rho(k), k \in \mathbb{N}\}$ (FIS 73) :

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=-n+1}^{n-1} \rho(k) = 0 \quad (12)$$

et

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=-n+1}^{n-1} \rho^2(k) = 0 \quad (13)$$

Ces conditions (12) et (13) sont vérifiées si la condition (6) l'est.

Ne pouvant s'assurer de l'ergodicité du processus stochastique, on recherche néanmoins des estimateurs ayant de bonnes propriétés.

Dans (HAN 60), HANNAN donne des conditions générales que doit vérifier un processus stochastique stationnaire de moyenne $u = 0$ pour que les estimateurs temporels soient des estimateurs convergens (***) :

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X(t_i)$$

est un estimateur convergent de la moyenne u si :

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=0}^{n-1} \gamma(k) = 0 \quad (14)$$

(***) \hat{A} est un estimateur convergent de a signifie :

$$\lim_{n \rightarrow \infty} P(|\hat{A} - a| \geq \epsilon) = 0 \quad \forall \epsilon > 0.$$

$$\hat{C}(k) = \frac{1}{n} \sum_{i=1}^{n-k} X(t_i)X(t_{i+k})$$

est un estimateur convergent de l'autocovariance $\gamma(k)$ d'ordre k si :

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=0}^{n-1} \gamma^2(k) = 0 \quad (15)$$

Ces conditions (14) et (15) sont équivalentes aux conditions (12) et (13). Elles sont vérifiées lorsque la condition (6) l'est. C'est donc cette condition (6) que nous utiliserons en pratique.

En conclusion, si on connaît une trajectoire du processus stochastique, il est possible d'utiliser les estimateurs temporels pour estimer tous ses moments. Si la durée d'observation $(t_n - t_0)$ est fixe, les estimateurs temporels définis dans la relation (9) sont non biaisés (***) .

S'il est impossible de connaître la trajectoire du processus sur toute la durée d'observation, soit parce que le processus est continu, soit parce qu'on ne connaît pas tous les instants t_i où $X(t)$ varie, on échantillonne alors le processus à des instants arbitrairement choisis t_i . On évite généralement qu'ils soient liés au phénomène étudié, en particulier lorsqu'il existe des phénomènes cycliques (LER 77). On fait l'hypothèse que pendant l'intervalle $[t_{i-1}, t_i]$, le processus ne varie pas de manière significative.

Généralement, on procède à un échantillonnage régulier tous les Δt :

$$\Delta t = t_i - t_{i-1} = \text{constante} \quad i = 2, 3, \dots, n$$

Si la durée d'observation $(t_n - t_0) = n\Delta t$, les estimateurs temporels sont de la forme :

$$\widehat{E[X^p(t)]} = \frac{1}{n} \sum_{i=1}^n X^p(t_i)$$

(***) \hat{A} estimateur de a est non biaisé signifie :

$$E(\hat{A}) = a$$

Désormais, nous notons $X(t_i), X_i$ et nous considérons différents estimateurs pour les propriétés du premier et deuxième ordre d'un processus discret.

II-1.1.- Moyenne (COX 69, BOX 70, AND 71)

Comme estimateur de μ est retenu \bar{X}

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \quad (16)$$

Cet estimateur est non biaisé. Sa variance est :

$$\text{Var}(\bar{X}) = \frac{1}{n} \sum_{k=-n+1}^{n-1} \left(1 - \frac{|k|}{n}\right) \gamma(k) \quad (17)$$

Cette expression a un sens si la série écrite au second membre converge (la condition (2) est une condition suffisante).

II-1.2.- Variance (BOX 70, LER 77)

Comme estimateur de σ^2 , on peut utiliser :

$$\widehat{\text{Var}(X(t))} = \frac{1}{n-1} \left[\sum_{i=1}^n X_i^2 - \frac{1}{n} \left(\sum_{i=1}^n X_i \right)^2 \right] \quad (18)$$

qui est non biaisé. Nous ne parlons pas de ses propriétés étant donné que c'est un estimateur de l'autocovariance d'ordre 0 (voir ci-dessous).

II-1.3.- Autocovariance (COX 69, AND 71, KLE 74, LER 77)

L'autocovariance d'ordre k , $\gamma(k)$ peut être estimée par l'estimateur :

$$\widehat{\gamma(k)} = \frac{1}{n-k} \left[\sum_{i=1}^{n-k} X_i X_{i+k} - \frac{1}{n-k} \left(\sum_{i=1}^{n-k} X_i \right) \left(\sum_{i=1}^{n-k} X_{i+k} \right) \right] \quad (19)$$

L'utilisation de cet estimateur doit être limitée à des valeurs de k telles que k/n soit faible. Généralement, on considère les valeurs de k inférieures à $n/10$ (KLE 74) ou $n/4$ (BOX 70, LER 77).

Si la série $\{\gamma(k), k \in \mathbb{N}\}$ converge, $\widehat{\gamma(k)}$ est asymptotiquement

non biaisé. Sa variance et ses covariances ont des expressions compliquées qui dépendent des moments du 4^{ème} ordre (HAN 60, AND 71) et sont généralement de l'ordre de 1/n.

D'autres estimateurs non biaisés peuvent être construits, mais ils sont très sensibles à l'évolution des observations (COX 69).

Parmi les autres estimateurs de $\gamma(k)$, citons :

$$\widehat{\gamma}_1(k) = \frac{1}{n-k} \sum_{i=1}^{n-k} (X_i - \bar{X})(X_{i+k} - \bar{X}) \quad (20)$$

$$\widehat{\gamma}_2(k) = \frac{n-k}{n} \widehat{\gamma}_1(k) \quad (21)$$

Nous ne nous attardons pas sur leurs propriétés qui ont été étudiées notamment par BARTLETT (BAR 56) et ANDERSON (AND 71). Simplement nous pouvons dire que l'estimateur $\widehat{\gamma}_2(k)$ est utilisé dans l'estimation des processus dits de "BOX et JENKINS" (BOX 70).

Pour notre part (chapitre V), nous utilisons le premier estimateur $\widehat{\gamma}(k)$.

II-1.4.- Autocorrélation (COX 69, FIS 73, LER 77)

Pour obtenir un estimateur de $\rho(k)$, il est possible de normer l'estimateur de la covariance en le divisant par l'estimateur de la variance :

$$\widehat{\rho}(k) = \frac{\widehat{\gamma}(k)}{\widehat{\gamma}(0)} \quad (22)$$

Si on utilise l'estimateur $\widehat{\gamma}_2(k)$ pour estimer $\gamma(k)$, l'estimateur $\widehat{\rho}(k)$ obtenu n'a pas de bonnes propriétés statistiques (FIS 73). Il est biaisé et non convergent. De plus, les différents estimateurs $\widehat{\rho}(k)$ sont corrélés.

Ceci a conduit à rechercher d'autres estimateurs, comme celui proposé par COX et LEWIS (COX 69) :

$$\widehat{\rho}_1(k) = \frac{\widehat{\gamma}_1(k)}{[\gamma'(k) \cdot \gamma''(k)]^{1/2}} \quad (23)$$

où $\widehat{\gamma}_1(k)$ est l'estimateur défini dans la relation (20)

$$\gamma'(k) = \frac{1}{n-k} \sum_{i=1}^{n-k} (X_i - \frac{1}{n-k} \sum_{j=1}^{n-k} X_j)^2$$

$$\gamma''(k) = \frac{1}{n-k} \sum_{i=1}^{n-k} (X_{i+k} - \frac{1}{n-k} \sum_{j=1}^{n-k} X_{j+k})^2$$

Cet estimateur $\widehat{\rho}_1(k)$ est asymptotiquement non biaisé et convergent, tandis que les différents estimateurs $\widehat{\rho}_1(k)$ sont corrélés.

Pour notre part (chapitre V), nous utilisons très simplement l'estimateur défini en (22) suivant les conclusions de BOX et JENKINS (BOX 70) qui le préfèrent parce qu'il est facile à calculer tandis que ses propriétés ne sont guère plus mauvaises.

II-1.5.- Densité spectrale de puissance (BAR 56, HAN 60, COX 69, AND 71)

Pour estimer la densité spectrale de puissance f , deux approches sont possibles :

- utilisation du périodogramme,
- utilisation de la définition de f en tant que transformée de Fourier de la suite d'autocovariance $\{\gamma(k)\}$.

Nous les examinons successivement.

a) Utilisation du périodogramme (BAR 56, HAN 60, COX 69, AND 71)

Habituellement, dans l'analyse spectrale d'une série d'observations ponctuelles régulièrement espacées dans le temps, on peut utiliser le périodogramme égal au carré du module de la transformée de Fourier finie de la suite des variables aléatoires X_i , à un facteur multiplicatif près :

$$I(\lambda) = \frac{1}{2\pi n} \left| \sum_{k=1}^n (X_k - \bar{X}) \exp(ik\lambda) \right|^2 \quad \lambda \in [-\pi, +\pi] \quad (24)$$

Généralement, l'étude de $I(\lambda)$ n'est faite que pour les valeurs λ de la forme :

$$\lambda_p = \frac{2\pi p}{n} \quad p = 1, 2, \dots, \text{ent}[\frac{n}{2}]$$

Il est démontré (COX 69, AND 71) qu'aucune information supplémentaire n'est obtenue en considérant d'autres valeurs de λ . Par ailleurs les propriétés d'orthogonalité de $\cos \frac{2\pi p}{n}$ et $\sin \frac{2\pi p}{n}$ pour $p = 1, 2, \dots, n$, facilitent les calculs généraux.

En développant l'expression de $I(\lambda)$ précédente, on obtient :

$$I(\lambda) = \frac{1}{2\pi n} \sum_{j=1}^n \sum_{k=1}^n (X_j - \bar{X})(X_k - \bar{X}) \cos[(j-k)\lambda] \quad \lambda \in [-\pi, +\pi]$$

$$\Leftrightarrow I(\lambda) = \frac{1}{2\pi} \sum_{k=-n+1}^{n+1} \widehat{\gamma}_2(k) \cos \lambda k \quad \lambda \in [-\pi, +\pi] \quad (25)$$

$I(\lambda)$ est souvent utilisé pour rechercher des périodicités cachées dans l'analyse de séries chronologiques (HAN 60, AND 71), bien qu'ayant de mauvaises propriétés statistiques. Il est asymptotiquement non biaisé et non convergent.

b) Utilisation de la définition de f en fonction des autocovariances

Il est possible d'obtenir un estimateur de f en utilisant un estimateur de l'autocovariance dans la définition (1).

$$\widehat{f}(\lambda) = \frac{1}{2\pi} \sum_{k=-\infty}^{+\infty} \widehat{\gamma}(k) \cos \lambda k \quad \lambda \in [-\pi, +\pi] \quad (26)$$

Par exemple, on peut utiliser l'estimateur $\widehat{\gamma}_2(k)$ défini par la relation (21), en restreignant la somme à un intervalle fini, comme celui de l'échantillon.

On peut justifier cette restriction dans la sommation par le fait suivant : si $|\gamma|$ est une fonction monotone décroissante, on peut alors négliger la contribution des différentes estimations de $\gamma(k)$ voisines de zéro ce qui se produit pour k suffisamment grand (lorsque le processus n'est pas périodique).

Cet estimateur $\widehat{f}(\lambda)$ n'a pas de bonnes propriétés statistiques : il est asymptotiquement non biaisé et non convergent (AND 71).

c) Utilisation d'estimateurs lissés de f

Dans cette partie, nous considérons maintenant l'obtention d'estimateurs :

- asymptotiquement non biaisés et
- ayant une variance en $1/n$.

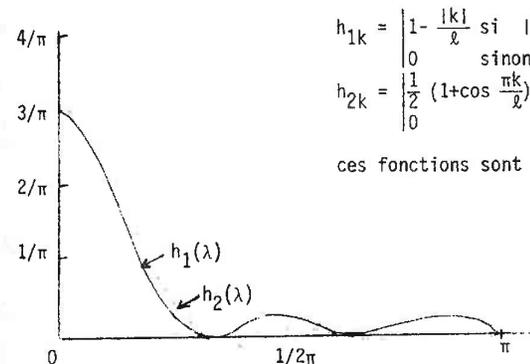
Pour satisfaire ces objectifs, nous considérons des estimateurs lissés de f

$$\widehat{f}(\lambda) = \frac{1}{2\pi} \sum_{k=-n+1}^{n-1} h_k \widehat{\gamma}(k) \cos \lambda k \quad \lambda \in [-\pi, +\pi] \quad (27)$$

- où
- $\widehat{\gamma}(k)$ est un estimateur de l'autocovariance d'ordre k et
 - h_k les coefficients de Fourier d'une fonction de pondération spectrale h définie ainsi :

$$h(\lambda) = \frac{1}{2\pi} \sum_{k=-\infty}^{+\infty} h_k \exp(ik\lambda) \quad \lambda \in [-\pi, +\pi]$$

Ces fonctions de pondération spectrale sont aussi appelées fenêtres. Le graphe de ces fonctions est caractérisé par un grand lobe centré sur le point 0 et bordé par des lobes plus petits (fig. 3.2).



$$h_{1k} = \begin{cases} 1 - \frac{|k|}{l} & \text{si } |k| \leq l < n \\ 0 & \text{sinon} \end{cases}$$

$$h_{2k} = \begin{cases} \frac{1}{2} (1 + \cos \frac{\pi k}{l}) & \text{si } |k| \leq l < n \\ 0 & \text{sinon} \end{cases}$$

ces fonctions sont paires

Fig 3.2 : Exemple de fenêtres (tiré de COX 69)

Différentes fenêtres ont été définies, notamment par BARTLETT, DANIELL, TUKEY, HANNING, HAMMING, PARZEN (HAN 60, COX 69, AND 71).

Généralement, on utilise une fenêtre tronquée comme celle proposée par BARTLETT :

$$h_k = h_{-k} = \begin{cases} 1 - |k|/\ell & \text{si } k \leq \ell < n \\ 0 & \text{sinon} \end{cases}$$

On se restreint à un certain nombre d'autocovariances $\gamma(k)$. Dans la plupart des cas, les autocovariances $\gamma(k)$ sont généralement voisines de zéro, sauf pour un nombre restreint ℓ , ce qui suggère de supprimer les autocovariances d'ordre k supérieur à ℓ .

L'utilisation de tels estimateurs tend à minimiser les effets dus à l'estimation de densité spectrale de processus stochastique infini en considérant un intervalle fini.

Dans l'estimation de la densité spectrale, il faut obtenir le pouvoir résolvant maximal, ce qui signifie la possibilité de distinguer les différentes estimations de $\widehat{f(\lambda)}$ avec diverses fréquences λ , pour un échantillon de taille donnée.

Pour notre part (chapitre V), nous utilisons un estimateur lissé de la densité spectrale avec la fenêtre tronquée suivante :

$$h_k = h_{-k} = 1 - \frac{|k|}{n} \quad |k| = 1, 2, \dots, n-1$$

d'où l'expression suivante pour l'estimateur employé :

$$\widehat{f(\lambda)} = \frac{1}{2\pi} \left[\widehat{\gamma(0)} + 2 \sum_{k=1}^{n-1} \left(1 - \frac{k}{n}\right) \widehat{\gamma(k)} \right] \quad (28)$$

II-2.- Stationnarité faible (COX 69, AND 71, LER 77)

Seules les propriétés du premier et deuxième ordre sont indépendantes du temps. Il est possible de tester le caractère faiblement stationnaire d'un processus (PRS 69).

Pour estimer la moyenne, la variance, l'autocovariance d'ordre k et la densité, on utilise les mêmes estimateurs que ceux définis pour les processus stochastiques strictement stationnaires.

II-3.- Non stationnarité

Toutes les propriétés statistiques étant fonction du temps, toute grandeur caractéristique du processus stochastique doit être estimée point par point. Pour cela, il faut disposer de plusieurs répliques du processus si possible indépendantes et utiliser les estimateurs classiques (FOU 67).

En général, lorsqu'on a une estimation d'une grandeur caractéristique du processus stochastique, on désire connaître la qualité de la mesure. Habituellement, la détermination de la précision statistique d'une estimation se ramène au calcul d'une variance sur cette estimation.

III - CONSTRUCTION D'INTERVALLES DE CONFIANCE

Le problème de la précision statistique d'une estimation peut être résolu en construisant un intervalle de confiance pour la valeur théorique exacte.

Si nous nous intéressons à la moyenne u d'un processus stochastique stationnaire, le calcul de l'intervalle de confiance de la moyenne u au seuil α donne les bornes de l'intervalle dans laquelle u a une probabilité $(1-\alpha)$ de se trouver :

$$1-\alpha = P[w_{\alpha,n} < u < v_{\alpha,n}]$$

$w_{\alpha,n}$ et $v_{\alpha,n}$ sont les bornes de l'intervalle de confiance de u au seuil α en ayant fait n mesures.

Il existe différentes méthodes pour calculer un intervalle de confiance. Nous limitons notre étude à celles concernant les espérances mathématiques de processus stochastiques faiblement stationnaires.

Considérons l'échantillon x_1, x_2, \dots, x_n suite de réalisations des variables aléatoires X_1, X_2, \dots, X_n mutuellement indépendantes et de même loi.

Pour estimer la moyenne u inconnue, on utilise l'estimateur arithmétique \bar{X} :

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

Cet estimateur \bar{X} est non biaisé, convergent et efficace (FOU 67).
L'inégalité de Bienaymé-Tchebicheff donne un intervalle de confiance sur u . En effet :

$$P[|\bar{X}-u| < \varepsilon] \geq 1 - \frac{\text{Var}(\bar{X})}{\varepsilon^2} \quad \forall \varepsilon > 0$$

$$\Leftrightarrow P[|\bar{X}-u| < \varepsilon\sqrt{\text{Var}(\bar{X})}] \geq 1 - \frac{1}{\varepsilon^2} \quad \forall \varepsilon > 0$$

Pour obtenir un intervalle de confiance au seuil $\alpha = 5\%$, on a $1-1/\varepsilon^2 = 0,95$ ou plus simplement $\varepsilon \approx 4,47$. Ainsi la largeur de l'intervalle de confiance est

$d \approx 4.47$ écart-type de \bar{X} autour de la moyenne u .

L'inégalité de Bienaymé-Tchebicheff ne fait aucune hypothèse sur la loi de l'estimateur \bar{X} , on suppose simplement que $\text{Var}(\bar{X})$ existe.

Très souvent, on peut calculer un intervalle de confiance plus serré.

Si la loi de \bar{X} est gaussienne, l'inégalité de Bienaymé Tchebicheff peut s'écrire :

$$P[|\bar{X}-u| < \varepsilon\sqrt{\text{Var}(\bar{X})}] = \frac{1}{\sqrt{2\pi}} \int_{-\varepsilon}^{+\varepsilon} \exp\left(-\frac{t^2}{2}\right) dt = 2\Phi(\varepsilon) - 1$$

$\Phi(\varepsilon)$ étant la fonction de répartition de la loi normale centrée réduite. Au seuil $\alpha = 5\%$, la largeur de cet intervalle est :

$d = 1.96$ écart-type de \bar{X} autour de la moyenne u .

Cette hypothèse de normalité se vérifie si :

- le processus est strictement stationnaire
- les variables aléatoires X_i sont mutuellement indépendantes, suivent la même loi et ont une variance finie.

Ces conditions permettent d'utiliser le théorème de la limite centrale pour obtenir la loi limite de \bar{X} .

Ainsi, l'hypothèse de normalité permet de réduire presque de moitié la largeur de l'intervalle de confiance obtenu.

Cependant, pour utiliser cette inégalité, il faut être capable d'estimer $\text{Var}(\bar{X})$, problème que nous traitons maintenant. Plusieurs cas sont à envisager suivant la disponibilité ou non de plusieurs répliques du processus.

Etant confronté à l'analyse des mesures obtenues pendant la simulation et voulant calculer des intervalles de confiance (IV.2.5), nous développons cette partie.

III-1.- Plusieurs répliques indépendantes du processus (FIS 73, KLE 74, KOB 78)

Soient p répliques indépendantes du processus stochastique stationnaire constituées des réalisations suivantes :

$$x_{ij} \text{ réalisation de } X_{ij} \quad i = 1, \dots, n_j \quad j = 1, \dots, p$$

En considérant chaque réplique, on peut définir un estimateur de u , \bar{X}_j :

$$\bar{X}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} X_{ij}$$

Les p répliques étant indépendantes, les estimateurs \bar{X}_j sont indépendants. Comme estimateur de u , on peut choisir \bar{X} défini à partir des estimateurs \bar{X}_j :

$$\bar{X} = \frac{1}{p} \sum_{j=1}^p \bar{X}_j$$

\bar{X} est un estimateur non biaisé :

$$E(\bar{X}) = E\left[\frac{1}{p} \sum_{j=1}^p \bar{X}_j\right] = \frac{1}{p} E\left(\sum_{j=1}^p \bar{X}_j\right) = \frac{pu}{p} = u$$

Calculons $\text{Var}(\bar{X})$:

$$\text{Var}(\bar{X}) = \text{Var}\left[\frac{1}{p} \sum_{j=1}^p \bar{X}_j\right] = \frac{1}{p^2} \text{Var}\left(\sum_{j=1}^p \bar{X}_j\right) = \frac{1}{p^2} \sum_{j=1}^p \text{Var}(\bar{X}_j)$$

Les estimateurs \bar{X}_j sont mutuellement indépendants et suivent asymptotiquement la même loi : on peut appliquer le théorème limite centrale pour un processus stationnaire r-dépendant :

X est un processus stationnaire r-dépendant si $\gamma(k) = 0 \quad \forall k > r$

THEOREME r-limite centrale (Théorème 2)

Soit X processus strictement stationnaire r-dépendant de moyenne $E[X(t)] = u$ satisfaisant la condition suivante :

$$E[|X(t)|^3] < \infty$$

alors l'estimateur temporel \bar{X} de la moyenne u défini par :

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

est asymptotiquement normal de moyenne u et de variance

$$\text{Var}(\bar{X}) = \frac{1}{n} \left[1 + 2 \sum_{k=1}^r \left(1 - \frac{k}{n} \right) \gamma(k) \right]$$

Ce théorème est important : il permet de connaître la loi limite de l'estimateur temporel de la moyenne \bar{X} .

Ainsi les estimateurs \bar{X}_j sont mutuellement indépendants et asymptotiquement gaussiens (en supposant que le théorème ci-dessus est vérifié). On peut donc utiliser la statistique classique pour calculer $\text{Var}(\bar{X})$ à partir de celles des estimateurs \bar{X}_j :

$$\text{Var}(\bar{X}) = \text{Var}\left(\frac{1}{p} \sum_{j=1}^p \bar{X}_j\right) = \frac{1}{p^2} \sum_{j=1}^p \text{Var} \bar{X}_j = \frac{\text{Var}(\bar{X}_j)}{p}$$

Il est possible d'estimer $\text{Var}(\bar{X}_j)$ par l'estimateur non biaisé suivant :

$$\widehat{\text{Var}(\bar{X}_j)} = \frac{1}{p-1} \sum_{j=1}^p (X_j - \bar{X})^2 = \frac{1}{p-1} \left[\sum_{j=1}^p (\bar{X}_j)^2 - \frac{1}{p} \left(\sum_{j=1}^p \bar{X}_j \right)^2 \right]$$

Ceci nous permet d'obtenir un estimateur de $\text{Var}(\bar{X})$:

$$\widehat{\text{Var}(\bar{X})} = \frac{1}{p(p-1)} \sum_{j=1}^p (\bar{X}_j - \bar{X})^2$$

En utilisant le classique théorème limite centrale

$$\frac{\bar{X} - u}{(\widehat{\text{Var}(\bar{X})})^{1/2}} \text{ suit une loi normale centrée réduite.}$$

Comme nous utilisons un estimateur de la variance de \bar{X} obtenu à partir de variables aléatoires \bar{X}_j asymptotiquement normales et indépendantes

$$\frac{\bar{X} - u}{(\widehat{\text{Var}(\bar{X})})^{1/2}} \text{ suit une loi du Student à } (p-1) \text{ degrés de liberté.}$$

On peut donc construire un intervalle de confiance de u au seuil α :

$$P\left[\bar{X} - (\widehat{\text{Var}(\bar{X})})^{1/2} \cdot z_{\alpha/2}^{p-1} < u < \bar{X} + (\widehat{\text{Var}(\bar{X})})^{1/2} \cdot z_{\alpha/2}^{p-1} \right] = 1 - \alpha$$

avec $z_{\alpha/2}^{p-1}$ lu dans une table du Student à p-1 degrés de liberté au seuil $\alpha/2$.

III-2.- Une seule réplique du processus

Nous nous plaçons maintenant dans le cas où nous disposons d'une seule réplique du processus. Pour estimer la variance de \bar{X} , nous considérons trois méthodes :

- méthode utilisant la fonction d'autocorrélation,
- méthode des blocs,
- méthode utilisant des cycles indépendants.

III-2.1.- Méthode utilisant les coefficients d'autocorrélation (AND 71, KLE 74)

Les différentes observations sont utilisées pour estimer la variance.

$$\text{Var}(\bar{X}) = \frac{1}{n^2} \left(\text{Var} \sum_{i=1}^n X_i \right) = \frac{1}{n^2} \sum_{i,j=1}^n \text{Cov}(X_i, X_j)$$

X étant stationnaire et γ étant une fonction paire, on peut réécrire :

$$\text{Var}(\bar{X}) = \frac{1}{n} \left[\gamma(0) + 2 \sum_{k=1}^{n-1} \left(1 - \frac{|k|}{n}\right) \gamma(k) \right]$$

Utilisant un estimateur de $\gamma(k)$, $\text{Var}(\bar{X})$ peut être estimée par :

$$\widehat{\text{Var}}(\bar{X}) = \frac{1}{n} \left[\widehat{\gamma}(0) + 2 \sum_{k=1}^{n-1} \left(1 - \frac{|k|}{n}\right) \widehat{\gamma}(k) \right]$$

Ayant déterminé un estimateur de $\text{Var}(\bar{X})$, l'inégalité de Bienaymé-Tchebicheff peut servir pour calculer un intervalle de confiance.

Cette méthode n'est à utiliser que lorsque les autocovariances décroissent rapidement vers zéro pour limiter le nombre d'estimations (définition des estimateurs $\widehat{\gamma}(k)$).

Si f est continue au point 0 (la condition (2) est suffisante), la limite de $\text{Var}(\bar{X})$ est (AND 71) :

$$\lim_{n \rightarrow \infty} \text{Var}(\bar{X}) = \frac{2\pi f(0)}{n}$$

On peut donc aussi utiliser les estimateurs de la densité spectrale f au point 0 pour estimer $\text{Var}(\bar{X})$. Dans (FIS 73), on peut trouver des compléments à cette approche faisant intervenir l'analyse spectrale.

III-2.2.- Méthode des blocs (CON 63, KLE 74, LER 77, KOB 78)

Cette méthode proposée par CONWAY (CON 63) suggère de scinder l'ensemble des observations x_1, x_2, \dots, x_n , réalisations des variables aléatoires X_1, X_2, \dots, X_n en p blocs d'égal longueur ℓ . (Si le nombre de mesures n n'est pas un multiple de p , on écarte généralement les mesures centrales).

$$\frac{|x_1, \dots, x_\ell|}{\text{bloc 1}} \quad \dots \quad \frac{|x_{(j-1)\ell+1}, \dots, x_{j\ell}|}{\text{bloc } j} \quad \dots \quad \frac{|x_{(p-1)\ell+1}, \dots, x_{p\ell}|}{\text{bloc } p}$$

Sur chaque bloc est défini un estimateur de la moyenne u :

$$\bar{X}_j = \frac{1}{\ell} \sum_{i=1}^{\ell} X_{(j-1)\ell+i} \quad j = 1, 2, \dots, p$$

On fait alors l'hypothèse qu'il est possible de calculer ℓ tel que les estimateurs \bar{X}_j soient mutuellement indépendants.

A partir des estimateurs \bar{X}_j est défini \bar{X} , autre estimateur de u :

$$\bar{X} = \frac{1}{p} \sum_{j=1}^p \bar{X}_j$$

Le coefficient de corrélation $\rho(\bar{X}_j, \bar{X}_{j+1})$ entre les estimateurs de deux blocs successifs peut être calculé. Il dépend des coefficients d'autocorrélation $\rho(k)$ de la série initiale. Lorsque la suite $\{\rho(k)\}$ est positive décroissante en tendant rapidement vers 0, $\rho(\bar{X}_j, \bar{X}_{j+1})$ devient négligeable (KLE 74).

Ayant constitué p blocs d'égal longueur, approximativement indépendants, l'intervalle de confiance de u peut être calculé de la même manière qu'en 3.1.

Si on fait l'hypothèse que le processus X est stationnaire r -dépendant, ℓ doit être supérieur à r afin de pouvoir appliquer le théorème limite centrale r -dépendant.

Sachant que

$$\text{Var}(\bar{X}) = \frac{\text{Var}(\bar{X}_j)}{p}$$

et utilisant l'estimateur classique de $\text{Var}(\bar{X}_j)$:

$$\widehat{\text{Var}}(\bar{X}_j) = \frac{1}{p-1} \left[\sum_{j=1}^p (\bar{X}_j)^2 - \frac{1}{p} \left(\sum_{j=1}^p \bar{X}_j \right)^2 \right]$$

$\text{Var}(\bar{X})$ peut être estimée par :

$$\widehat{\text{Var}}(\bar{X}) = \frac{1}{p(p-1)} \left[\sum_{j=1}^p (\bar{X}_j)^2 - \frac{1}{p} \left(\sum_{j=1}^p \bar{X}_j \right)^2 \right]$$

On obtient donc l'intervalle de confiance de la moyenne μ au seuil α :

$$P\left[\bar{X} - (\widehat{\text{Var}} \bar{X})^{1/2} \cdot z_{\alpha/2}^{p-1} < \mu < \bar{X} + (\widehat{\text{Var}} \bar{X})^{1/2} \cdot z_{\alpha/2}^{p-1}\right] = 1 - \alpha$$

avec $z_{\alpha/2}^{p-1}$ lu dans une table de Student à $(p-1)$ degrés de liberté au seuil $\alpha/2$.

Pour utiliser cette méthode il faut pouvoir déterminer la taille minimale d'un bloc. A notre connaissance, il n'existe aucune solution théorique mais seulement des méthodes séquentielles.

Parmi celles existantes, on peut citer :

- a) Une méthode basée sur les coefficients d'autocorrélation entre la moyenne de deux blocs successifs (procédure de MECHANIC et MacKay) (KLE 74)

L'algorithme proposé (KLE 74) est basé sur l'idée suivante : "Augmenter la taille ℓ d'un bloc jusqu'à ce que l'autocorrélation existant entre la moyenne de deux blocs successifs devienne négligeable".

En pratique, cette méthode est utilisable même si la suite des coefficients d'autocorrélation $\{\rho(k)\}$ devient négative avant de décroître vers zéro. Il suffit que cette suite soit monotone décroissante en valeur absolue (KLE 74).

Cette procédure itérative nécessite un temps de calcul relativement long pour déterminer une taille de bloc adéquate (KLE 74).

- b) Une méthode basée sur les coefficients d'autocorrélation $\rho(k)$ (LER 77)

J. LEROUDIER dans (LER 77) propose d'estimer ℓ_0 à partir de la fonction d'autocorrélation de telle sorte que

$$|\rho(\ell_0)| = \left| \frac{\gamma(\ell_0)}{\gamma(0)} \right| \ll 1.$$

On procède de manière itérative en estimant $\rho(\ell_0)$, jusqu'à ce que $\rho(\ell_0) \approx 0$. En pratique on peut prendre :

$$|\widehat{\rho}(\ell_0)| < 0.05$$

La longueur de bloc ℓ est choisie dans l'intervalle $[\ell_0, 2 * \ell_0]$. Théoriquement, il ne faudrait considérer que des blocs séparés par une distance ℓ , c'est à dire un bloc sur deux pour s'assurer de l'indépendance des estimateurs. Dans (CON 63), CONWAY démontre que si la longueur ℓ des blocs est identique, la variance de l'estimateur \bar{X} défini en utilisant tous les blocs est inférieure à la variance de l'estimateur \bar{X} défini en ne considérant qu'un bloc sur deux. Cependant, si cet estimateur est de variance inférieure, CONWAY ne donne aucune méthode pour estimer sa variance. Cette dernière peut être majorée par la variance de l'estimateur qui n'utilise que la moitié des blocs, ce qui donne un intervalle de confiance plus grand que nécessaire.

Dans (LER 77), J. LEROUDIER rapporte que selon son expérience, il est possible d'utiliser tous les blocs et de prendre en compte toutes les mesures effectuées.

C'est cette méthode que nous utiliserons en suivant l'expérience de LEROUDIER (chapitre V).

En conclusion, cette méthode des blocs est très intéressante bien que nécessitant l'hypothèse suivante :

La suite des coefficients d'autocorrélation $\{\rho(k)\}$ est monotone décroissante en valeur absolue.

Ce qui implique notamment l'absence de périodicité.

Par ailleurs, l'estimation de ℓ_0 peut être obtenue à partir d'une méthode (LER 77) basée sur une analyse spectrale d'échantillon (FIS 67).

III-2.3.- Méthodes utilisant des cycles indépendants (CRA 74, CRA 75)

Cette méthode est utilisable pour les systèmes dits à points de régénération.

Un système stochastique est dit à point de régénération s'il existe avec une probabilité 1, des instants non nuls tels que le système soit à ces instants (y compris le passé) exactement dans le même état qu'à l'instant initial $t = 0$.

Les instants où le système se trouve dans un état de régénération sont appelés points de régénération.

En choisissant un point de régénération comme état initial du système, on peut décomposer la trajectoire du processus stochastique en plusieurs trajectoires statistiquement indépendantes et identiques, séparées chacune par un point de régénération.

On peut alors utiliser la statistique classique en n'oubliant pas que la durée d'un cycle est une variable aléatoire corrélée à l'échantillon.

Pour appliquer cette méthode, deux conditions doivent être satisfaites :

- Le système admet des points de régénération,
- Le temps moyen entre deux points de régénération est fini.

Etant donné qu'il est très difficile de définir les points de régénération dans un système et de montrer que le temps moyen séparant deux points est fini, on peut appliquer cette méthode :

- soit en faisant l'hypothèse que le temps moyen entre deux points est fini (LAV 75),
- soit en choisissant des points de régénération approchés pour garantir un temps moyen fini (CRA 75b).

Cette méthode, bien que très intéressante, n'est pas toujours facile à utiliser. Il faut choisir des points de régénération, ce qui n'est pas toujours simple. Ne l'ayant pas utilisée, nous n'en parlerons pas plus longuement.

IV - ANALYSE DES SERIES CHRONOLOGIQUES

Dans cette partie, nous nous intéressons à diverses méthodes d'analyse des séries chronologiques. En cette matière les ouvrages sont relativement nombreux. Nous faisons essentiellement référence aux travaux de HANNAN (HAN 60) COX et LEWIS (COX 69), BOX et JENKINS (BOX 70), ANDERSON (AND 71), GRANGER et NEWBOLD (GRA 77) sans oublier des ouvrages plus pratiques et plus techniques (BEN 66).

Ayant été confrontés au problème d'analyser des séries chronologiques, nous avons appliqué plusieurs méthodes appropriées et notre point de vue est celui d'un utilisateur.

IV-1.- Définition (AND 71, BAR 71)

Une série chronologique (ou une chronique) est un ensemble d'observations chiffrées et ordonnées généralement dans le temps.

$$\{x_i | i = 1, 2, \dots, n\}$$

Pour les analyser, nous avons recours à la théorie des processus stochastiques. En effet, on peut considérer une série chronologique comme une réalisation d'un processus stochastique.

Dans cette étude, nous considérons uniquement les séries discrètes et nous les analysons en considérant des propriétés du processus stochastique sous-jacent.

IV-2.- Généralités

Les séries chronologiques peuvent être étudiées à des fins diverses :

- faire de la prédiction en connaissant le passé,
- obtenir des renseignements sur la structure du processus générateur.

Cette deuxième approche nous intéresse plus précisément : nous cherchons à obtenir des caractéristiques sur une classe de processus stochastiques dont on puisse considérer la série analysée comme une trajectoire possible.

L'analyse de la série chronologique x_1, x_2, \dots, x_n , réalisation de la suite de variables aléatoires X_1, X_2, \dots, X_n peut s'effectuer en utilisant un modèle statistique où les différentes variables aléatoires X_i peuvent s'écrire

$$X_i = f_i + U_i \quad i = 1, 2, \dots, n$$

où f_i correspond à une partie déterministe, fonction de différents paramètres tandis que U_i est une variable aléatoire.

Dans les modèles les plus anciens, U_i sert à caractériser l'erreur d'observation. Très souvent, on fait l'hypothèse que c'est un bruit blanc,

ce qui signifie que :

$$\begin{aligned} & \bullet E(U_i) = 0 \\ & \bullet E(U_i U_j) = \begin{cases} \sigma^2 & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \end{aligned}$$

La connaissance du modèle est obtenue par la détermination de f_i partir des observations en considérant le système d'équations :

$$x_i = f_i + u_i \quad i = 1, 2, \dots, n$$

où u_i est une réalisation de la variable aléatoire U_i .

Le temps ou plutôt l'ordre d'apparition des observations tient une place importante. Ce facteur peut être pris en compte soit dans la partie déterministe f_i , soit dans la partie aléatoire U_i ou encore en combinant ces deux approches.

Si on se réfère aux économistes qui utilisent beaucoup les séries chronologiques (PHL 79), leurs premiers modèles faisaient intervenir le temps dans f_i en distinguant quatre composantes dans X_i :

- une tendance générale T_i dont il est difficile de donner une définition précise : elle englobe les phénomènes de croissance et de décroissance.
- une variation conjoncturelle C_i à caractère rythmique, mais non strictement cyclique ou périodique.
- un mouvement saisonnier S_i à caractère périodique.
- des fluctuations accidentelles U_i à caractère aléatoire.

L'identification des composantes déterministes (T_i, C_i, S_i) s'effectue généralement en les supposant indépendantes et combinées suivant différents schémas :

- additif $f_i = T_i + S_i + C_i$
- multiplicatif $f_i = T_i S_i C_i$
- mixte (en combinant les deux précédents).

Bien que de manipulation aisée, ces modèles ne peuvent s'appliquer qu'à un nombre restreint de séries, les statisticiens ont donc recherché des modèles plus généraux faisant aussi intervenir le temps dans la partie aléatoire, notamment en considérant U_i comme un processus stochastique stationnaire (BOX 70, AND 71).

IV-3.- Démarche d'analyse

L'analyse d'une série chronologique peut se dérouler en trois étapes que nous détaillerons ensuite.

i) Visualisation

Tout d'abord, il faut visualiser l'ensemble des données. Ceci permet d'apprécier qualitativement certaines caractéristiques comme une tendance, l'existence de composantes (pseudo-) périodiques, des discontinuités ou des valeurs aberrantes.

Ceci peut être fait en traçant un graphe de la série en fonction du temps.

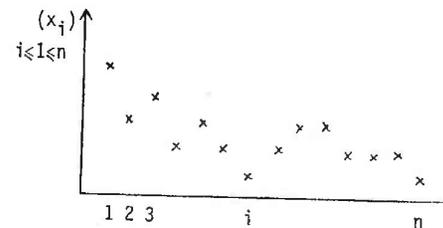


Fig 3.3 : Exemple de graphe de données

Etant donné la difficulté d'interprétation de ce graphe, cette étude préalable peut être complétée en visualisant d'autres séries obtenues à partir de la série initiale. Ceci peut faciliter, entre autres, l'appréciation portée sur la dépendance temporelle des observations.

ii) Vérification de certaines caractéristiques du processus stochastique sous-jacent

La théorie développée concerne essentiellement des processus stationnaires. Pour être en mesure de l'appliquer, il faut donc obtenir des séries stationnaires (processus stochastiques sous-jacents stationnaires). Or en pratique, la plupart des séries ne l'étant pas, il faut donc identifier les composantes déterministes (tendance, cycles) et transformer

éventuellement les séries pour appliquer la théorie statistique.

iii) Identification du processus stochastique sous-jacent

Après la détermination de paramètres concernant la partie déterministe, il peut être intéressant de rechercher un modèle probabiliste pour la partie aléatoire.

Nous distinguons deux catégories de modèles suivant l'indépendance ou non des différentes variables aléatoires :

- les variables aléatoires sont indépendantes. Si elles suivent la même loi, la détermination de la loi de distribution peut se faire soit en utilisant les statistiques classiques (FOU 67) (lois connues) soit en estimant à partir de l'échantillon une fonction qui a les propriétés de densité. Dans le cas opposé, on peut estimer la loi de probabilité comme une combinaison linéaire de distributions de probabilité connues (SCH 75).

- les variables aléatoires sont dépendantes. Il faut utiliser des modèles prenant en compte cette dépendance, par exemple ceux dits de BOX et JENKINS (BOX 70, AND 71, AND 75, GRA 77).

Après avoir présenté la démarche, nous considérons chaque étape successivement.

IV-4.- Visualisation

Avant d'entreprendre une analyse très fine, il faut se faire une idée sur diverses propriétés du processus stochastique sous-jacent. Pour ce faire, on peut tracer différents graphes.

IV-4.1.- Le graphe des données

Un examen rapide de ce graphe permet d'infirmer ou non l'existence de tendance, de composantes (pseudo-périodiques ...).

La construction d'un tel graphe n'est pas toujours aussi simple qu'il n'y paraît a priori : un changement d'échelle en abscisse ou en ordonnée, un tracé continu ou non donnent une vision différente de la série.

Par ailleurs, avant de poursuivre l'analyse, il est important de détecter les valeurs "aberrantes" et de les supprimer ou modifier soit automatiquement en donnant une règle (BEN 66) soit manuellement. Cette dernière

solution semble préférable, car il n'est pas toujours facile de faire la distinction entre une tendance et des fluctuations aléatoires.

Lorsque le processus stochastique sous-jacent est supposé faiblement stationnaire, on peut utiliser différents types de graphes pour mettre en évidence l'existence de composantes cycliques ou de dépendance temporelle entre les observations. Ces graphes correspondent à l'estimation de diverses caractéristiques du processus stochastique (corrélogramme, périodogramme ...).

IV-4.2.- Le corrélogramme

C'est le graphe de la fonction d'autocovariance γ estimée pour différentes valeurs k , $k = 1, 2, \dots, K < n$.

Il permet de détecter la présence de composantes sinusoïdales dues à des phénomènes (pseudo-) périodiques et de se faire une idée sur la dépendance temporelle des observations.

Pour que les estimations soient significatives, il faut au moins une cinquantaine d'observations en se limitant à des valeurs de k ne dépassant pas le quart des mesures (BOX 70).

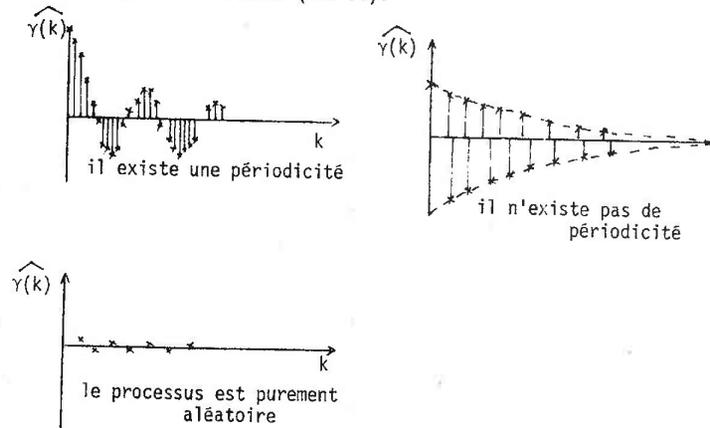


Fig 3.4 : Exemple de corrélogrammes (tiré de BEN 66)

Très souvent, la recherche de périodicités cachées s'effectue par une analyse harmonique de la chronique et en construisant divers graphes que nous présentons maintenant en précisant la théorie sous-jacente.

IV-4.3.- Le périodogramme de Schuster (AND 71)

Ce graphe introduit initialement par Schuster en 1898 servait à détecter et estimer l'amplitude d'une composante sinusoïdale de fréquence connue λ noyée dans du bruit.

Dans cette méthode, le modèle considéré est :

$$X_i = f_i + U_i \quad i = 1, 2, \dots, n$$

où U_i est un bruit blanc et f_i périodique de périodes connues T_j , diviseurs entiers de n .

$$T_j = n/k_j \quad j = 1, 2, \dots, q$$

$$\{k_1, k_2, \dots, k_q\} \subset \{1, 2, \dots, \text{ent}[\frac{n-1}{2}]\}$$

f_i peut être représentée par sa transformée de Fourier :

$$f_i = \alpha_0 + \sum_{j=1}^q \left(\alpha(k_j) \cos \frac{2\pi k_j}{n} i + \beta(k_j) \sin \frac{2\pi k_j}{n} i \right)$$

Les coefficients $a(k_j)$, $b(k_j)$ ajustent $\alpha(k_j)$ et $\beta(k_j)$ au sens des moindres carrés (LIN 63), ils minimisent l'expression $h(k)$ qui représente la somme des carrés des erreurs :

$$h(k) = \sum_{i=1}^n \left(x_i - a_0 - \sum_{j=1}^q \left(a(k_j) \cos \frac{2\pi k_j}{n} i + b(k_j) \sin \frac{2\pi k_j}{n} i \right) \right)^2$$

Les estimations des différents coefficients $\alpha_0, \alpha(k_j)$ et $\beta(k_j)$ sont :

$$a_0 = \frac{1}{n} \sum_{i=1}^n x_i = \bar{x}$$

$$a(k_j) = \frac{2}{n} \sum_{i=1}^n x_i \cos \frac{2\pi k_j}{n} i \quad j = 1, 2, \dots, q$$

$$b(k_j) = \frac{2}{n} \sum_{i=1}^n x_i \sin \frac{2\pi k_j}{n} i \quad j = 1, 2, \dots, q$$

Ce sont des réalisations de variables aléatoires $A(k_j)$ et $B(k_j)$ qui sont mutuellement indépendantes.

Les périodes T_j étant des diviseurs entiers de n , après une suite de calculs faisant intervenir les propriétés des séries $\cos \frac{2\pi k_j}{n} i$ et $\sin \frac{2\pi k_j}{n} i$ que nous ne détaillons pas, on obtient :

$$h(k) = \sum_{i=1}^n x_i^2 - n\bar{x} - \frac{n}{2} \sum_{j=1}^q [a^2(k_j) + b^2(k_j)]^2$$

Dans l'hypothèse où f_i comporte une seule période T , $h(k)$ se réécrit :

$$h(k) = \sum_{i=1}^n x_i^2 - n\bar{x} - \frac{n}{2} (a^2(k) + b^2(k))$$

Minimiser $h(k)$ revient à maximiser la quantité $r^2(k) = a^2(k) + b^2(k)$ (réalisation de variable aléatoire $R^2(k)$).

Le graphe de la fonction $R^2(k)$ estimée pour des valeurs k allant de 1 à $\text{ent}[\frac{n}{2} - 1]$ est appelé périodogramme de Schuster.

Il est possible d'essayer d'interpréter ce graphe en déterminant k^* tel que $r^2(k^*)$ soit maximal. La période cherchée est alors $T^* = n/k^*$.

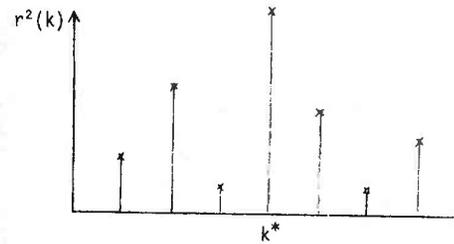


Fig 3.5 : Exemple de périodogramme

Cette interprétation est rarement simple (COX 69, AND 71, PHL 79) : cela suppose que la série chronologique est strictement périodique. Ceci est rarement le cas : très souvent, de nombreux phénomènes aléatoires ou

ou non amènent des perturbations dans les cycles. En outre, son allure est généralement heurtée : chaque point correspond à la contribution d'une fréquence particulière.

IV-4.4.- Filtre de Buys-Ballot (AND 71)

La méthode proposée consiste à chercher un ajustement de f_i supposée strictement périodique de période T , en faisant l'hypothèse que n/T est un entier. (Si ce n'est pas le cas, on peut éliminer les premières mesures).

Supposons $n = Th$, on peut arranger les observations dans le filtre de Buys-Ballot. C'est un tableau à 2 entrées dont le nombre de colonnes correspond à la périodicité présumée. Chaque ligne contient les différents éléments d'une période.

| | | | |
|----------------|----------------|-----|----------|
| x_1 | x_2 | ... | x_T |
| x_{1+T} | x_{2+T} | ... | x_{2T} |
| . | . | . | . |
| . | . | . | . |
| $x_{(h-1)T+1}$ | $x_{(h-1)T+2}$ | ... | x_{hT} |
| y_1 | y_2 | | y_T |

Pour chaque colonne est calculée la moyenne des éléments qui lui appartiennent. On obtient alors une nouvelle série y_1, y_2, \dots, y_T obtenue à partir de x_1, x_2, \dots, x_n avec

$$y_i = \frac{1}{h} \sum_{j=0}^{h-1} x_{i+jT} \quad i = 1, 2, \dots, T$$

L'erreur quadratique $R(T)$ obtenue en ajustant la série $\{x_i\}$ par la série $\{y_i\}$ est :

$$R(T) = \sum_{i=1}^{hT} (x_i - y_i)^2 = \sum_{i=1}^T \sum_{j=0}^{h-1} (x_{i+jT} - y_i)^2$$

Sachant que f_i est strictement périodique, on peut considérer les observations x_{i+jT} pour $i = 1, 2, \dots, T$ comme des réalisations de variables aléatoires mutuellement indépendantes. Ceci nous permet d'utiliser les propriétés de variance :

$$\sum_{j=0}^{h-1} (x_{i+jT} - y_i)^2 = \sum_{i=1}^T (x_{i+jT})^2 - \sum_{i=1}^T (y_i)^2$$

D'où :

$$R(T) = \sum_{j=0}^{h-1} \sum_{i=1}^T (x_{i+jT})^2 - \sum_{j=0}^{h-1} \sum_{i=1}^T (y_i)^2$$

$$R(T) = \sum_{i=1}^n x_i^2 - h \sum_{i=1}^T (y_i)^2$$

$R(T)$ mesure la variance résiduelle non expliquée par la périodicité. On peut tracer le graphe de cette fonction en faisant varier T de 1 à $\text{ent}[\frac{n-1}{2}]$. Pour trouver la période T^* , on recherche la valeur minimale de la fonction $R(T)$.

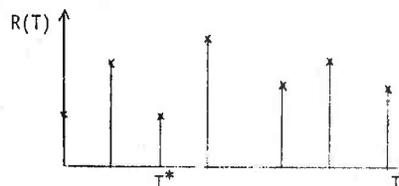


Fig 3.6 : Exemple de graphe des résidus en appliquant le filtre de Buys-Ballot

L'interprétation de ce graphe est toujours délicate. Lorsqu'il y a plusieurs sommets, il est difficile de choisir. Les différents tests existants nécessitent des hypothèses sur les aléas (AND 71).

IV-4.5.- Le Spectrogramme (HAN 60, COX 69, AND 71)

Le Spectrogramme est le graphe de la densité spectrale f estimée pour les fréquences : $\lambda = 2\pi p/n$ $p = 1, 2, \dots, \text{ent}[n/2]$.

Utilisant un estimateur lissé de f (voir 2.1.5), l'interprétation de ce graphe est plus aisée. L'estimation pour une fréquence correspond à une moyenne pondérée sur un certain nombre de fréquences voisines.

Ainsi l'existence de composantes cycliques se traduit par l'existence de pics élevés même si la période est faible. Nous le montrons sur un exemple tiré de (HAN 60) en considérant deux graphes : corrélogramme et spectrogramme calculés sur la même série (fig 3.7). Bien que contenant la même information, le spectrogramme est plus simple à interpréter pour déterminer les harmoniques (ou cycles) principales.

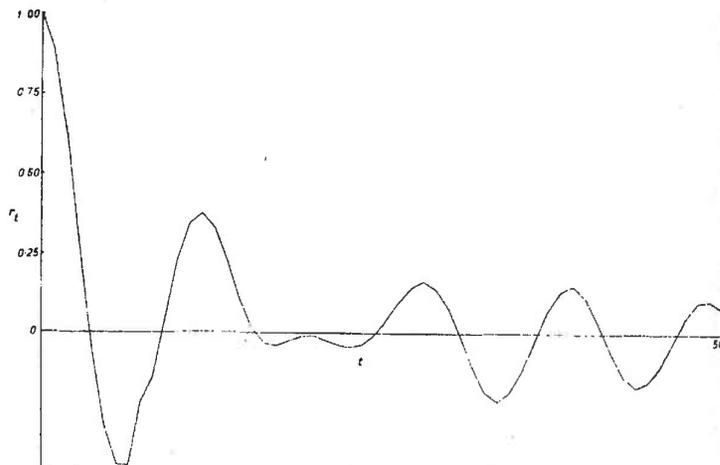


Fig 3.7a : Corrélogramme

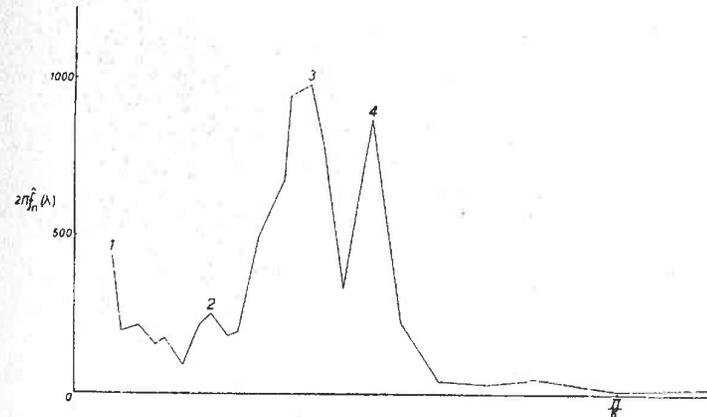


Fig 3.7b : Spectrogramme associé

Dans le cadre de notre travail, nous avons écrit diverses procédures permettant d'obtenir les divers graphes présentés ci-dessus (chapitre V).

IV-5.- Recherche de caractéristiques sur le processus stochastique sous-jacent

Dans cette étape, nous essayons de vérifier certaines propriétés du processus stochastique sous-jacent : stationnarité, existence de périodicités, dépendance temporelle entre les différentes observations.

IV-5.1.- Stationnarité

Seul le caractère faiblement stationnaire d'un processus peut être testé (1.2 et 2.2) en utilisant un test basé sur une analyse spectrale évolutive (PRS 69).

Pour notre part, nous considérons comme stationnaires des séries qui ne présentent pas de tendance en suivant (BEN 66, COX 69).

Nous examinons maintenant quelques méthodes sur la recherche de tendance et leur éventuelle suppression.

a) Recherche de tendance

Dans la pratique, cette recherche peut s'effectuer selon deux types de techniques (le choix s'effectuant après avoir observé les données) :

- Analyse de régression (BEN 66, AND 71)

Si on considère que la tendance peut être approchée par un polynôme de degré q , on utilise le modèle général $X_i = f_i + U_i$ (4.2) où f_i est un polynôme

$$f_i = \sum_{j=0}^q \alpha_j i^j \quad q \ll n$$

Les estimations a_j des coefficients α_j sont obtenues selon la méthode des moindres carrés (LIN 63) : minimisation de la somme du carré des erreurs.

$$\sum_{i=1}^n \left(x_i - \sum_{j=0}^q \alpha_j i^j \right)^2$$

L'utilisation de polynômes orthogonaux permet de simplifier le calcul des coefficients a_j (CAR 69, AND 71).

Par ailleurs, il est possible de tester le degré q du polynôme permettant d'interpoler la tendance (AND 71). Pour leur part, BOX et JENKINS suggèrent d'observer le corrélogramme jusqu'à ce que la fonction d'autocorrélation décroisse en valeur absolue (BOX 70).

Après avoir estimé les coefficients α_j , il faut vérifier l'adéquation du modèle retenu. Ceci peut s'effectuer notamment par la détection d'une éventuelle autocorrélation des résidus :

$$\hat{u}_i = x_i - \hat{f}_i \quad i = 1, 2, \dots, n$$

où \hat{f}_i est la tendance estimée. Pour ce faire, on peut utiliser le test proposé par DURBIN et WATSON (DUR 50) qui suppose théoriquement que les résidus sont distribués selon une loi normale. Cependant, des études de robustesse montrent que ce test peut être utilisé lorsque la taille de l'échantillon est suffisamment importante, même si cette hypothèse de normalité n'est pas vérifiée (AIV 70).

Lorsque la tendance n'est pas de type polynomiale, on peut :

- transformer la série initiale pour se ramener à ce type de tendance,
- rechercher d'autres méthodes comme l'utilisation de filtres linéaires (BAR 71),
- approcher la tendance par d'autres fonctions (courbe logistique..).
- Utilisation de tests statistiques

Désirant tester l'hypothèse d'une tendance apparente contre l'hypothèse nulle d'absence de tendance, il faut choisir entre des tests très puissants permettant de détecter des tendances spécifiques et des tests moins puissants utilisables sous un ensemble d'hypothèses très générales.

Lorsque les séries analysées représentent des durées interévénements, les tests habituellement considérés impliquent deux types de modèles (COX 69) :

- l'arrivée des événements suit un processus de Poisson dans lequel le paramètre λ varie de manière simple,
- un processus de renouvellement comme modèle sous-jacent.

Un processus discret est un processus de renouvellement si les variables aléatoires X_1, X_2, \dots, X_n sont mutuellement indépendantes et ont même loi de probabilité pour $i = 2, 3, \dots, n$.

Pour détecter des tendances très générales, on peut utiliser une approche basée sur la statistique non-paramétrique (BEN 66) :

- décomposition de la série en p suites supposées indépendantes de k observations,
- constitution de 2 séries à partir de ces p suites, l'une concernant les moyennes, l'autre le carré des moyennes :

$$\{\bar{y}_i, i = 1, 2, \dots, p\} \quad \{\bar{y}_i^2, i = 1, 2, \dots, p\}$$

- où \bar{y}_i est la moyenne de la $i^{\text{ème}}$ suite.
- utilisation de tests de tendance non-paramétriques sur ces deux séries.

En l'absence de tendance, on peut accepter l'hypothèse de moyenne et variance constante. Ceci nous permet d'accepter l'hypothèse de stationnarité.

Parmi les tests existants, on peut citer ceux portant sur les séquences, le signe des différences (successives ou par rapport à la moyenne) ou ceux de Cox-Stuart ... A ce sujet, on peut consulter divers ouvrages comme ceux de SIEGEL (SIE 56), WALSH (WAL 62) et CONOVER (CON 71). Ce sont des tests très généraux et faciles à mettre en oeuvre. Dans (WAL 62), on peut trouver une étude comparative des puissances de ces tests concernant la recherche de tendance.

b) Élimination de la tendance

Pour se ramener à l'étude d'un processus stochastique stationnaire, la tendance doit être éliminée. Nous avons essentiellement le choix entre deux procédés.

- Approche polynomiale (AND 71)

Ayant estimé la tendance f_i par un polynôme \hat{f}_i de degré q (4.5.1.a), il est possible de considérer la série résiduelle

$$y_i = x_i - \hat{f}_i \quad i = 1, 2, \dots, n$$

- Méthode des différences finies (BOX 70, AND 71)

Lorsque la tendance est relativement bien approchée par un polynôme de degré q , on peut appliquer l'opérateur linéaire différence ∇

$$\nabla x_i = x_i - x_{i-1} \quad i = 2, 3, \dots, n$$

à la série. En appliquant cet opérateur ∇ , $(q+1)$ fois la série obtenue ne comporte plus de tendance.

Habituellement, l'application de ∇ , q fois se fait par itération successive :

$$\nabla^q x_i = \nabla(\nabla^{q-1} x_i)$$

Ceci nous conduit à

$$\nabla^q x_i = \sum_{j=0}^q (-1)^{q-j} C_q^j x_{i+j} \quad i = 1, 2, \dots, n-q$$

La suppression de tendance est une étape intermédiaire importante dans l'analyse d'une série, notamment pour éviter des distortions importantes lors de l'estimation de la densité spectrale (HAN 60, COX 69). Cependant, elle ne doit avoir lieu que si le nombre des observations est suffisamment important afin de pouvoir dissocier entre tendance et fluctuations aléatoires (BEN 66).

IV-5.2.- Existence de périodicité

Comme pour l'étude de la tendance, nous considérons successivement leur mise en évidence et leur suppression.

a) Recherche de composantes cycliques (AND 71)

Nous considérons uniquement le cas de périodicités entières. La recherche de telles périodicités s'effectue à partir du modèle $X_i = f_i + U_i$ où U_i est un bruit blanc et f_i peut être représentée par une combinaison linéaire de fonctions trigonométriques (décomposition en série de Fourier).

Pour déterminer une période T , nous analysons tout d'abord les divers graphes vus précédemment (périodogramme et spectrogramme). Ensuite, il est possible d'approfondir l'analyse pour vérifier si la période mise en évidence n'est pas due à des fluctuations accidentelles en utilisant des tests statistiques :

- tests paramétriques

- test de Fisher, plus communément appelé test du périodogramme cumulé (HAN 60, AND 71, FIS 73).
- test de moyenne égale dans une analyse de variance en considérant l'hypothèse nulle $f_1 = f_2 = \dots = f_T$, (AND 71).

- tests non paramétriques

On peut utiliser des tests non paramétriques (WAL 62), mais ils sont généralement peu puissants.

Lorsqu'une série comporte une tendance et une composante cyclique indépendantes, on ne peut estimer directement la composante périodique sauf si la tendance peut être approchée par une fonction affine ou linéaire. Dans ce dernier cas, on peut ajuster les paramètres du modèle selon la méthode des moindres carrés. Dans les autres cas, il est souhaitable de rechercher en premier l'existence de tendance pour la supprimer si nécessaire (BEN 66).

b) Suppression des composantes cycliques (AND 71, BAR 71)

Pour supprimer la périodicité T d'une chronique, il est possible d'utiliser un opérateur de moyenne mobile.

Un opérateur de moyenne mobile sur $(m+p+1)$ termes et de coefficients a_j ($j = -m, \dots, p$) est un opérateur linéaire L :

$$L : \mathbb{R}^n \longrightarrow \mathbb{R}^{n-m-p-1}$$

$$(x_i)_{1 \leq i \leq n} \longmapsto (y_i = \sum_{j=-m}^p a_j x_{i+j})_{m+1 \leq i \leq n-p}$$

vérifiant la contrainte suivante sur les coefficients a_j :

$$a_j \geq 0 \text{ et } \sum_{j=-m}^p a_j = 1$$

Sauf cas particuliers, on utilise des coefficients égaux en prenant $m = p$. Lorsque le nombre de termes est impair, on obtient une moyenne mobile centrée et équipondérée. Ce type de procédé est habituellement employé pour "désaisonnaliser" les séries économiques.

IV-5.3.- Corrélation sériale (HAN 60, COX 69, AND 71)

Afin d'identifier le processus stochastique sous-jacent à la série analysée, il est important de connaître les dépendances temporelles existant entre les diverses observations.

Pour tester l'hypothèse nulle que le processus stochastique est une famille de variables aléatoires mutuellement indépendantes, on peut prendre comme hypothèse contraire que les différentes variables aléatoires sont corellées.

Pour déterminer les dépendances temporelles, on observe tout d'abord le graphe des données de la série et le corrélogramme correspondant. Ceci est généralement suffisant.

Cependant, cette étude peut être complétée en utilisant divers tests statistiques. Pour tester une autocorrélation d'ordre 1, on peut utiliser le test de Von Neumann ou un test basé sur le premier coefficient d'autocorrélation circulaire (HAN 60). Ces tests faciles à utiliser sont construits à partir de processus auto-régressifs d'ordre 1 (voir 4.6.2). Plus généralement, on peut obtenir des indications sur la corrélation sériale de la série en utilisant une estimation du coefficient d'autocorrélation d'ordre k et en testant si sa valeur théorique est significativement différente de 0 (COX 69). La théorie relative à la plupart de ces tests considère des modèles auto-régressifs (voir 4.6.2) (AND 71) : ainsi pour un processus auto-régressif d'ordre p , $\gamma(k)$ est égal à 0 pour $k > p$. Pour mettre en oeuvre ces tests, il suffit d'estimer les paramètres du modèle retenu (AND 71).

Parmi les séries que nous avons eues à analyser (IV.2.4), aucune ne présentait de tendance ou périodicités. Aussi nous nous sommes contentés d'examiner les divers graphes présentés en complétant l'étude par des tests statistiques (V.2). Nous n'avons pas eu à utiliser les méthodes concernant la suppression de telles caractéristiques.

IV-6.- Identification du processus stochastique sous-jacent

Dans cette partie, nous supposons le processus stochastique sous-jacent stationnaire. Ceci nécessite éventuellement des transformations sur la série comme la suppression de tendance ou périodicités.

Compte tenu de la série, on est amené à choisir entre différents types de distributions. Quel que soit le modèle probabiliste retenu, la procédure d'identification comporte trois étapes :

- choix d'un modèle,
- estimation des paramètres du modèle,
- vérification des hypothèses faites.

Suivant l'existence ou non d'une dépendance temporelle entre les variables aléatoires constituant le processus stochastique, nous distinguons deux catégories de modèles.

IV-6.1.- Variables aléatoires mutuellement indépendantes (FOU 67)

Les valeurs x_1, x_2, \dots, x_n de la série (éventuellement transformée) peuvent être considérées comme un échantillon de n réalisations mutuellement indépendantes d'une variable aléatoire X dont on cherche une loi de distribution F . Celle-ci est estimée à partir de la fonction de répartition empirique F_n :

$$F_n(x) = \frac{k(x)}{n}$$

où $k(x)$ est le nombre de réalisations de X inférieures à x .

Lorsque l'échantillon est suffisamment important, les fonctions de répartition théorique F et empirique F_n sont très proches.

Nous considérons maintenant la procédure d'identification.

a) Choix d'un modèle

L'utilisation de la statistique descriptive (AIV 70, FIS 73, JAI 78) fournit des caractéristiques sur la distribution statistique :

- la tendance centrale : moyenne, médiane, mode ;
- le degré de dispersion : variance, écart moyen absolu par rapport à la moyenne, coefficient de variation, étendue ;
- la forme de la distribution : coefficients d'aplatissement et de dissymétrie ;
- la fonction de répartition empirique : histogramme ou diagramme

Nous résumons ces caractéristiques dans un tableau (fig 3.8).

| | Définition | Signification | Estimateurs |
|---|---|--|--|
| moyenne | $u = E(X)$ | valeur moyenne | $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$ |
| médiane | $x_0 : P[X < x_0] = P[X > x_0] = \frac{1}{2}$ | valeur médiane | $X_{(n+1)/2}$ n impair $\frac{1}{2}(X_{n/2} + X_{n/2+1})$ n pair |
| mode | $x_0 : dF(x_0) = \max_x dF(x)$ | valeur la plus probable | $\frac{b_{j^*} + b_{j^*+1}}{2}$ avec j^* tel que $H_{j^*} = \max_j H_j$ |
| variance | $\sigma^2 = E[(X - E(X))^2]$ | dispersion par rapport à la moyenne | $S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$ |
| écart moyen absolu par rapport à la moyenne | $m = E[X - E(X)]$ | valeur moyenne de la dispersion | $M = \frac{1}{n-1} \sum_{i=1}^n X_i - \bar{X} $ |
| coefficient de variation | $C^2 = \frac{\sigma^2}{u^2}$ | rapport de la variance par le carré de la moyenne | $C^2 = S^2 / (\bar{X})^2$ |
| étendue | $E = \text{SUP}(X) - \text{INF}(X)$ | écart maximum entre les observations | $E = \text{SUP}(X_i) - \text{INF}(X_i)$ |
| coefficient de dissymétrie | $\gamma_1 = \frac{E[(X-u)^3]}{\sigma^3}$ | mesure la symétrie de la densité | $\hat{\gamma}_1 = \frac{1}{n} \sum_{i=1}^n \frac{(X_i - \bar{X})^3}{S^3}$ |
| coefficient d'aplatissement | $\gamma_2 = \frac{E[(X-u)^4]}{\sigma^4} - 3$ | mesure l'aplatissement de la densité | $\hat{\gamma}_2 = \frac{1}{n} \sum_{i=1}^n \frac{(X_i - \bar{X})^4}{S^4} - 3$ |
| histogramme | $F_n(b_{j+1}) - F_n(b_j) = H_j$ $b_j < b_{j+1} \quad j = 0, 1, \dots, k$ | nombre de réalisations dans la classe $[b_j, b_{j+1}[$ | $H_j = \frac{1}{n} \sum_{i=1}^n I_j(X_i)$ $I_j(X_i) = 1$ si $b_j \leq x_i < b_{j+1}$ 0 sinon |

Fig 3.8 : Caractéristiques d'une loi de distribution (adapté de FIS 73)

l'échantillon x_1, x_2, \dots, x_n est ordonné ($x_i \leq x_{i+1}$), c est une réalisation des variables aléatoires X_1, X_2, \dots, X_n .

En observant l'histogramme, on peut être amené à choisir :

- soit une loi de distribution connue (binomiale, normale à une ou plusieurs dimensions, gamma,...),
- soit une loi dont il faut estimer la densité à partir d'un histogramme. Différentes méthodes existent pour construire un estimateur de la densité comme celles des fonctions orthogonales (WAH 71) ; nous n'en dirons rien de plus ici,
- soit une loi dont la densité est une combinaison linéaire de densités connues.

$$\varphi(x) = \sum_{i=1}^k c_i f_{\lambda_i}(x) \quad \forall x \quad (1)$$

- où :
- k est le nombre de densités connues,
 - $\{f_{\lambda_i} | 1 \leq i \leq k\}$ est une famille de lois de probabilité à laquelle appartient chaque distribution f_{λ_i} ,
 - $\{c_i | 1 \leq i \leq k\}$ probabilité a priori d'apparition d'une réalisation suivant cette loi.

Généralement, (SCH 75), on se fixe a priori le nombre de composants intervenant dans la densité. Lorsqu'on a peu d'informations, mais beaucoup d'observations, on peut choisir la famille des lois normales. Si les réalisations de l'échantillon sont essentiellement positives et si l'histogramme est nettement asymétrique, on peut utiliser une famille de loi gamma ou du χ^2 ,...

b) Estimation des paramètres

Les paramètres d'une loi connue peuvent être estimés à partir des moments ou par la méthode du maximum de vraisemblance entre autres (FOU 67). Cette dernière donne des estimateurs possédant de bonnes propriétés : non biaisé et de variance minimale.

Dans le cas d'un échantillon hétérogène, les paramètres intervenant dans les distributions peuvent être estimés selon diverses approches. Les plus classiques sont des techniques qui estiment les paramètres à partir des observations en choisissant un modèle a priori (DAY 69). Depuis ces dernières années sont apparues des techniques procédant par approximations successives liées à l'introduction des données : ce sont des tech-

niques de type bayésien (PAT 66, AGR 70) ou d'apprentissage (AGR 70, SCH 75). Parmi toutes ces études, il nous faut citer celle d'A. SCHROEDER qui a conçu et mis en oeuvre une technique itérative de reconnaissance des composants d'un mélange (SCH 75). La méthode proposée permet de calculer les différents paramètres intervenant dans la densité $\varphi(x)$ du mélange (1) connaissant :

- l'échantillon $\{x_i | 1 \leq i \leq n\}$,
- le nombre k de distributions,
- la famille de distributions $\{f_{\lambda_i} | 1 \leq i \leq k\}$

de façon à ce que les vraisemblances calculées en partitionnant l'échantillon en k classes $\{P_i | 1 \leq i \leq k\}$ soient maximales si on considère que P_i est un échantillon distribué suivant la loi de densité f_{λ_i} .

Cette méthode très générale peut s'appliquer sur n'importe quel type de mélange si la famille de distributions retenue est définie explicitement.

c) Vérification des hypothèses faites (COX 69, CON 71)

Après avoir estimé les paramètres de la loi de distribution choisie, il faut vérifier l'adéquation du modèle. Ceci peut se faire en utilisant la théorie classique des tests en testant l'hypothèse nulle H_0 contre l'hypothèse H_1

$$H_0 : F = F_0$$

$$H_1 : F \neq F_0$$

- où :
- F_0 est la fonction de répartition théorique choisie,
 - F est la fonction de répartition de X .

Pour tester l'hypothèse H_0 contre l'hypothèse H_1 , on a le choix entre différents tests : test de Kolmogoroff, test de Von Mises, test du χ^2 . Ces tests sont construits à partir d'une norme D_n , variable aléatoire mesurant l'écart existant entre F_0 et F_n .

Ces tests sont facilement automatisables en utilisant soit les lois limites (test de Kolmogoroff et de Von Mises) soit des approximations (test du χ^2). Pour ce faire, on peut se référer à une étude de R.F. LING sur les différentes approximations utilisables pour automatiser les tests statistiques les plus usuels (LIN 78).

Dans le cadre de ce projet, nous avons écrit des procédures mettant en oeuvre ce qui vient d'être présenté (chapitre V). Nous les avons utilisées pour rechercher des modèles probabilistes concernant les séries que nous avons eues à analyser.

Après avoir considéré le cas de variables aléatoires mutuellement indépendantes (rarement vérifié en analyse de chroniques), nous présentons des modèles plus généraux prenant en compte la dépendance temporelle.

IV-6.2.- Variabiles aléatoires dépendantes (BOX 70, AND 71, AND 75, GRA 77)

Nous considérons maintenant des modèles stochastiques stationnaires linéaires appelés aussi "modèles de BOX et JENKINS". BOX et JENKINS ont popularisé cette classe de modèles en étendant leur usage à des séries non stationnaires et en proposant une méthode pour les identifier (BOX 70).

Dans cette présentation, nous faisons essentiellement référence aux travaux de (BOX 70, AND 71, AND 75, GRA 77).

a) Modèles linéaires simples

- Présentation (BOX 70)

On considère le modèle général suivant :

X est un processus stochastique discret et infini. X_t peut s'écrire comme une combinaison linéaire de variables aléatoires.

$$X_t = A_t + \sum_{i=1}^p \varphi_i X_{t-i} + \sum_{j=1}^q \theta_j A_{t-j} \quad (1)$$

où

- $\{\varphi_i | 1 \leq i \leq p\}$ est une suite réelle finie,
- $\{\theta_j | 1 \leq j \leq q\}$ est une suite réelle finie,
- $\{A_t | t \in \mathbb{N}\}$ est un processus stochastique discret composé de variables aléatoires mutuellement indépendantes et suivant une même loi normale $\mathcal{N}(0, \sigma_A^2)$.

Introduisons l'opérateur linéaire B , opérateur "retard" défini ainsi :

$$BX_t = X_{t-1} \quad \forall t \in \mathbb{N}$$

On peut donc réécrire l'équation (1) satisfaite par X_t :

$$\left(1 - \sum_{i=1}^p \varphi_i B^i\right) X_t = \left(1 + \sum_{j=1}^q \theta_j B^j\right) A_t$$

Introduisons les deux polynômes $\varphi_p(B)$ et $\theta_q(B)$:

$$\varphi_p(B) = 1 - \sum_{i=1}^p \varphi_i B^i$$

$$\theta_q(B) = 1 + \sum_{j=1}^q \theta_j B^j$$

Alors (1) se réécrit :

$$\varphi_p(B) X_t = \theta_q(B) A_t \quad (2)$$

Cette équation (2) définit un processus mixte d'ordre (p, q) appelé plus généralement processus ARMA (p, q) , (processus autorégressif à moyenne mobile d'ordre p et q).

Si $q = 0$, (2) est l'équation d'un processus AR(p), (processus autorégressif d'ordre p).

Si $p = 0$, (2) est celle d'un processus MA(q), (processus à moyenne mobile d'ordre q).

X est un processus linéaire stationnaire si le polynôme $\varphi_p(B)$ est inversible (toutes ses racines sont extérieures au cercle unité). Il est dit inversible si le polynôme $\theta_q(B)$ est lui-même inversible.

- Propriétés

• processus AR(p) : si $\varphi_p(B)$ est inversible, X est un processus stochastique stationnaire de moyenne nulle dont la fonction d'autocovariance $\gamma(k)$ vérifie le système d'équations de Yule-Walker :

$$\gamma(k) = \sum_{i=1}^p \varphi_i \gamma(i-k) \quad k = 1, 2, \dots, p$$

• processus MA(q) : X est un processus stochastique stationnaire dont la fonction d'autocovariance $\gamma(k)$ se définit ainsi :

$$\gamma(k) = \begin{cases} \sigma_A^2 \sum_{j=0}^{q-k} \theta_j \theta_{j+k} & \text{si } k \leq q \text{ en posant } \theta_0 = 1 \\ 0 & \text{sinon} \end{cases}$$

• processus ARMA(p,q) : Si $\varphi_p(B)$ est inversible, X est un processus stochastique stationnaire entièrement déterminé par la connaissance du vecteur des paramètres $(\varphi_1, \varphi_2, \dots, \varphi_p, \theta_1, \theta_2, \dots, \theta_q)$.

b) Identification de modèle linéaire

La méthode proposée par BOX et JENKINS est une méthode itérative comportant trois étapes :

- i) choix d'un modèle,
- ii) estimation du vecteur $(\varphi_1, \varphi_2, \dots, \varphi_p, \theta_1, \theta_2, \dots, \theta_q)$,
- iii) vérification de l'adéquation du modèle.

- Choix d'un modèle

Il est effectué en analysant deux fonctions :

- la fonction d'autocovariance $\gamma(k)$,
- la fonction d'autocorrélation partielle $\phi(k)$ (***),

(***) La fonction d'autocorrélation partielle $\phi(k)$ mesure l'autocorrélation existant entre X_t et X_{t-k} . Elle est définie par :

$$\phi(k) = |P_k^*| / |P_k|$$

- où - $|P_k|$ est le déterminant de la matrice des coefficients d'autocorrélation $P_{k;i,j} = \rho(|i-j|)$ $i, j \in [1, k]$,
- $|P_k^*|$ est le déterminant de la matrice P_k^* identique à P_k sauf pour la dernière colonne qui est le vecteur des coefficients d'autocorrélation $P_{k;i,j}^* = \begin{cases} \rho(|i-j|) & \text{si } j \neq k \\ \rho(i) & \text{si } j = k \end{cases}$
- $\rho(i)$ est le coefficient d'autocorrélation de rang i.

Pour estimer $\phi(k)$, on peut remplacer les coefficients d'autocorrélation intervenant dans la définition par leurs estimateurs. Les différentes estimations de $\phi(k)$ peuvent être calculées facilement en prenant la méthode de Durbin (DUR 60).

et en prenant en compte les considérations suivantes (tableau fig 3.9) concernant l'estimation de ces deux fonctions $\widehat{\gamma}(k)$ et $\widehat{\phi}(k)$.

| processus | $\widehat{\gamma}(k)$ | $\widehat{\phi}(k)$ |
|-----------|--------------------------|--------------------------|
| AR(p) | décroissante | ≈ 0 pour $k > p$ |
| MA(q) | ≈ 0 pour $k < q$ | décroissante |
| ARMA(p,q) | décroissante | décroissante |

Fig 3.9 : Récapitulatif sur les fonctions $\widehat{\gamma}(k)$ et $\widehat{\phi}(k)$

Concernant le choix d'un modèle, différents critères ont été suggérés (CHA 77), en particulier pour déterminer l'ordre d'un processus AR(p), (McC 78). L'ordre (p,q) du modèle retenu peut être testé en utilisant les lois limites des estimateurs $\widehat{\gamma}(k)$ et $\widehat{\phi}(k)$. Généralement des modèles tels que $p+q < 2$ sont satisfaisants (BOX 70, AND 75).

Si on a à choisir entre deux modèles, il faut utiliser un critère de parcimonie pour retenir le modèle comportant le minimum de paramètres à estimer.

- Estimation des paramètres du modèle (BOX 70, AND 75)

Le vecteur $(\varphi_1, \varphi_2, \dots, \varphi_p, \theta_1, \dots, \theta_q)$ peut être estimé selon une technique de régression non linéaire en minimisant l'expression :

$$S(\varphi, \theta) = \sum_{i=1}^n \alpha_i^2$$

où $\alpha_i = [\theta_q(B)]^{-1} \varphi_p(B) x_i$ représente l'écart entre le modèle et la série.

La série $\{\alpha_i\}$ qui minimise $S(\varphi, \theta)$ est la série $\{\widehat{a}_i\}$, souvent appelée série des résidus :

$$\widehat{a}_i = x_i - \widehat{x}_i \quad \text{où } \widehat{x}_i \text{ représente l'estimation obtenue.}$$

Considérons le cas général d'un processus ARMA (p,q) de moyenne nulle. Si $E(X_t) \neq 0$, on considère alors le processus $\{X_t - \hat{u}\}$, \hat{u} étant une estimation de $E(X_t)$. La série $\{\alpha_i\}$ vérifie la relation suivante

$$\alpha_i = x_i + \varphi_1 x_{i-1} + \dots + \varphi_p x_{i-p} + \theta_1 \alpha_{i-1} + \dots + \theta_q \alpha_{i-q}$$

Pour estimer cette suite $\{\alpha_i\}$, il est donc nécessaire de connaître les valeurs $\alpha_0, \alpha_1, \dots, \alpha_{1-q}$ et x_0, \dots, x_{1-p} pour calculer α_1 . Ces valeurs n'étant pas toujours connues, on a le choix entre deux solutions :

- choisir des valeurs initiales nulles,
- poser $\alpha_1 = \alpha_2 = \dots = \alpha_p = 0$ et commencer à calculer α_{p+1} .

La première méthode donne de bons résultats si les racines du polynôme $\varphi_p(B)$ ne sont pas trop proches du cercle unité, sinon il est conseillé de choisir la seconde qui entraîne la "perte" de p termes sur les n termes que comporte la série. Dans le cas d'un processus MA(q), ces deux procédures sont identiques (CHA 77).

Très souvent, pour estimer les paramètres d'un processus AR(p), on peut utiliser une procédure basée sur les équations de Yule-Walker (FIS 73) qui est beaucoup plus simple que les procédures itératives servant à minimiser des fonctions non linéaires.

Ceci explique que les modèles AR(p) sont parfois préférés aux modèles ARMA(p,q) ou MA(q), même si le nombre de paramètres à estimer est plus élevé. Cependant, avec les différentes techniques d'optimisation non linéaire (comme celles citées dans (ABA 79)), on peut appliquer sans trop de difficultés un critère de parcimonie pour choisir un modèle.

- Vérification

Après avoir choisi un modèle et estimé ses paramètres, il faut vérifier l'adéquation du modèle en analysant la série des résidus $\{\alpha_i\}$ qui doit être une réalisation de variables aléatoires mutuellement indépendantes suivant une loi normale $\mathcal{N}(0, \sigma_A^2)$ (par définition du modèle).

Pour ce faire, diverses méthodes ont été proposées comme :

- réutiliser la même procédure d'analyse sur la série des résidus et

- modifier ainsi l'ordre (p,q) du modèle retenu initialement.
- tester l'autocorrélation existant entre les résidus. A ce propos les articles de recherche abondent en propositions de tests : test du porte-manteau (DAV 79), test du périodogramme cumulé (BOX 70) et bien d'autres encore (cités dans (CHA 77)).

c) Extension des modèles linéaires

En modifiant le modèle de base ARMA(p,q), il est possible de considérer des modèles plus généraux permettant d'approcher des processus non stationnaires mais présentant une certaine homogénéité.

- Modèles intégrés

Ces modèles sont utilisables lorsque la série analysée présente une tendance qui peut être approchée par un polynôme. Ils font intervenir l'opérateur différence ∇ , déjà rencontré et qu'on peut redéfinir par :

$$\nabla = 1 - B$$

où B est l'opérateur 'retard'.

On obtient alors le modèle ARIMA(p,d,q) défini ainsi :

$$\Phi(B)X_t = \theta(B)A_t$$

avec $\Phi(B) = \varphi_p(B)(1-B)^d = \varphi_p(B)\nabla^d$.

Le polynôme $\Phi(B)$ de degré p+d admet la racine 1 avec l'ordre de multiplicité d, les autres racines sont extérieures au cercle unité.

Ainsi, si on applique l'opérateur ∇ , d fois sur la série initiale, on peut appliquer la théorie précédente à la série résultante.

Lorsqu'il existe des sauts aléatoires dans une série apparemment stationnaire, on peut utiliser le modèle suivant :

$$\Phi(B)(X_t + M) = \theta(B)X_t$$

M étant une fonction correctrice ce qui revient à supprimer les valeurs "aberrantes" en les corrigeant de manière adéquate.

- Modèles saisonniers

Ces modèles peuvent tenir compte de l'existence de composantes cycliques. Ainsi pour modéliser une série de période T, on introduit deux nouveaux opérateurs.

$$\phi_p(B^T) = 1 - \phi_1 B^T - \dots - \phi_p B^{Tp}$$

$$\theta_q(B^T) = 1 + \theta_1 B^T + \dots + \theta_q B^{Tq}$$

et l'opérateur saisonnier différence, noté ∇_T .

$$\nabla_T = 1 - B^T$$

On obtient alors le modèle SARIMA d'ordre $(P,D,Q)_T$:

$$\phi_p(B^T) \nabla_T^D x_t = \theta_q(B^T) A_t$$

Un processus stochastique satisfaisant cette équation a une fonction d'autocorrélation γ et une fonction d'autocorrélation partielle ϕ qui sont identiques à celles du modèle ARIMA correspondant en remplaçant T par 1, et en considérant les observations séparées par une période T.

- Modèles généraux

En combinant les deux modèles précédents, on obtient le modèle multiplicatif $(p,d,q) \times (P,D,Q)_T$ défini ainsi :

$$\phi_p(B) \phi_p(B^T) \nabla_T^d \nabla_T^D x_t = \theta_q(B) \theta_q(B^T) A_t$$

Ces modèles permettent d'"ajuster" un grand nombre de séries chronologiques. Cependant, il faut remarquer qu'une telle procédure d'identi-

fication n'est pas toujours facile à mettre en oeuvre. Cependant, BOX et JENKINS racontent que l'ordre $p+q$ des modèles ARMA (p,q) retenus dépasse rarement 2 et que d est généralement inférieur à 2. Par ailleurs, ils ne proposent aucune méthode ou critère pour choisir T qui est supposée connue.

Actuellement, ces modèles sont largement développés, notamment pour faire de la prévision (ABA 79). De manière générale, ils permettent de trouver des modèles acceptables (CHA 77).

Ce type de modèles est très intéressant dans un domaine comme le nôtre, où il est nécessaire de pouvoir "ajuster" des séries.

Dans le cadre de notre travail, les séries que nous avons eues analyser ne présentaient aucune dépendance temporelle. Ceci explique que nous n'ayons pas mis en oeuvre cette procédure de BOX et JENKINS. Cependant, il nous paraît indispensable de la présenter pour l'intérêt que pourrait présenter cette méthode dans d'autres études du même type.

IV-7.- Conclusion

Comme nous pouvons le constater, l'analyse de séries chronologiques pose de nombreux problèmes qui ne sont pas toujours faciles à résoudre. Dans (CHA 77), on peut trouver une étude concernant l'évolution de la recherche en ce domaine.

Par ailleurs, ces études théoriques se concrétisent par la réalisation de "produits programmes" qui analysent des séries de type relativement spécifique. Dans le domaine économique TSP (TSP 76) met en oeuvre diverses techniques d'analyse. En ce qui concerne l'analyse de séries d'événements, P.A.W. LEWIS est à l'origine de "SASE IV" (LEW 76) qui automatise certaines techniques d'analyse proposées dans (COX 69).

Dans ce chapitre, nous avons exposé différentes méthodes d'analyse de séries en précisant celles déjà mises en oeuvre dans le cadre du projet MAESTRO, bien que toutes soient intéressantes dans notre domaine.

Dans la suite de ce travail, nous montrons comment elles interviennent (chapitre IV) et de quelle manière elles ont été mises en oeuvre (chapitre V).

CHAPITRE IV

MÉTHODOLOGIE D'ÉVALUATION QUANTITATIVE
DU COMPORTEMENT DYNAMIQUE D'UN
SYSTEME D'INFORMATION

I - OBJECTIFS

Quand on a pu construire une maquette logicielle, en supposant qu'elle décrive "correctement" le système qu'on se propose d'étudier, on peut l'utiliser pour évaluer le système réel en situation de fonctionnement en exécutant la maquette lors d'une session de simulation à événements discrets.

Pour mener cette étude, il est nécessaire de fixer un ou plusieurs critères pour apprécier :

- les performances du système lui-même,
- la satisfaction des usagers.

Pour cela, les critères retenus doivent être :

- représentatifs du fonctionnement du système dans son ensemble,
- fonction des "travaux" demandés au système.

De plus, ils doivent avoir des estimations significatives du point de vue statistique et calculables à partir de mesures faciles à obtenir.

Si nous reprenons la définition d'une maquette (II-1), ensemble de "processus coopérants par l'intermédiaire de structures de données communes partageables et concurrents pour l'obtention des ressources physiques", nous choisirons des indicateurs permettant d'apprécier quantitativement :

- les données traitées,
- l'utilisation des ressources physiques : ce sont elles qui exécutent tout ou partie d'un processus ; leur nombre étant limité, le comportement des processus est lié à leur utilisation (durée d'exécution, blocage en cas d'indisponibilité),
- les temps de réponse.

Pour ce faire, nous pouvons choisir des critères concernant :

- le volume des données traitées,
- les taux d'utilisation des ressources,
- les temps de réponse.

II - METHODOLOGIE

Comme dans toute étude, il est nécessaire d'avoir une méthodologie. Nous adoptons celle concernant la simulation à événements discrets. Nous rappelons sur un schéma (fig. 4.1) les différentes étapes qui la composent. Chacune pose divers problèmes que nous évoquons ensuite.

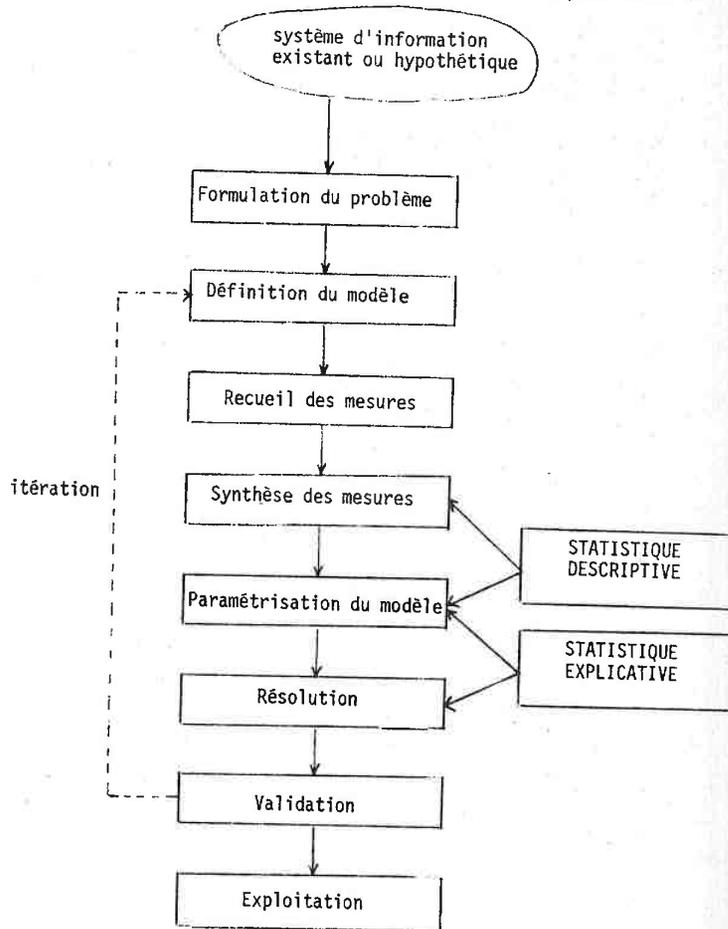


Fig 4.1 : Etapes d'une simulation

II-1.- Formulation du problème

Avant toute chose, il est nécessaire de définir clairement les objectifs d'une étude. Ainsi, lorsqu'on veut construire une maquette, il importe de choisir parmi les trois grands groupes d'objectifs (évaluation structurale, fonctionnelle ou quantitative) (I-1).

Exemple

La maquette prototype présentée au chapitre II (II-3) est essentiellement orientée évaluation quantitative du fonctionnement dynamique.

La construction d'une maquette fournit un modèle simplifié et réduit de la réalité (II-2), tout en conservant les propriétés les plus importantes, par rapport aux objectifs fixés.

Exemple

Dans la maquette prototype, différentes simplifications ont été faites dans la description du système. Nous avons favorisé l'aspect dynamique. Ceci se traduit par une évolution des données relativement fidèle tandis que la description des données elle-même est succincte : quelques attributs (un numéro d'ordre et un nombre de lignes par document par exemple). Nous négligeons toute référence aux clients et aux types de produits. Nous intéressant essentiellement au traitement des données et à leur évolution au cours du temps, une description plus détaillée n'apporterait sûrement pas une meilleure compréhension de leur cheminement, mais risquerait de compliquer l'évaluation quantitative sans pour autant l'améliorer.

Pour ces mêmes raisons, la description des traitements comporte peu de sémantique. Ils sont réduits en traitements élémentaires caractérisés chacun par une durée de service.

II-2.- Etablissement d'un modèle

Après avoir été écrit les composants standard proposés (KLE 79) et rappelés dans le chapitre II, le programme maquette résultant peut être exécuté à condition de le compléter par quelques éléments que nous rappelons sommairement.

II-2.1.- Clôture d'un modèle

Lorsque les relations existant entre le système et son environnement ne sont pas décrites dans le modèle, celui-ci est dit ouvert. Pour pouvoir l'utiliser en simulation, il est nécessaire de le clôre. La clôture d'une maquette peut être faite en ajoutant un ensemble de structures de données, processus ou ressources pour représenter les inter-relations du système avec son environnement.

Exemple

Ainsi dans l'exemple de maquette présenté dans le chapitre II, ce sont quatre processus qui assurent la clôture.

En entrée, les processus cycliques générateur et général alimentent les structures de données courrier et boîte. Ils correspondent à l'arrivée du courrier et des appels téléphoniques.

En sortie, deux autres processus cycliques videur 1 et videur 2 assurent le vidage des files bons 1 et bons 2. Ils modélisent l'envoi des bons de livraison aux clients et à la comptabilité.

Leurs principales caractéristiques sont résumées dans le tableau général relatif aux processus (fig 2.6).

II-2.2.- Paramétrage du modèle

Pour utiliser une maquette en simulation, il est nécessaire de fixer des valeurs aux paramètres décrivant le comportement du système. Celui-ci est déterminé par la composition et la dynamique de :

- l'architecture du système qui peut se définir comme l'ensemble des ressources et leur organisation,
- la charge, c'est-à-dire l'ensemble des données que doit traiter le système.

Chaque fois que c'est possible, il faut distinguer le paramétrage de ces deux parties relativement disjointes : le modèle lui-même (architecture et fonctionnement) et son environnement.

Ainsi pour paramétrer une maquette, nous considérons :

- d'une part, dans le modèle lui-même :

- les lois du cheminement et des modifications des données,
- les durées de service pour tout traitement élémentaire.
- d'autre part dans son environnement
 - les caractéristiques des données pour fixer des valeurs à leurs attributs,
 - leur mode d'arrivée (groupée ou individuelle).

Exemple

Le paramétrage proposé pour la maquette prototype est le suivant :

- les paramètres du modèle :
 - les lois du cheminement : nombre de lignes proposées à la livraison et à livrer par commande.
 - les durées de service : loi des durées de traitements élémentaires concernant la saisie des commandes écrites et des appels téléphoniques (renseignements, commandes), l'élaboration des propositions de livraison, l'examen de ces propositions, la saisie des modifications, l'édition des bons de livraison et leur préparation.
- les paramètres de l'environnement :
 - les caractéristiques des données : probabilité pour qu'un appel téléphonique donne lieu à une commande et loi du nombre de lignes par commande (écrite et téléphonique).
 - les arrivées : loi du nombre de commandes écrites par jour (et heure d'arrivée du courrier) et des inter-arrivées concernant les appels téléphoniques.

L'estimation de ces paramètres peut être obtenue à partir de mesures effectuées sur le système réel décrit par la maquette et sur son

environnement, lorsque c'est possible. Sinon, il faut soit faire des hypothèses extrapolées de la réalité ou basées sur l'intuition soit utiliser des standards lorsqu'ils existent.

Lorsque la phase d'estimation est terminée, il faut compléter la maquette :

- en fixant des valeurs aux paramètres concernant le modèle lui-même,
- en choisissant comment alimenter la maquette pendant la simulation.

Ces problèmes étant très importants, nous les examinons par la suite (2.3 et 2.4) plus en détail.

II-2.3.- Collecte des résultats

Pour satisfaire nos objectifs d'évaluation quantitative du fonctionnement dynamique, il nous faut tout d'abord définir les résultats qu'on veut obtenir. Nous pouvons les considérer selon deux points de vue :

- statique pour obtenir une vue globale du système essentiellement constituée de moyennes calculées lorsque le système est en régime stationnaire (I-2).
- dynamique pour améliorer la compréhension du comportement du système, ce qui peut être réalisé en constituant des séries chronologiques de mesures sur lesquelles nous appliquons des techniques d'analyse décrites précédemment (III-4).

Pour obtenir les résultats souhaités, on peut utiliser deux types d'outils :

- processus ou procédures spécialisés ajoutés au modèle,
- adjonction d'appels de procédures spécialisées.

Par rapport aux objectifs définis au début de ce chapitre (IV-1), nous avons conçu et réalisé divers outils de mesure et d'analyse utilisables lors de la simulation d'une maquette. Nous les présentons dans le chapitre VI.

Pour leur mise en oeuvre, il suffit :

- de valuer convenablement certains paramètres du programme de simulation concernant notamment les ressources, les structures de données ... et la simulation proprement dite (durée de la simulation, précision statistique sur les résultats entre autres),
- d'ajouter éventuellement quelques appels de procédures spécialisées pour obtenir d'autres renseignements (histogrammes..).

II-3.- Collecte et analyse des mesures

Ayant seulement abordé ce problème dans sa généralité (2.2.2), nous développons ces deux aspects étroitement liés.

II-3.1.- Recueil des mesures sur le système réel

Avant d'entreprendre ce travail, il faut :

- déterminer le niveau de détail de l'observation, fonction des résultats souhaités. Sinon, on risque de mener une observation trop fine conduisant à un important volume de données long et coûteux à traiter, en fournissant des résultats susceptibles de masquer les phénomènes les plus importants en les "diluants". La détermination du niveau de détail peut s'obtenir à partir du modèle lui-même ;
- choisir les grandeurs à observer et définir un ensemble de mesures cohérent par rapport à ces grandeurs ;
- définir un plan de mesures pour limiter le nombre des mesures et déterminer un nombre suffisant de mesures.

Exemple

Le paramétrage de la maquette exemple a été obtenu à partir de mesures effectuées dans l'entreprise de référence.

A partir d'archives, nous avons établi des relevés concernant les données et leur flux sur plusieurs mois. Les séries chronologiques obtenues concernent notamment :

- *le nombre de commandes écrites par jour,*

- le nombre d'appels téléphoniques correspondant à une commande,
- le nombre de lignes par commande (téléphonée et écrite),
- la durée séparant deux appels téléphoniques,
- le nombre de bons de livraison journalier,
- le nombre de lignes par bon de livraison.

Ces mesures sont faciles à obtenir, puisque consignées dans des archives. Notre problème fondamental concerne l'appréciation des durées de service. Les difficultés rencontrées sont bien connues en ergonomie ou en organisation scientifique du travail : méfiance des personnes observées, grande disparité des mesures en fonction des personnes et des périodes.

II-3.2.- Synthèse des mesures

C'est une étape indispensable avant celle de modélisation qui nécessite très souvent des données réduites sous forme de modèles probabilistes, notamment pour le fonctionnement du modèle lui-même (loi de distribution sur les transformations des données, les durées de service...).

Pour être utilisable en simulation, tout modèle stochastique retenu doit être :

- le plus proche possible de la réalité (adéquation entre le modèle théorique retenu et les mesures),
- faisable (possibilité de générer des échantillons artificiels suivant le modèle retenu).

Cette détermination de modèles peut s'obtenir par analyse statistique (analyse de séries chronologiques par exemple).

Exemple

Pour paramétrer la maquette exemple, nous avons appliqué des procédés d'analyse statistique exposés dans le chapitre III (III.4) sur les séries chronologiques retenues. Les outils conçus et réalisés à ce propos sont présentés dans le chapitre V.

Nos principales difficultés concernent la détermination de modèles

probabilistes sur les durées de traitement. Nos estimations sont très approximatives et sujettes à caution, et n'ont qu'une valeur subjective. Nous résumons dans un tableau les modèles retenus pour les durées d'exécution (fig 4.2). Ce sont des modèles déterministes sauf pour le processus préparation.

Le nombre de lignes proposées à la livraison suit une loi binomiale de telle sorte que le nombre moyen de lignes en attente ne varie pas significativement au cours du temps.

| PROCESSUS | Traitement | Durée d'exécution |
|----------------|---|---|
| enregistrement | codage saisie | fixe, fonction du nombre de commandes et du nombre de lignes commandées |
| saisie | saisie des appels téléphoniques | fixe pour une demande de renseignements, proportionnelle au nombre de lignes dans le cas d'une commande |
| décision | édition prop. décision liv. saisie modif. édition déf. | fixe pour l'édition des propositions proportionnelle au nombre total de commandes proportionnelle au nombre de lignes en attente fixe pour l'édition des bons de livraison |
| préparation | préparation des bons | aléatoire (loi normale tronquée dont les paramètres sont proportionnels au nombre de lignes à livrer) |

Fig 4.2 : Durée d'exécution des processus

II-4.- Analyse et génération des entrées

Pour exécuter une maquette, il est nécessaire de l'alimenter.

II-4.1.- Entrées du modèle de simulation

On a le choix entre deux types de méthodes :

a) Entrée par trace (FER 78, KOB 78)

Les variables d'entrée sont des échantillons réels prélevés sur l'environnement existant pour alimenter le modèle pendant la simulation : les mesures sont utilisées directement en entrée dans l'ordre de leur apparition.

Cette méthode est utilisable lorsque l'environnement est accessible et mesurable. En outre, elle nécessite un échantillonnage de taille suffisamment importante pour assurer l'alimentation du modèle pendant une certaine durée.

Une telle approche est intéressante lors de la validation (voir 2.6), pour comparer les sorties réelles et simulées. Mais elle ne permet pas de bien comprendre les relations existant entre les paramètres du modèle lui-même et les performances. De plus, elle n'autorise pas l'éventuelle validation par des modèles analytiques (voir 2.6).

Exemple

La maquette exemple peut être alimentée à partir des séries chronologiques suivantes :

- inter-arrivée des appels téléphoniques (en conservant le type d'appel (renseignement ou commande)),
- le nombre de commandes écrites par jour,
- le nombre de lignes par commande téléphonique,
- le nombre de lignes par commande écrite.

b) Entrée par modèle statistique (FIS 73, LER 77, KOB 78)

Il faut construire un modèle stochastique de l'environnement et l'inclure dans la maquette. Ce dernier est alors utilisé pour fournir des échantillons artificiels, obtenus à l'aide de générateurs de nombres pseudo-aléatoires tout au long de la simulation.

Cette approche permet de vérifier certaines propriétés des processus stochastiques sous-jacents aux séries (stationnarité, dépendance temporelle ...).

Cependant pour l'utiliser, il faut pouvoir :

- déterminer des modèles stochastiques,
- générer des nombres pseudo-aléatoires à partir de ces modèles.

Cette solution ayant été retenue pour sa simplicité, nous examinons maintenant les problèmes qu'elle soulève et des méthodes pour les résoudre.

II-4.2.- Détermination de modèles stochastiques de l'environnement

Ce problème est identique à celui du paramétrage du modèle (2.2.2). A partir des mesures réelles, il nous faut obtenir les distributions de probabilité. Ceci se fait en connaissant :

- soit la forme explicite de la distribution de probabilité ou du modèle choisi,
- soit la fonction de répartition empirique lorsque la forme explicite de la distribution est inconnue (et si l'échantillon peut être considéré comme une réalisation de variables aléatoires mutuellement indépendantes).

Exemple

En ce qui concerne la maquette prototype, nous avons recherché des modèles probabilistes pour les entrées.

Ainsi la durée entre deux appels téléphoniques suit une loi exponentielle négative de paramètre $\lambda = 25/8$. Un quart des appels correspond à une commande dont le nombre de lignes suit une loi discrète tabulée.

Le nombre de commandes écrites par jour et leur nombre de lignes suivent des lois discrètes tabulées elles aussi.

Nous présentons, ci-dessous, les histogrammes relatifs à ces distributions (fig 4.3, 4.4, 4.5).

II.4.3.- Génération de nombres suivant une loi donnée (FIS 73, LER 77, KOB 78)

Des suites de nombres pseudo-aléatoires suivant les modèles probabilistes retenus peuvent être générées selon différentes méthodes :

- utilisation de l'échantillon comme entrée directe (c'est une réalisation de processus stochastique inconnu),
- utilisation d'un histogramme cumulé (c'est une tabulation de la fonction de répartition empirique),
- génération par loi inverse (si on connaît la forme explicite de la fonction inverse de la fonction de répartition),
- génération par méthode composée pour générer des nombres suivant une loi dont la forme explicite est connue en utilisant divers algorithmes.

Quelle que soit la méthode utilisée, il est nécessaire de pouvoir générer des échantillons de nombres pseudo-aléatoires suivant une loi uniforme sur [0,1]. La génération de tels échantillons peut être obtenue selon diverses méthodes (KNU 68, FIS 73, LER 77) :

- méthode congruentielle
- méthode des registres à décalages bouclés
- brassage de générateurs aléatoires.

Exemple

Dans la maquette prototype, nous utilisons certaines procédures existant en bibliothèque SIMULA (CII 72) pour générer des suites de nombres pseudo-aléatoires.

Dans chaque application, en fonction d'algorithmes spécifiques et du modèle probabiliste retenu, on peut mettre en oeuvre les méthodes que nous venons de présenter :

Ainsi, dans la maquette exemple (texte en annexe), nous avons écrit quelques procédures comme :

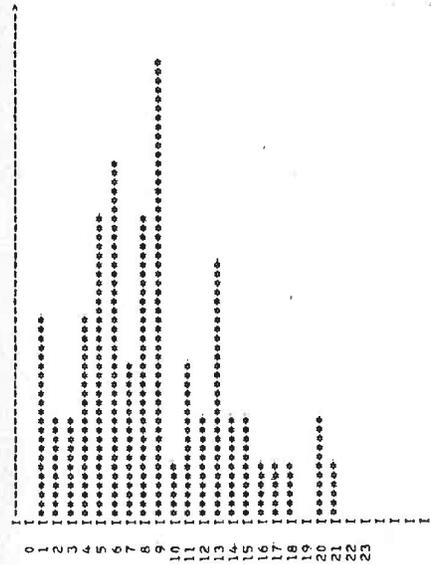
Binomiale (n,p) pour générer des nombres suivant une loi binomiale de paramètres n et p.

Reelnormale (a,b,c) pour générer des nombres suivant une loi normale de moyenne a, d'écart-type b et supérieurs à c.

Genhisto (Th) pour générer des suites de nombres pseudo-aléatoires en utilisant un histogramme cumulé dont les différentes valeurs sont contenues dans un tableau repéré par Th.

Exponent (dur) pour générer des nombres suivant une loi exponentielle négative de paramètre $\lambda = \text{dur}$ en utilisant la loi inverse.

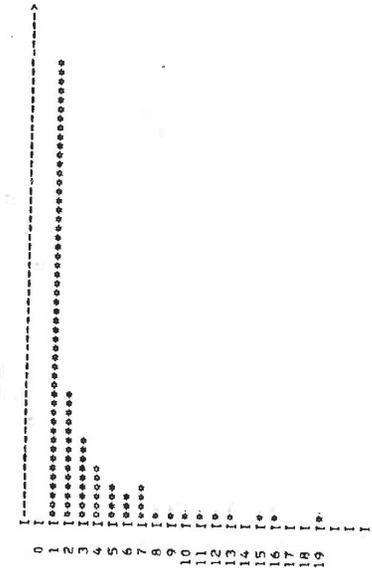
 *** HISTOGRAMME *** NB-COM-ECR-J



TOTAL ENTREES : 64
 VALEUR MINI. : 1.00
 VALEUR MAXI. : 21.00
 LARGEUR INTERVALLE : 1.00
 INTERVALL NO 1 : 1.00 A 2.00
 INTERVALL NO 22 : 22.00 A 23.00

Fig 4.3 : Histogramme du nombre de commandes écrites par jour

 *** HISTOGRAMME *** LG-COM-TEL



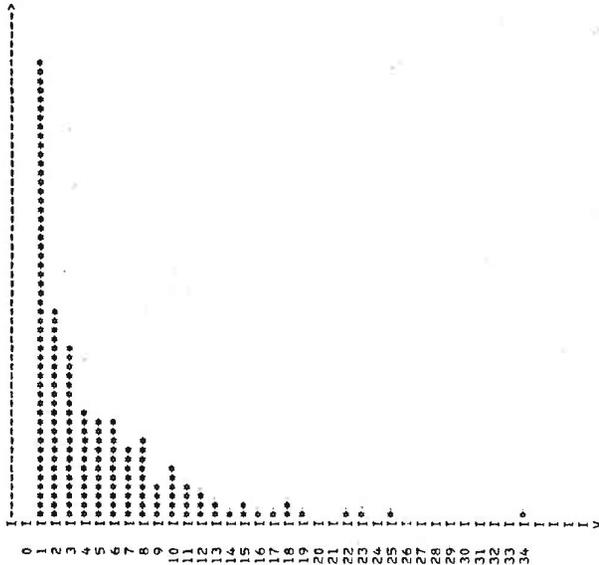
TOTAL ENTREES : 346
 VALEUR MINI. : 1.00
 VALEUR MAXI. : 50.00
 LARGEUR INTERVALLE : 1.00
 INTERVALLE NO 1 : 1.00 A 2
 INTERVALLE NO 18 : 18.00 A 19

Fig 4.4 : Histogramme du nombre de lignes par commande téléphonique

```

*****
*** HISTOGRAMME *** LG-COM-ECQ
*****

```



```

TOTAL ENTREES :      554
VALEUR MINI.  :      1.00
VALEUR MAXI.  :      78.00

LARGEUR INTERVALLE :      1.00
INTERVALLE NO  1 :      1.00 A      2.00
INTERVALLE NO 33 :      33.00 A     34.00

```

Fig 4.5 : Histogramme du nombre de lignes
par commande écrite

II-5.- Analyse des sorties

Souhaitant obtenir des résultats sur les volumes de données, l'utilisation des ressources et les délais de réponse au cours d'une seule simulation, il nous faut être en régime stationnaire pour que les résultats obtenus, généralement des moyennes aient un sens :

Un système est dit en régime stationnaire si la probabilité d'être dans un tel état est une constante. Cette dernière est une probabilité limite obtenue après une simulation suffisamment longue et est indépendante de l'état initial.

Ainsi, nous considérons que :

le système est en régime stationnaire si les volumes de données, les taux d'occupation et les délais de réponse ne varient pas de manière significative au cours du temps.

II-5.1.- Résultats souhaités

L'évaluation de performances est obtenue en effectuant des mesures sur certains composants du modèle. A partir des séries de mesures obtenues, on peut déduire par analyse statistique des grandeurs caractéristiques sur :

a) Volumes des données

Les données communes partageables mises en évidence dans le système réel sont modélisées par des structures de données ($\mathcal{D} = \mathcal{D}, \mathcal{N}$) (II-1.1.1).

Pour suivre l'évolution du système au cours du temps, on définit pour des structures de données une fonction valuation V :

$$V : \mathcal{D} \rightarrow \mathbb{R}^+$$

On suit alors l'évolution du processus stochastique $\{V(t)\} = \{V(D_t)\}$ où D_t est un élément de \mathcal{D} (données à l'instant t).

Il ne nous semble pas nécessaire de considérer toutes les structures de données d'une maquette pour suivre l'évolution du système modè-

lisé. Cependant, il semble important de tenir compte de celles concernant les données :

- issues de l'environnement (entrées du système),
- produites par le système vers l'environnement (sorties du système),
- ayant une "durée de vie" supérieure ou égale à celle de l'application décrite pour vérifier l'absence de "goulots d'étranglements".

Exemple

Dans la maquette exemple, nous considérons les structures de données suivantes :

- les entrées du système : courrier,
- les sorties du système : bons 1,
- "permanentes" : commandes.

Comme fonction valuation, nous avons retenu le nombre de lignes des divers documents modélisés.

b) Occupation des ressources

Concernant les ressources physiques, il est possible :

- d'étudier le processus stochastique $\{N(t)\}$:

$N(t)$ nombre d'accès utilisés à l'instant t .

Suivant le type de ressources (II-1.1.1), $N(t)$ prend ses valeurs dans des ensembles différents :

- ressource critique $\{-\{0,1\}$,
- ressource à n points d'accès $\{-\{0,1,2,\dots,n\}$,
- ressource à nombre infini d'accès $\{-\mathbb{N}$.
- de calculer un taux d'occupation.

c) Temps de réponse

On peut définir le temps de réponse comme le temps mis par tout ou partie du système pour effectuer un ou plusieurs traitements.

Pour suivre l'évolution du système, on est amené à considérer le processus stochastique $\{T(t)\}$

$T(t)$ associé au temps de réponse pour une partie du système.

Pour connaître la durée d'exécution simulée d'un ou plusieurs processus, nous avons défini des régions qui correspondent à des "sous-systèmes".

Une région est déterminée par deux éléments : point (s) d'entrée et point (s) de sortie.

C'est une entité incorporée au modèle pour collecter des résultats statistiques qui n'a aucune influence sur le déroulement de la simulation.

Désirant avoir un mécanisme général, nous avons été amenés à définir des entités standard: les transactions.

Les transactions sont des objets (au sens de SIMULA) qui englobent des entités mobiles du modèle : données ou processus.

Elles servent à mesurer les temps de séjour simulé dans une région.

L'entrée ou la sortie d'une région s'effectuent par leur intermédiaire. Leurs mouvements ne sont soumis à aucune condition.

Cette notion de région est analogue à celles des "queues" dans GPSS (IBM 70) et des "régions" définies par VAUCHER dans GPSS (VAU 77), tandis que celle de transaction est différente.

Remarques

- L'utilisateur définit lui-même les régions qu'il désire.
- Une transaction peut être présente dans plusieurs régions qui ont même point d'entrée. L'entrée d'une transaction dans ces régions a lieu au même instant. (Cette restriction nous permet de calculer le temps de séjour simulé dans les diverses régions).
- Hormis le cas précédent, l'entrée d'une transaction ne peut avoir lieu que si elle n'est déjà présente dans aucune autre région.

Connaissant les temps de séjour des transactions sorties au cours de la simulation, il est possible de calculer un temps de séjour moyen.

Exemple

Dans la maquette exemple, quatre régions ont été définies :

- écrites (codification et saisie des commandes écrites),
- téléphonées (saisie des appels téléphoniques),
- administratif (traitement administratif d'une commande),
- physique (préparation physique d'un bon de livraison).

II-5.2.- Mesures

Pendant l'exécution d'une maquette, on effectue des mesures pour obtenir des estimations de grandeurs caractéristiques (moyennes).

a) Volumes de données

Pour suivre l'évolution du volume d'une structure de données, on la mesure à des instants t_1, t_2, \dots, t_n régulièrement espacés dans le temps

$$t_i - t_{i-1} = \text{constante} \quad i = 1, 3, \dots, n$$

La série chronologique obtenue v_1, v_2, \dots, v_n est une trajectoire du processus stochastique $\{V(t)\}$ et peut être analysée selon diverses méthodes (III-4).

Comme l'application décrite est généralement cyclique, il ne nous a pas semblé utile de conserver tout l'historique des valuations mais plus judicieux de mesurer périodiquement. En effet, ces structures de données sont essentiellement mesurées pour voir l'évolution du système (existence de goulots d'étranglement, tendance ...). Il faut donc que les instants de mesures soient significatifs (mesures de structures de données vides par exemple).

Exemple

Dans la maquette prototype, la périodicité des mesures a été dé-

terminée à partir de trois facteurs :

- le cycle des calendriers des ressources utilisées par le (s) processus producteur (s) ou utilisateur (s),
- le cycle des processus,
- la périodicité de l'application décrite,

Toutes ces périodes étant d'un jour, les structures de données retenues sont mesurées tous les jours et à des heures différentes :

- 9 H 30 : pour courrier, après l'arrivée des commandes,
- 21 H : pour bons 1 et commandes, en fin de journée.

Pour bien appréhender le comportement dynamique du système, ces mesures régulières sont insuffisantes. Elles donnent seulement une vue générale.

C'est pourquoi, nous voulons procéder à des relevés à chaque modification de la structure de données en tenant compte de son type.

Ce procédé n'a encore été mis en oeuvre que pour les files d'attente : nous comptabilisons le nombre d'entrées, de sorties, les contenus maximum et minimum ainsi que le nombre de fois où la file est vide.

Exemple

Dans la maquette exemple, pour chaque file d'attente, l'utilisateur peut obtenir des renseignements sur :

- le nombre de documents (mesure à chaque variation),
- le nombre total de lignes des documents (mesure journalière).

Les résultats présentés ci-dessous sont obtenus après l'exécution de la maquette exemple sur 60 jours simulés en régime stationnaire après une période d'initialisation de 5 jours.

| STRUCT.DONNÉE | MESURES | | EVALUATION DES VOLUMES D'INFORMATION : VOLUME | | | |
|---------------|---------|------------|---|---------|-------|--------|
| | NOMBRE | INTERVALLE | MINIMAL | MAXIMAL | MOYEN | ACTUEL |
| COMMANDES | 60 | 1 0 0 | 45 | 543 | 154 | 335 |
| COURRIER | 60 | 1 0 0 | 1 | 129 | 50 | 49 |
| ORDRES | 60 | 1 0 0 | 0 | 314 | 76 | 314 |
| BONS1 | 60 | 1 0 0 | 0 | 189 | 82 | 115 |

Fig 4.7 : Récapitulatif des volumes d'information
(structures de données)

| FILE (S) | ENTREES | SORTIES | CONTENU DE LA FILE | | | RET A ZERO |
|-----------|---------|---------|--------------------|--------|--------|------------|
| | | | MAXIMUM | MOYEN | ACTUEL | |
| COMMANDES | 904 | 830 | 124 | 32.947 | 68 | 0 |
| COURRIER | 561 | 561 | 20 | 0.603 | 0 | 60 |
| ORDRES | 1070 | 1009 | 61 | 13.159 | 61 | 66 |
| BONS1 | 1009 | 1009 | 40 | 5.587 | 0 | 57 |

Fig 4.8 : Récapitulatif des files d'attente

b) Occupation des ressources

Concernant les ressources physiques, il est possible de calculer un taux d'utilisation τ , égal au rapport de la durée de réservation par la durée de disponibilité de la ressource :

$$\tau = \frac{\int_U n_U(t) dt}{\int_D n_D(t) dt}$$

- où
- U est la période d'utilisation,
 - D est la période de disponibilité,
 - $n_U(t)$ est le nombre d'accès utilisés à l'instant t,
 - $n_D(t)$ est le nombre d'accès disponibles à l'instant t (y compris ceux qui sont utilisés).

Comme le nombre d'accès disponibles est fixe :

$$n_D(t) = n_D \quad \forall t \in D$$

τ peut se réécrire :

$$\tau = \frac{\int_U n_U(t) dt}{n_D \times D}$$

Remarques :

- Dans le calcul de la période de disponibilité D, il faut tenir compte des calendriers associés aux ressources.
- Si le processus stochastique $\{N(t)\}$ est (quasi-)stationnaire, un intervalle de confiance sur le taux d'occupation moyen peut être calculé en utilisant une méthode de blocs (III.3).

Exemple

Dans la maquette exemple, le taux d'utilisation de chaque ressource physique est calculé. Par contre, nous obtenons un intervalle de confiance uniquement pour les ressources z et p.

Les tableaux 4.9, 4.10, 4.11 synthétisent les résultats obtenus dans les conditions énoncées précédemment, suivant différents types de ressources.

Par ailleurs, la présentation de ces résultats est complétée par l'édition de diagrammes récapitulants le nombre d'accès utilisés par période de temps et en pourcentage (fig 4.12, 4.13).

La figure 4.14 donne un intervalle de confiance du taux d'utilisation des ressources z et p à différents seuils α .

| CALRC | PROCESSUS SERVIS | DUREE DISPONIBLE | TAUX UTILISATION | ETAT ACTUEL |
|-------|---------------------|---------------------|---------------------|----------------|
| X | 1122 | 7 12 0 | 0.536 | LIBRE |
| Z | 60 | 3 18 0 | 0.834 | LIBRE |
| P | 1005 | 20 0 0 | 0.528 | OCCUPE |

Fig 4.9 : Récapitulatif de l'utilisation des ressources critiques avec calendrier

| CALRCPP | PROCESSUS SERVIS | DUREE DISPONIBLE | TAUX UTILISATION | ETAT ACTUEL |
|---------|---------------------|---------------------|---------------------|----------------|
| Y | 939 | 20 0 0 | 0.160 | LIBRE |

Fig 4.10 : Récapitulatif de l'utilisation des ressources critiques préemptibles avec priorité et calendrier

| RAMINF | NOMBRE D UTILISATEURS | | | |
|------------|-----------------------|-------|-------|--------|
| | SERVIS | MOYEN | MAXI. | ACTUEL |
| ORDINATEUR | 1646 | 0.05 | 2 | 0 |

Fig 4.11 : Récapitulatif de l'utilisation des ressources à nombre infini d'accès

DIAGRAMME D OCCUPATION DE LA RESSOURCE :ORDINATEUR
TYPE - RAMINF

DUREE DE DISPONIBILITE : 60 0 0
DUREE DE RESERVATION : 2 21 18

| | NB ACCES | PROBA.UTILIS. |
|---|----------|---------------|
| * | | |
| * | | |
| * | | |
| * | 0 | 0.950 |
| * | | |
| * | 1 | 0.044 |
| * | | |
| * | 2 | 0.001 |
| * | | |
| * | 3 | 0.000 |
| * | | |
| * | 4 | 0.000 |
| * | | |
| * | 5 | 0.000 |
| * | | |
| * | 6 | 0.000 |
| * | | |
| * | 7 | 0.000 |
| * | | |
| * | 8 | 0.000 |
| * | | |
| * | 9 | 0.000 |
| * | | |
| * | 10 | 0.000 |
| * | | |

0 1 2 3 4 5 6 7 8 9 10 ET PLUS ACCES

REMARQUE : LA PROBABILITE DONNEE POUR 10 ACCES
EST LA PROBABILITE POUR 10 ACCES ET PLUS

Fig 4.12 : Diagramme d'utilisation des accès de la ressource ord

DIAGRAMME D OCCUPATION DE LA RESSOURCE :Y
 TYPE - CALRCPP - NOMBRE D ACCES : 1



DUREE DE DISPONIBILITE : 20 0 0
 DUREE DE RESERVATION : 3 4 40

| NO ACCES | PROBA.UTILIS. |
|----------|---------------|
| 0 | 0.840 |
| 1 | 0.160 |

Fig 4.13 : Diagramme d'utilisation de ressources critiques :
avec calendrier : z
avec priorité, préemptible et avec calendrier : y

DIAGRAMME D OCCUPATION DE LA RESSOURCE :Z
 TYPE - CALRC - NOMBRE D ACCES : 1



DUREE DE DISPONIBILITE : 3 18 0
 DUREE DE RESERVATION : 3 3 5

| NO ACCES | PROBA.UTILIS. |
|----------|---------------|
| 0 | 0.108 |
| 1 | 0.892 |

 * RESSOURCE NO 4 Z *

NOMBRE DE MESURES FAITES PENDANT LA SIMULATION : 240

NOMBRE DE BLOCS : 60
 TAILLE D UN BLOC : 24

INTERVALLE DE CONFIANCE DU TAUX D OCCUPATION MOYEN
 SEUIL ALPHA = 0.01 (0.695 , 0.970)
 SEUIL ALPHA = 0.05 (0.728 , 0.941)
 SEUIL ALPHA = 0.10 (0.745 , 0.920)

ESTIMATION DU TAUX D OCCUPATION MOYEN : 0.834

 * RESSOURCE NO 5 P *

NOMBRE DE MESURES FAITES PENDANT LA SIMULATION : 2258

NOMBRE DE BLOCS : 60
 TAILLE D UN BLOC : 24

INTERVALLE DE CONFIANCE DU TAUX D OCCUPATION MOYEN
 SEUIL ALPHA = 0.01 (0.525 , 0.662)
 SEUIL ALPHA = 0.05 (0.427 , 0.830)
 SEUIL ALPHA = 0.10 (0.443 , 0.614)

ESTIMATION DU TAUX D OCCUPATION MOYEN : 0.528

** PRECISION SUR LES TAUX D OCCUPATION **

| RESSOURCE | TAUX MOYEN OCCUP. | ECART SEUIL 1% | ECART SEUIL 5% | ECART SEUIL 10% |
|-----------|-------------------|----------------|----------------|-----------------|
| Z | 0.8343 | +0.0100 | -0.1062 | +0.0891 |
| P | 0.5284 | +0.0100 | -0.1019 | +0.0855 |

Fig 4.14 : Intervalle de confiance du taux d'acceptation moyen
des ressources z et p

c) Temps de réponse

Les résultats sur les temps de séjour simulés des transactions sont obtenus en comptabilisant le nombre de transactions entrées et sorties.

Lorsqu'une transaction entre dans une région, on note l'heure simulée. Ceci permet de calculer son temps de séjour simulé à sa sortie.

A la fin de la simulation, on peut :

- calculer un temps de séjour moyen s , connaissant les temps de séjour t_i , $i = 1, \dots, N$ des N transactions sorties pendant la simulation :

$$s = \frac{1}{N} \sum_{i=1}^N t_i$$

- obtenir un intervalle de confiance sur le temps de séjour moyen s lorsque le processus stochastique $\{T(t)\}$ est (quasi-) stationnaire en utilisant une méthode de blocs (III-3).

Exemple

Dans la maquette exemple, les résultats obtenus concernant les régions sont résumés dans un tableau (fig 4.15).

La figure 4.16 donne un intervalle de confiance sur les temps de réponse moyens à différents seuils α pour les régions physique et administratif.

Cette présentation des résultats peut être améliorée par l'édition d'histogrammes donnant la répartition des temps de réponse. Nous présentons un exemple relatif à la région administratif (fig 4.17).

| REGION (S) | ENTREES | SORTIES | CONTENU DE LA REGION | | | TEMPS DE SEJOUR | | |
|---------------|---------|---------|----------------------|--------|--------|-----------------|--------|-------|
| | | | MAXIMUM | MOYEN | ACTUEL | MAXIMUM | MOYEN | MINUM |
| ECRITES | 561 | 561 | 22 | 1.720 | 0 | 1 1 37 | 0 4 25 | 0 1 1 |
| TELEPHONEES | 905 | 905 | 1 | 0.040 | 0 | 0 0 43 | 0 0 4 | 0 1 1 |
| ADMINISTRATIF | 904 | 836 | 124 | 32.950 | 68 | 7 5 0 | 2 4 51 | 0 1 3 |
| PHYSIQUE | 1071 | 1009 | 62 | 13.750 | 62 | 1 21 45 | 0 19 0 | 0 1 1 |

Fig 4.15 : Récapitulatif sur les temps de réponse sur les régions

```
*****
* REGION NO 4 PHYSIQUE *
*****
```

NOMBRE DE MESURES FAITES PENDANT LA SIMULATION : 1009

NOMBRE DE BLOCS : 60
TAILLE D UN BLOC : 1 0 0

INTERVALLE DE CONFIANCE DU TEMPS DE SEJOUR MOYEN

SEUIL ALPHA = 0.01 (0 15 2 , 0 22 58)
SEUIL ALPHA = 0.05 (0 15 58 , 0 22 1)
SEUIL ALPHA = 0.10 (0 16 27 , 0 21 32)

ESTIMATION DU TEMPS DE SEJOUR MOYEN : 0 19 0

```
*****
* REGION NO 3 ADMINISTRATIF *
*****
```

NOMBRE DE MESURES FAITES PENDANT LA SIMULATION : 836

NOMBRE DE BLOCS : 60
TAILLE D UN BLOC : 1 0 0

INTERVALLE DE CONFIANCE DU TEMPS DE SEJOUR MOYEN

SEUIL ALPHA = 0.01 (1 15 1 , 2 18 42)
SEUIL ALPHA = 0.05 (1 18 18 , 2 15 25)
SEUIL ALPHA = 0.10 (1 20 0 , 2 13 43)

ESTIMATION DU TEMPS DE SEJOUR MOYEN : 2 4 51

** PRECISION SUR LES TEMPS DE SEJOUR MOYENS **

```
*****
R E G I O N | TPS MOYEN SEJOUR | ECART SEUIL 1% | ECART SEUIL 5% | ECART SEUIL 10% |
*****
PHYSIQUE | 0 19 0 | +DU- 0 3 57 | +DU- 0 3 1 | +DU- 0 2 32 |
ADMINISTRATIF | 2 4 51 | +DU- 0 13 50 | +DU- 0 10 33 | +DU- 0 8 51 |
*****
```

Fig 4.16 : Intervalle de confiance pour les temps de réponse moyens

Par contre, si on veut étudier le comportement stationnaire (lorsqu'il existe) en une seule simulation, il faut essayer de diminuer le biais introduit par les conditions initiales. Pour cela, on a deux solutions :

- Choisir un état initial qui soit un état possible d'état stationnaire

L'estimation de la moyenne d'un processus stochastique quasi-stationnaire dont on recueille une trajectoire à partir de l'instant t_0 est en fait une estimation de la moyenne conditionnelle. La variable aléatoire $X(t)$, $t > t_0$ dépend en général de $X(t_0)$ si la fonction d'auto-corrélation est non nulle. Ceci justifie le choix d'une réalisation possible de $X(t)$ à partir de la distribution stationnaire $\lim_{t \rightarrow \infty} X(t)$ pour $X(t_0)$.

- Eviter de prendre des mesures pendant un certain temps

On peut espérer que l'influence de la distribution initiale de $X(t_0)$ sur la distribution de $X(t)$ diminue à mesure que croît la durée $(t-t_0)$ (décroissance de la fonction d'autocorrélation d'un processus stochastique quasi-stationnaire). Mais il n'existe aucune réponse théorique permettant de déterminer combien de temps $X(t)$ est influencé par $X(t_0)$. Cela dépend généralement de la distribution initiale et du processus stochastique étudié.

Exemple

Dans le logiciel MAESTRO, l'utilisateur a le choix entre ces deux solutions qui peuvent se combiner ensemble :

- initialiser différentes valeurs des composants d'une maquette,
- lancer une première exécution pendant un temps déterminé et ne pas prendre en compte les mesures.

En conclusion, si on fait une seule simulation, elle doit être représentative des différentes simulations. Comme état initial, il faut donc choisir parmi les états les plus probables. En outre, distinguer deux parties : phase transitoire et phase stationnaire ne se justifie

pas théoriquement, à moins que l'état initial n'appartienne à une classe transiente d'états (ensemble d'états qui peuvent être quittés sans retour possible) (LER 77).

b) Conditions finales, durée de la simulation

La réponse à ces problèmes dépend de la précision statistique désirée sur les résultats.

Considérons le cas particulier d'une file d'attente (les clients non servis attendent dans une file). Si on veut obtenir conjointement des résultats sur la longueur moyenne de la file et sur le temps d'attente moyen, on est amené à constituer deux séries de mesures :

- contenu de la file mesuré à intervalles de temps réguliers

$t_1, t_2, \dots, t_n :$

x_1, x_2, \dots, x_n réalisation des variables aléatoires
 X_1, X_2, \dots, X_n

- temps d'attente des N clients servis :

y_1, y_2, \dots, y_N réalisation des variables aléatoires
 Y_1, Y_2, \dots, Y_N

Pour estimer $E[X(t)]$ et $E[Y(t)]$, nous utilisons les estimateurs classiques, respectivement \bar{X} et \bar{Y} définis par :

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \quad \text{et} \quad \bar{Y} = \frac{1}{N} \sum_{i=1}^N Y_i$$

Si on veut les estimer dans la même simulation, on est placé devant le problème suivant :

- si n est fixe, N est une variable aléatoire et \bar{Y} est biaisé,
- si N est fixe, n est une variable aléatoire et \bar{X} est biaisé.

Ne pouvant fixer à la fois n et N , un des deux estimateurs est biaisé.

Pour diminuer le biais introduit par la terminaison, on peut augmenter soit le nombre de mesures, soit la durée de la simulation.

Il n'existe pas de théorie permettant de déterminer la durée d'une simulation. Cette durée est liée à la précision statistique, qui peut être obtenue en calculant un intervalle de confiance selon deux approches possibles :

- fixer la taille de l'échantillon arbitrairement et utiliser une des méthodes suivantes :

- plusieurs répliques indépendantes du processus (LER 77, KOB 78),
- analyse spectrale (FIS 73),
- utilisation d'une représentation avec les modèles auto-régressifs (FIS 73),
- méthode des blocs (CON 63),
- cycles et points de régénération (CRA 74, CRA 75),

Lorsque le nombre de mesures est faible, les approximations effectuées et les lois limites utilisées ne sont pas toujours justifiées. Aussi, les résultats obtenus ne sont pas toujours satisfaisants.

Différentes études ont été réalisées à ce sujet et sont publiées dans les "Winter Simulation Conférences" de 1976, citées dans (LAW 79).

- arrêter la simulation lorsqu'une certaine précision est atteinte en utilisant des procédures séquentielles, très souvent construites à partir des méthodes précédentes (LAW 79). Cependant, ce type d'approche soulève des difficultés notamment du point de vue statistique pour estimer la variance du processus stochastique (KLE 74).

Exemple

Par le fait que dans le logiciel MAESTRO, nous fixons la durée de la simulation, les estimateurs du temps de séjour moyen s sont donc biaisés tandis que ceux du taux d'occupation moyen τ et du volume moyen de données sont non biaisés. (propriétés des estimateurs temporels, III-2.1).

Pour diminuer le biais introduit sur s , il faut donc choisir une durée de simulation suffisamment longue.

En conclusion, la précision d'une estimation obtenue en une seule simulation nécessite souvent une simulation relativement longue et coûteuse. La largeur de l'intervalle de confiance dépend de trois facteurs : le processus stochastique considéré, le nombre de mesures et l'estimateur choisi.

II-6.- Validation

Comme "le but d'une simulation est de prédire quelques aspects de la réalité" (NAYLOR), il nous faut vérifier la bonne représentativité d'un modèle, par rapport aux objectifs fixés en le validant.

Dans sa thèse (LER 77), J. Leroudier distingue trois types de validation :

- La validation structurelle : un modèle structurellement valide fournit des résultats satisfaisants et possède une structure de fonctionnement correspondant à celle de la réalité.

- La validation de répétitivité : un modèle validé permet de retrouver une réalité connue qui peut être un autre modèle déjà validé ou le système réel.

- La validation prédictive : un tel modèle peut être utilisé hors du domaine de validation pour faire de la prédiction. On dit aussi qu'il est robuste (voir 2.7) : il est capable de fournir des résultats satisfaisants dans d'autres situations.

Nous nous intéressons, dans cette partie, à la validation de répétitivité : un modèle ne peut être accepté que s'il reflète une réalité connue.

La validation d'un modèle de simulation s'effectue par comparaison avec une autre méthode d'évaluation. Nous en distinguons deux :

- mesures prélevées sur le système réel ou pendant l'exécution d'un autre modèle de simulation,
- résultats fournis par un modèle mathématique.

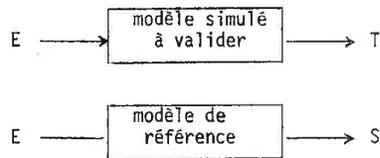
Nous examinons successivement ces deux types de méthodes. Elles sont largement utilisées dans le domaine de l'évaluation des performances des systèmes informatiques (FER 78). Dans notre champ d'intérêt sur

les systèmes d'information, nous n'avons pas connaissance de travaux à ce sujet. Aussi, contrairement à ce qui précède, nous faisons un exposé un peu théorique sur cette question et l'expérimentation de certaines méthodes présentées ici reste à faire.

II-6.1.- Validation par rapport à la réalité ou un autre modèle déjà validé (FIS 73, KLE 74)

Dans ce cas, pour chaque variable à mesurer, on dispose de deux séries de mesures : l'une provenant du modèle de référence (réalité ou autre modèle), l'autre prélevée au cours de la simulation.

Pour comparer ces deux séries, diverses méthodes statistiques existent. Généralement, on peut adopter le principe suivant :



E, T, S sont des processus stochastiques multi-dimensionnels qui modélisent respectivement les entrées des modèles, les sorties du modèle à valider et les sorties du modèle de référence.

Nous étudions alors le processus stochastique différence D :

$$D = T - S$$

Comme tout processus multi-dimensionnel, D est difficile à étudier. Nous considérons successivement ses N composantes D_j , ($j = 1, 2, \dots, N$) qui sont des processus stochastiques discrets et nous étudions les N suites de variables aléatoires :

$$\{D_j(t_i) \mid i = 1, 2, \dots, n_j\} \quad j = 1, 2, \dots, N$$

Pour faire cette étude, nous considérons 2 cas :

a) $D_j(t_i)$ est connu pour $i = 2, \dots, n_j$

α) Si le modèle est "correct", les deux processus stochastiques T et S doivent être identiques. Ceci nous permet de faire les hypothèses suivantes pour chaque composante D_j :

- Les variables aléatoires $D_j(t_i)$, $i = 1, \dots, n_j$ sont mutuellement indépendantes.
- $E[D_j(t_i)] = 0 \quad \forall i \in [1, n_j]$.

En effet, on peut penser que les erreurs $d_j(t_i)$ $i = 1, \dots, n$ réalisations des variables aléatoires $D_j(t_i)$ ne sont pas systématiques, ce qui justifie l'hypothèse d'indépendance.

Pour tester l'hypothèse nulle d'indépendance, on peut utiliser les différents tests vus précédemment (tests paramétriques ou non paramétriques, voir III-4.5).

β) Une autre solution consiste à faire une comparaison d'échantillons en testant l'hypothèse nulle qu'ils ont la même population mère. Cette approche est préférable, elle prend en compte les distributions statistiques. Mais elle ne peut être mise facilement en oeuvre que si les processus T_j et S_j sont purement aléatoires.

γ) Une autre approche consiste à exprimer sous forme de régression multi-linéaire les différences constatées $d_j(t_i)$, $i = 1, \dots, n_j$. Cette différence $d_j(t)$, variable à expliquer, s'exprime en fonction de p variables explicatives $x_k(t)$, ($k = 1, 2, \dots, p$) (supposées indépendantes) qui sont p fonctions du modèle qu'on peut éventuellement modifier pour le valider.

$$d_j(t) = \sum_{k=1}^p a_k x_k(t)$$

où les coefficients a_k sont calculés pour minimiser une norme au sens des moindres carrés (LIN 63): minimiser la somme des carrés des erreurs.

$$\sum_{i=1}^{n_j} \left[d_j(t_i) - \sum_{k=1}^p a_k x_k(t_i) \right]^2 \text{ minimal.}$$

Si les coefficients a_k sont proches de 0, on peut admettre que les fonctions explicatives $x_k(t)$ du modèle sont correctes et considérer le modèle comme validé. Sinon, il faut modifier la partie du modèle concernant la fonction $x_k(t)$ pour minimiser a_k .

On peut remarquer qu'il est possible de choisir des modèles de régression plus complexes, permettant notamment de s'affranchir de l'hypothèse d'indépendance des variables $x_k(t)$.

Généralement, la forme du modèle de régression est déterminée soit par des idées a priori, soit par essais successifs.

Exemple

Pour valider la maquette prototype par rapport aux mesures réelles, nous pouvons considérer que l'erreur sur le temps de réponse dépend de certains paramètres comme le nombre de lignes de la commande, le nombre de lignes de commande en attente, le nombre de lignes ou bons de livraison à préparer, le nombre de bons de livraison de la commande.

b) $D_j(t_i)$ est inconnu pour $i = 1, 2, \dots, n_j$

On peut alors étudier séparément les deux processus stochastiques T_j et S_j en recherchant diverses caractéristiques (périodicité, tendance ... voir III-4.3.2.).

Lorsque T_j et S_j sont des processus (quasi-) stationnaires, on peut calculer un intervalle de confiance sur leur moyenne théorique. S'il y a recouvrement des intervalles, on peut considérer le modèle comme validé.

Lorsque S_j et T_j sont non stationnaires, on ne peut calculer d'intervalle de confiance sur leur moyenne théorique. On peut alors rechercher des modèles les approchant au mieux (voir III-4.3.3). Si ces derniers sont "proches", on peut considérer le modèle comme validé.

Comme nous pouvons le constater, ces méthodes sont bien adaptées à une validation de répétitivité. Si on considère l'approche analyse de régression, on peut considérer que le modèle fonctionne à peu près de la même façon que le modèle témoin (FER 78). Bien que plus complexe,

cette méthode est meilleure, elle permet d'amorcer un début de validation structurelle qui peut aussi s'effectuer par une analyse de variance, comme le propose LEROUDIER, dans (LER 77). Ceci suppose un plan d'expériences pour étudier l'influence des différents facteurs explicatifs qui peuvent être qualitatifs (par exemple la politique de service) ou quantitatifs (durée d'exécution par exemple).

II-6.2.- Validation par rapport à un modèle mathématique

Dans cette partie, nous examinons la validation d'un modèle simulé par rapport à un modèle mathématique. La comparaison des deux modèles se réduit à vérifier que leurs paramètres respectifs coïncident et à comparer les résultats obtenus qui sont essentiellement des moyennes (taux d'occupation, débit, temps de réponse ...).

Si on veut résoudre le modèle mathématiquement, nous devons utiliser essentiellement la théorie des files d'attente. Celle-ci s'est rapidement développée pour répondre aux problèmes posés par la modélisation des systèmes informatiques.

Nous présentons quelques résultats et méthodes utilisables en évaluation de performances de systèmes informatiques puis nous tenterons de voir comment les appliquer dans notre domaine.

Parmi les méthodes mathématiques, nous distinguons deux familles.

a) Méthodes de résolutions mathématiques exactes

- Méthode numérique d'analyse d'un système markovien (KLE 76, STE 76)

Cette méthode est applicable à tout modèle pouvant se réduire à un modèle markovien du premier ordre à nombre fini d'états (KLE 76, STE 76). La solution stationnaire d'un tel modèle est obtenue à partir de l'information contenue dans les valeurs propres et les vecteurs propres de la matrice de transition du processus de Markov associé au modèle.

Cette approche très générale permet de prendre en compte des modèles où interviennent des dépendances complexes entre les probabilités de transition entre stations ou les taux de services et l'état du réseau.

Bien que les calculs soient automatisables, elle devient difficile à appliquer lorsqu'augmente le nombre d'états de la chaîne (encombrement mémoire important, convergence d'algorithme lente).

- Méthode analytique (JAC 63, BAS 75)

Cette méthode concerne l'analyse mathématique des réseaux de files d'attente. Elle trouve son origine dans les travaux de JACKSON (JAC 63) qui portent sur des réseaux de files d'attente où :

- la loi de distribution des temps de service est exponentielle négative.
- les lois d'arrivées de l'extérieur sont des processus de Poisson.
- le mouvement des clients d'une station à l'autre est décrit par une chaîne de Markov d'ordre 1.

De plus, la discipline de service dans les stations est exclusivement une gestion FIFO. Les résultats obtenus ont été généralisés par BASKETT, CHANDY, MUNTZ et PALACIOS dans le théorème de BCMP (BAS 75). Ils concernent les réseaux ouverts, fermés ou mixtes dans lesquels transitent des clients de classes différentes, représentant des comportements probabilistes distincts. Les stations du réseau peuvent être de 4 types :

- type 1 : la discipline de service est FIFO et la loi de distribution du temps de service est exponentielle négative de paramètre pouvant être fonction de la longueur de la file.
- type 2 : la station a un nombre infini de serveurs, et la loi de distribution du temps de service est une loi de Cox qui peut être différente pour chaque classe de clients.
- type 3 : la discipline de service est processeur partagé et la loi du temps de service distincte pour chaque classe de clients est une loi de Cox.
- type 4 : la discipline de service est LIFO avec préemption et reprise et la loi du temps de service est aussi une loi de Cox, éventuellement différente pour chaque classe de clients.

Pour les réseaux ouverts ou mixtes, le processus d'arrivée est poissonien.

Il est alors démontré que la solution stationnaire existe toujours dans le cas d'un réseau fermé. Sinon cette solution existe sous réserve que deux conditions simples soient vérifiées et peut être calculée.

Pour un exposé détaillé, on peut consulter par exemple (GEL 75 a) ou (KLE 76). Vu la forme de ces expressions, leur calcul peut être automatisé de façon efficace et simple (MER 78).

Bien que les résultats obtenus soient exacts, les hypothèses d'utilisation étant restrictives (arrivée poissonienne, distributions des temps de service par exemple), différentes techniques de résolution approchée ont été mises au point pour palier cette limitation.

b) Méthodes de résolution approchée

Ces méthodes sont moins restrictives sur leurs hypothèses. Nous distinguons trois types :

- Méthode itérative de résolution de réseaux à stations non exponentielles

Cette méthode s'applique à des réseaux fermés satisfaisant les hypothèses du théorème de Jackson (JAC 63) sauf celle sur les distributions des temps de service qui sont générales. Elle utilise la notion de station complémentaire : pour chaque station i , on peut définir une station complémentaire B_i qui résume les caractéristiques du réseau complémentaire à la station i :

- la loi de probabilité du nombre de clients présents dans la station i est la même dans le réseau de Jackson que dans le réseau à 2 stations.
- la loi de probabilité des temps de réponse (temps passé dans la station i) est la même dans les deux réseaux.

Ces résultats sont connus sous le nom de théorème de NORTON (CHA 75 a, CHA 75 b).

Par ailleurs, il est possible de définir des résultats d'"équivalence" un peu plus faibles en s'assurant qu'il y a simplement identité des flux et non identité des lois de probabilités. Ainsi, l'application

du théorème de CHANG et LAVENBERG (CHA 74) permet d'obtenir des relations simples entre les taux d'utilisation des différents serveurs.

Cette méthode est facile à appliquer lorsque le réseau est constitué d'une station "générale" et d'une station exponentielle. Dans le cas d'un réseau général, on procède par itérations successives en ajustant les paramètres des stations afin de satisfaire conjointement les équations de conservation des flux dans le réseau et la conservation du nombre de clients.

Elle a été validée par CHANDY (CHA 75 b) et améliorée par MARIE (MAR 76).

- Méthode de (quasi-)décomposabilité (COU 77)

Cette méthode a son origine dans les techniques d'agrégation utilisées en économie. L'idée est de former avec les variables d'états d'un système dynamique un nombre limité de groupes (agrégats) tels que :

- les interactions entre groupes peuvent être analysées indépendamment des interactions au sein des groupes,
- les interactions à l'intérieur des groupes peuvent s'analyser indépendamment des interactions entre groupes.

Lorsque ces propriétés sont vérifiées, le système est dit quasi-décomposable et SIMON et ANDO ont montré que l'on pouvait distinguer deux échelles de temps dans le fonctionnement du système de la manière suivante (SIM 61) :

- à court terme, un équilibre local est atteint à l'intérieur de chaque sous-système, indépendamment des autres sous-systèmes,
- à long terme, un équilibre global s'établit sans modifier les équilibres locaux.

Ces résultats ont été étendus et formalisés dans le cas de réseaux de files d'attente par COURTOIS (COU 77). Ils permettent d'appliquer les théorèmes de Jackson et BCMP à des réseaux de files d'attente où existent des dépendances suffisamment faibles entre les stations pour qu'on puisse les étudier séparément.

Cette méthode peut être considérée comme une hiérarchie à deux niveaux : l'un avec les variables du système original, l'autre avec les

agrégats. La technique utilisée consiste à remplacer la partie du réseau correspondant aux variables d'un agrégat par une file d'attente équivalente au sens suivant :

- les distributions de probabilité du nombre total de clients dans les deux systèmes sont identiques,
- les distributions de probabilité des variables d'états n'appartenant pas à l'agrégat sont identiques.

L'intérêt de cette méthode est de ramener l'étude de systèmes quasi-décomposables à celle de systèmes plus petits. Elle comporte deux étapes :

- calcul de l'état d'équilibre des sous-systèmes indépendants.
- étude d'un nouveau système dont les variables d'états sont les agrégats obtenus lors de la première étape.

La précision de cette méthode est liée à la qualité de l'approximation. On peut trouver des exemples d'application dans le cas de systèmes informatiques (GEK 76, JOM 78) entre autres.

- Méthode d'approximation par diffusion (PUJ 78)

Cette méthode utilise la théorie des chaînes de Markov à temps continu et à état continu pour étudier des chaînes de Markov à temps continu et à état discret.

L'approximation d'un modèle par un processus de diffusion se fait en trois étapes (KLE 76) :

- détermination des paramètres de la diffusion,
- traitement des conditions aux limites,
- choix de la discrétisation permettant de revenir du continu au discret.

Les hypothèses retenues à chaque étape affectent la précision des résultats obtenus.

Cette méthode fut développée initialement pour étudier les files d'attente G/G/1 (file à un serveur dont les lois d'arrivée et de service sont quelconques). Les processus d'entrée et de service sont remplacés par leurs moyennes et l'approximation par les processus de diffusion permet des variations de cette moyenne en introduisant les deux premiers moments des lois d'arrivée et de service.

Ainsi, un processus de saut à état discret $N(t)$, (nombre de clients dans une file) est approximé par un processus de Markov continu $X(t)$ dont les variations $\Delta X(t)$ sont distribuées normalement : $\Delta X(t)$ représente la différence de deux processus (nombre d'entrées et nombre de sorties depuis l'instant origine) et tend vers une loi normale puisque les deux processus supposés indépendants tendent vers une loi normale.

La solution approchée d'un tel modèle est obtenue en résolvant un système d'équations aux dérivées partielles selon des techniques d'analyse numérique en tenant compte des conditions aux limites.

Cette méthode a été généralisée par KOBAYASHI et peut s'appliquer à un réseau général de files d'attente. Elle permet d'obtenir des résultats en état stationnaire (KOB 74 a) et pour des cas particuliers en état transitoire (KOB 74 b). Les problèmes posés par la précision de cette technique d'approximation ont été examinés (REI 74). Il en ressort que :

- la précision est maximale lorsque le temps de service suit une loi exponentielle,
- les erreurs sont faibles lorsque le taux d'utilisation est voisin de 1.

Cette méthode a été utilisée et développée pour étudier les systèmes informatiques (GEL 75 b), (GEL 77) par exemple.

Pour une présentation détaillée et relativement complète, on peut consulter (PUJ 78).

- Méthode d'isolation (PUJ 78)

Cette méthode proposée par J. LABETOULLE et G. PUJOLLE consiste à subdiviser le système global en L sous-systèmes à étudier séparément en utilisant différentes techniques (résolution exacte ou approchée). L'idée générale en est la suivante : comme il est possible de définir l'état d'un système de files d'attente par un certain nombre de paramètres qui représentent :

- l'état global du système,
- l'état des différents sous-systèmes,

• les interfaces entre chaque sous-système et le reste du système, il suffit de modéliser les interfaces en y faisant apparaître les paramètres les plus influents. Ceci permet de connaître pour chaque sous-système. :

- les caractéristiques du ou des flux d'entrée (interface entrée),
- les caractéristiques des temps de service (interface sortie).

La solution générale du modèle est alors obtenue en résolvant le système d'équations modélisant les interfaces. On est souvent amené à utiliser des méthodes d'analyse numérique, les équations ne formant généralement pas un système linéaire.

Après avoir tenté de présenter diverses résolutions mathématiques (exactes ou approchées), nous essayons de voir dans quelles conditions elles semblent bien adaptées à notre domaine.

c) Application à la validation de maquettes

D'une manière générale une file d'attente (ou station) comporte un ou plusieurs serveurs demandés par les clients rangés dans une file. Habituellement, on associe dans un modèle clients et serveurs respectivement à processus et ressource.

Dans une maquette, les processus pouvant être cycliques, le service d'un client correspond alors à une "partie" d'exécution cyclique de traitement de données. Les ressources physiques apparaissent naturellement comme des serveurs. (En réalité le problème est plus complexe, les véritables clients sont plutôt les données sur lesquelles s'exécutent les programmes séquentiels de processus).

Pour nous ramener à la théorie des réseaux de files d'attente, nous faisons les associations suivantes entre composants de maquette et files d'attente : ressources physiques et serveurs, structures de données et files, exécution d'un cycle du programme séquentiel du processus et temps de service.

Dans l'application de cette théorie surgissent certaines difficultés dues soit aux transformations subies par les données, soit aux interruptions et blocages. Nous allons les examiner et tenter de les résoudre.

- Lois d'arrivée ou de service

Les résultats connus concernent généralement des réseaux pour lesquels les distributions de probabilité des inter-arrivées ou des durées de service sont des lois exponentielles négatives ou des composées (loi de Erlang, loi hyperexponentielle ou loi de Cox).

L'utilisation des résultats théoriques nous amène à choisir entre deux solutions, soit utiliser les propriétés de robustesse du modèle par rapport aux lois d'arrivée ou de service, soit ajuster au mieux les distributions par des lois de Cox.

- Probabilités de transition

Dans les réseaux théoriques (Jackson, BCMP), ces probabilités sont des constantes.

Pour satisfaire cette condition, il faut fixer ces probabilités de transition. Lorsque les relations entre stations sont faibles, on peut généralement les évaluer sans trop de difficultés et ajuster ces probabilités par des constantes.

Outre ces difficultés, il nous faut aussi considérer les problèmes posés par l'utilisation des ressources.

- Utilisation simultanée de plusieurs ressources par un processus élémentaire

Pour des raisons techniques (risques de blocage dus aux phénomènes de calendriers), un processus ne peut requérir qu'une seule ressource à nombre fini d'accès. Aussi, lorsqu'un processus requiert deux ressources pour s'exécuter, l'une est nécessairement non bloquante, (ce peut être l'ordinateur, modélisé par une ressource à nombre infini d'accès par exemple).

La résolution de ce problème peut s'effectuer de deux manières :

- soit par la négligence pure et simple des durées d'utilisation de l'ordinateur en mode conversationnel et l'éventuelle modification des distributions des durées de service pour tenir compte de cet "oubli".

- soit utiliser le modèle de file d'attente simple de SKINNER (SKI 67) où le serveur n'est pas toujours disponible. Dans ce modèle, après avoir servi le ième client de la file pendant une durée V_i , le serveur devient oisif et pendant cette période Z_i , il est indisponible pour tous les clients éventuellement en attente. Lorsque la file est vide, il se met en inactivité pendant une durée aléatoire Q qui n'est pas interrompue par l'arrivée d'un client. Moyennant quelques hypothèses sur les lois d'arrivée (poissonniennes), les variables aléatoires V_i , Z_i et Q , on peut obtenir des résultats exacts. Ainsi, dans le cas d'une requête effectuée en conversationnel, il est possible de modéliser le délai de réponse ordinateur par la phase d'oisiveté.

- Utilisation de ressources préemptibles avec priorité

A notre connaissance, il n'existe aucun résultat théorique exact sur de tels réseaux, seul est traitée la file d'attente simple (KLE 76). Deux solutions sont alors envisageables :

- soit éliminer ces phénomènes de préemption et priorité, et utiliser les résultats généraux existants.
- soit se ramener à une file d'attente simple et utiliser les résultats concernant ce type de file.

- Existence de ressources avec calendriers

Le cas de réseaux de files d'attente où les ressources ne sont pas toujours disponibles n'a pas été traité à notre connaissance.

Pour pallier cette difficulté, nous avons deux possibilités :

- soit éliminer les calendriers en considérant que les ressources sont toujours disponibles. Dans ce cas, nous ne tenons pas compte des éventuelles interruptions de processus lorsqu'une ressource cesse d'être disponible.
- soit nous ramener à une file d'attente simple et utiliser le modèle de SKINNER où Q représente les périodes d'indisponibilité.

Parmi les difficultés rencontrées, les plus importantes concernent les contraintes de synchronisation :

- Contraintes de synchronisation

La condition de déclenchement des processus n'est pas seulement liée à la présence des données. Elle nécessite parfois l'arrivée d'un signal particulier (déclenchement de processus à des dates précises : arrivée du courrier par exemple ou lorsqu'un autre processus est terminé).

En outre, un processus non terminé peut être suspendu d'une part lorsqu'arrive une fin de période de disponibilité relative à une ressource utilisée, d'autre part lorsqu'un processus plus prioritaire requiert une ressource bloquante qu'il possède.

Comme nous pouvons le constater, même si les méthodes mathématiques sont faciles à utiliser (existence de packages automatisant ces méthodes tels que QNAP (MEP 78)), l'application des résultats sur les réseaux de files d'attente ne semble pas très bien adaptée à notre domaine d'étude. Avant de conclure, il est souhaitable de réaliser tout un ensemble d'expériences mettant en oeuvre différentes hypothèses simplificatrices (suppression des contraintes de synchronisation, des calendriers ...) pour apprécier la robustesse des maquettes.

Quoiqu'il en soit, la validation d'une maquette peut être menée en choisissant parmi les deux solutions suivantes :

- considérer le problème à un niveau de détail moins fin, pour s'affranchir de ces contraintes.
- simplifier pour supprimer des causes de blocage et utiliser des méthodes de résolution approchée (quasi-décomposabilité, diffusion) pour appréhender le réseau obtenu à partir de la maquette.

Ce sont des problèmes nouveaux qui se posent et le dépassement des difficultés rencontrées passe par le développement d'études expérimentales pour dégager les facteurs essentiels et par la réalisation d'outils d'analyse plus appropriés.

II-7.- Robustesse (POT 77)

Lorsqu'un modèle est construit et validé, il peut être utilisé pour prédire un comportement dans différentes situations à condition qu'il soit robuste. Cette notion de robustesse et son étude sont récentes. C'est vers 1965 que les propriétés de robustesse des réseaux de files d'attente ont commencé à être étudiées, lors des premières expériences de modélisation des systèmes informatiques.

Les propriétés de robustesse d'un modèle se définissent par rapport aux différentes hypothèses faites dans la modélisation (niveau de détail, loi des entrées ...) et s'apprécient par rapport aux variables de sorties.

Un modèle est dit robuste par rapport à une hypothèse et un type de résultats si le fait de ne pas satisfaire cette hypothèse ne modifie pas ces résultats de manière significative.

Ainsi, un modèle peut être considéré comme robuste dans une hypothèse donnée, pour un certain type de mesures et beaucoup moins pour un autre type. C'est la connaissance de ces différences de comportement qui permet d'utiliser un modèle dans de bonnes conditions.

Peu d'études théoriques existent dans ce domaine. Seuls sont connus quelques résultats concernant les files d'attente. Ils sont bien exposés dans la thèse de D. POTIER (POT 77).

Le mode de représentation d'un système par un réseau de files d'attente implique des hypothèses liées soit à la structure même du type de modèle choisi (architecture en réseau de files d'attente), soit aux contraintes imposées pour l'obtention d'une solution explicite (voir 2-6). D. POTIER examine les propriétés de robustesse par rapport aux hypothèses faites sur les points suivants :

- nature de la source (source infinie),
- distribution des temps de service (loi exponentielle),
- capacité des files d'attente (capacité infinie),

en considérant deux types de mesures :

- temps de réponse moyen,
- taux d'utilisation du serveur.

En l'absence de résultats théoriques, la seule méthode consiste pour chaque cas à comparer les résultats obtenus à partir du modèle "exact" dans lequel l'hypothèse simplificatrice n'est pas faite avec ceux fournis par le modèle approché correspondant.

D'où la nécessité de construire un plan d'expériences et évaluer les conséquences (KLE 74).

Exemple

Dans notre modèle, il semble intéressant d'étudier la robustesse notamment par rapport aux points suivants :

- distribution des temps de service (en choisissant des lois de Cox),
- entrées (loi exponentielle pour les inter-arrivées),
- fin de service lorsqu'une ressource cesse d'être disponible,
- lancement de certains processus à des instants précis,

en considérant deux types de mesures :

- temps de réponse,
- taux d'utilisation des ressources.

II-8.- Utilisation d'une maquette en prédiction

Sous réserve de robustesse, après avoir été validée, une maquette peut être exploitée à des fins de prédiction en considérant différentes expériences où interviennent des modifications sur :

- l'architecture de la maquette et son fonctionnement,
- les paramètres de son environnement.

Ainsi, il peut être intéressant d'utiliser une maquette pour essayer de voir comment réagit le système face à des situations inhabituelles en modifiant certains paramètres du fonctionnement ou de l'environnement.

Exemple

Dans la maquette prototype, on peut effectuer différentes modifications et voir leur influence sur le comportement du système, par exemple :

- modification des périodes de disponibilité des ressources (calendriers),
- modifier le nombre d'accès d'une ressource en modifiant le calendrier associé : doubler le nombre d'accès et diminuer de moitié la durée de disponibilité par exemple,
- affecter différemment les ressources,
- modifier le flux d'arrivée des commandes écrites en faisant arriver d'importantes commandes,
- modifier les arrivées des appels téléphoniques en tenant compte des tranches horaires, des jours de la semaine ...
-

II-9.- Conclusion

Dans ce chapitre, nous avons présenté une méthodologie pour évaluer quantitativement le fonctionnement dynamique d'une maquette en essayant d'avoir une vue synthétique sur différents problèmes à résoudre :

- le paramétrage du modèle,
- l'analyse des entrées et des sorties,
- la précision des résultats,
- la validation et la robustesse du modèle.

Dans cette progression, nous avons voulu présenter diverses solutions possibles avant de choisir une solution utilisable et pouvant être mise en oeuvre.



CHAPITRE V

OUTILS DE MESURE ET D'ANALYSE
DES ENTRÉES

Dans ce chapitre, nous présentons les outils d'analyse statistique concernant l'étude de séries chronologiques et la recherche de modèles probabilistes adéquats. Ceux-ci mettent en oeuvre des méthodes exposées dans le chapitre III. Ils nous ont servi lors du paramétrage de la maquette exemple (annexe) qui a été réalisée à partir de mesures réelles recueillies dans l'entreprise étudiée.

I - SPECIFICATIONS FONCTIONNELLES

I-1.- Objectifs

Ayant eu à analyser diverses séries chronologiques (paramétrage d'une maquette (IV-2.2.2), analyse de résultats (IV-2.5), nous avons conçu un outil statistique mettant en oeuvre des méthodes exposées dans le chapitre III pour déterminer des modèles probabilistes sur les processus stochastiques sous-jacents.

I-2.- Démarche

L'analyse d'une série chronologique (suite de mesures ordonnées dans le temps) se déroule en trois étapes (III-4.3) que nous rappelons :

- visualisation pour apprécier qualitativement des caractéristiques comme une tendance, des composantes périodiques ou une dépendance temporelle ;
- vérification de caractéristiques du processus stochastique sous-jacent : stationnarité (absence de composantes déterministes) et corrélation sériale ;
- identification du processus stochastique sous-jacent en considérant deux catégories de modèles suivant l'existence ou non de corrélation sériale.

I-3.- Caractéristiques générales

L'outil réalisé comporte trois parties reprenant ces trois étapes. Il est relativement standard et permet d'analyser des séries enregistrées dans un fichier de même structure que le fichier FVOL (voir VI-2.1.2) pour des raisons de simplicité d'utilisation.

Pour obtenir des renseignements sur la série chronologique, l'utilisateur peut recourir aux programmes présentés maintenant.

II - SPECIFICATIONS TECHNIQUES

Dans cette partie, nous considérons les caractéristiques générales de cet outil. Comme la structure et le fonctionnement des programmes réalisés sont transparents à l'utilisateur, nous n'en parlons pas et montrons simplement des exemples de résultats qu'ils peuvent fournir.

Nous utilisons le langage SIMULA (DAH 68, DAH 70).

Dans cette présentation, nous reprenons successivement les trois phases: visualisation, recherche de caractéristiques et identification (III-4). Les résultats présentés concernent deux séries :

- SERIE-A : évolution mensuelle de prix sur 36 mois (tirée de (AND 71)),
- NB-COM-ECR-J : Nombre de commandes écrites par jour sur 64 jours, sert à alimenter la maquette prototype (II-2.3).

II-1.- Visualisation

Le programme E1-SERIE permet d'obtenir divers graphes (III-4.4) :

- graphe des données,
- corrélogramme,
- périodogramme de Shuster,
- filtre de Buys-Ballot,
- spectrogramme.

Exemple

Nous présentons des résultats correspondant aux séries NB-COM-ECR-J et SERIE-A :

- graphe des données (fig 5.1),
- corrélogramme (fig 5.2),
- périodogramme de Schuster (fig 5.3),
- filtre de Buys-Ballot (fig 5.4),
- spectrogramme relatif à la série NB-COM-ECR-J (fig 5.5).

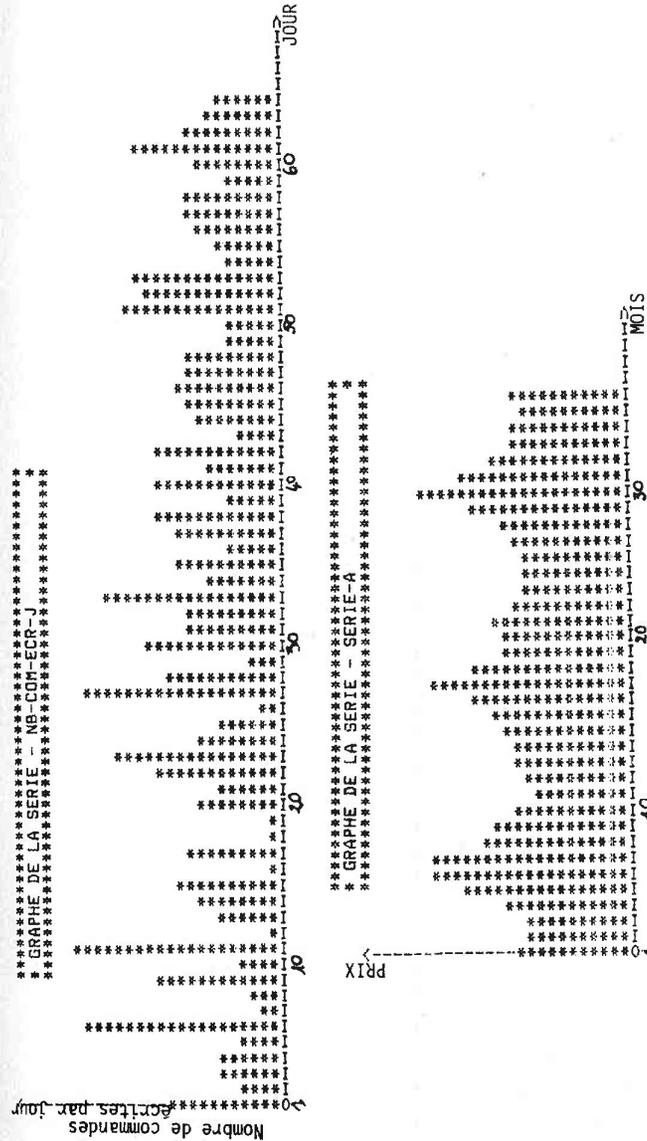
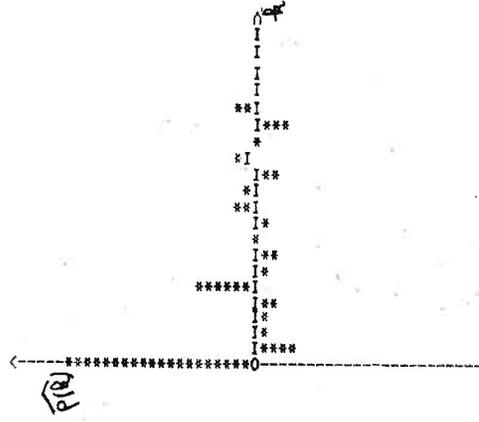


Fig 5.1 : Exemple de graphes des données
 Remarque : On ne remarque a priori ni de tendances ni de composantes cycliques sur la série NB-COM-ECR-J tandis qu'apparemment, SERIE-A présente une composante périodique.

```

*****
** CORRELOGRAMME DE LA SERIE - NB-COM-ECR-J **
*****

```



```

*****
** CORRELOGRAMME DE LA SERIE - SERIE-A **
*****

```

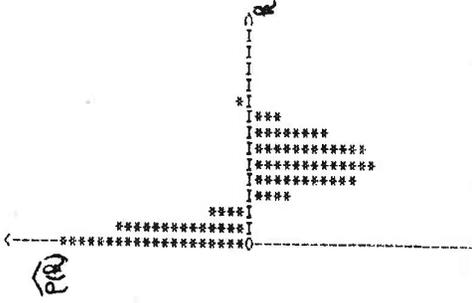


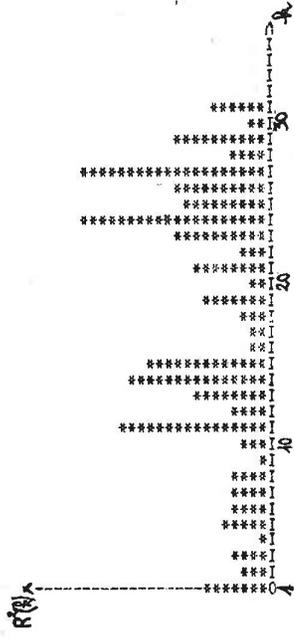
Fig 5.2 : Correlogrammes des séries NB-COM-ECR-J et SERIE-A

Remarque : En ce qui concerne NB-COM-ECR-J, la fonction d'autocorrélation décroît rapidement. On peut accepter à priori l'absence de cycles et admettre une faible dépendance temporelle entre deux observations successives. L'observation du correlogramme de SERIE-A montre l'existence d'une périodicité et de dépendance temporelle entre les mesures.

```

*****
** PERIODOGRAMME DE SHUSTER DE LA SERIE **
** NB-COM-ECR-J **
*****

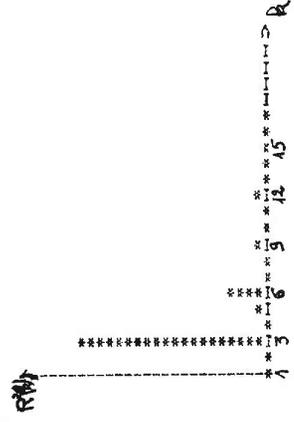
```



```

*****
** PERIODOGRAMME DE SHUSTER DE LA SERIE **
** SERIE-A **
*****

```

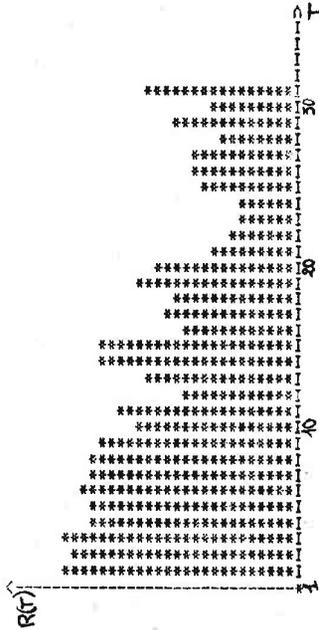


REMARQUE : HYPOTHESES DU MODELE - ABSENCE DE TENDANCE MOUVEMENT SAISONNIER DE LA FORME : $A \cos \omega t + B \sin \omega t$

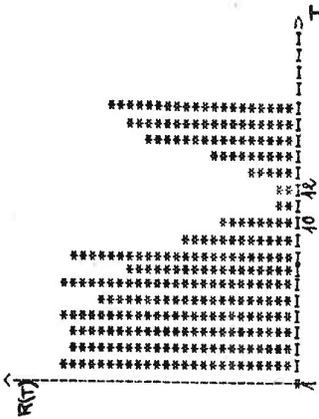
Fig 5.3 : Périodogrammes de Schuster

Remarque : Si on considère le périodogramme de la série NB-COM-ECR-J, on observe plusieurs pics. Comme leurs valeurs sont relativement faibles par rapport à la valeur moyenne, on peut à priori rejeter l'existence de périodicité. Par contre, l'observation du périodogramme de la série SERIE-A montre un pic très élevé pour la fréquence 3/36 et on peut fortement penser à une périodicité de $T = 12$ jours.

```
*****
* FILTRE DE BUYS-BALLOT *
* NB-COM-ECR-J *
*****
```



```
*****
* FILTRE DE BUYS-BALLOT *
* SERIE-A *
*****
```



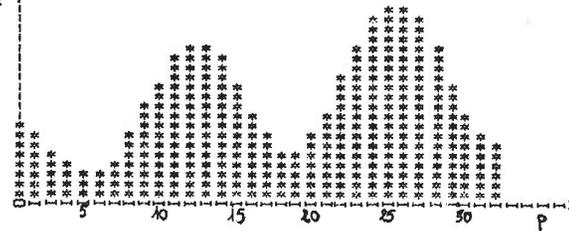
REMARQUE : HYPOTHESE DU MODELE - XI = SI + EPSI
 MOUVEMENT SAISONNIER
 SI EPSI ALEA

Fig 5.4 : Filtrés de Buys-Ballot

Remarque : L'examen du graphe des résidus en appliquant le filtre de Buys-Ballot nous amène aux constatations suivantes :

- SERIE-NB-COM-ECR-J : la valeur minimale de $R(T)$ correspond à $T = 23$, mais cette valeur étant relativement élevée par rapport à la valeur moyenne, on ne peut rien conclure.
- SERIE-A : la valeur minimale de $R(T)$ correspond à $T = 12$. Cette valeur étant très faible, par rapport aux autres résidus, on peut accepter l'existence de périodicité et le vérifier ultérieurement.

```
*****
* SPECTROGRAMME DE LA SERIE NB-COM-ECR-J *
*****
```



REMARQUE : CE GRAPHE N EST CORRECT QUE SI TOUS LES COEFFICIENTS
 DE CORRELATION SONT CALCULABLES A PARTIR D UNE
 FONCTION CONNUE -ESTIMATION PREALABLE -

Fig 5.5 : Spectrogramme de la série SERIE-A

Remarque : En observant ce graphe, on ne peut à priori rien déterminer : il y a plusieurs valeurs élevées et on ne peut rien conclure avant de poursuivre une analyse plus approfondie.

II-2.- Recherche de caractéristiques sur le processus stochastique sous-jacent

Le programme E2-SERIE permet d'obtenir des indications sur certaines propriétés du processus stochastique sous-jacent telles que la stationnarité, l'existence de cycles ou de dépendance temporelle (III-4.5).

Nous examinons successivement ces trois points et présentons les résultats obtenus sur la série SERIE-A (fig 5.6, fig 5.7, fig 5.8).

II-2.1.- Stationnarité

Considérant comme stationnaires des séries ne comportant pas de tendance, le programme E2-SERIE automatise les tests non paramétriques suivants qui détectent des tendances très générales : trend test, run test, test des rangs de Spearman, test du coefficient de rang de Kendall, tests de Cox et Stuart (WAL 62).

II-2.2.- Existence de composantes cycliques

Tout d'abord, l'utilisateur doit examiner les graphes fournis par le programme E1-SERIE. Il peut compléter cette étude par l'utilisation du programme E2-SERIE qui automatise les tests suivants : test du nombre de points critiques, test du χ^2 sur la longueur des phases (WAL 62), test du périodogramme cumulé (FIS 73), test par analyse de variance (AND 71).

II-2.3.- Dépendance temporelle

En premier lieu, l'utilisateur doit examiner le graphe des données et le corrélogramme. S'il suppose l'existence d'une dépendance temporelle d'ordre 1, il peut utiliser le programme E2-SERIE qui automatise les tests suivants : test de Von Neumann et test du coefficient de corrélation sériale en utilisant la z-transformation de Fisher (AND 71).

Remarque :

Chaque fois qu'on teste une hypothèse, aucun des tests proposés ne doit être déterminant à lui seul (certains tests sont mieux adaptés que d'autres). Ainsi, l'attitude à avoir est de n'accepter une hypothèse que si elle est simultanément satisfaite par plusieurs tests.

Parmi les séries à analyser (IV-2.3), aucune ne présentait de composantes déterministes ou de forte dépendance temporelle. Aussi, nous nous sommes contentés de mettre en oeuvre le plus souvent des tests statistiques non paramétriques faciles à utiliser mais pas toujours très puissants.

** RECHERCHE DE TENDANCE ** SERIE : SERIE-A

H0 : IL N EXISTE PAS DE TENDANCE
H1 : IL EXISTE UNE TENDANCE

RUN TEST

NOMBRE DE MESURES UTILISEES : 36 NOMBRE DE RUNS : 6

DECISION : CONSULTER LA TABLE DU RUN TEST

au seuil $\alpha = 10\%$ l'hypothèse H0 est rejetée car $6 \notin]13,24]$

TREND TEST

NOMBRE DE MESURES UTILISEES : 36 VALEUR DU COEFFICIENT A : 328

DECISION : CONSULTER LA TABLE DU TREND TEST

au seuil $\alpha = 10\%$ l'hypothèse H0 est acceptée car $328 \in]245,362]$

TEST DES RANGS DE SPEARMAN

VALEUR DU COEFFICIENT DE SPEARMAN : 0.228
NOMBRE DE MESURES UTILISEES : 36

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 1.350 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 1.350 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 1.350 - HYPOTHESE H0 ACCEPTEE

TEST DU NOMBRE DE DIFFERENCES POSITIVES

NOMBRE DE MESURES UTILISEES : 36
NOMBRE DE DIFFERENCES POSITIVES : 19

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 0.854 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 0.854 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 0.854 - HYPOTHESE H0 ACCEPTEE

Fig 5.6 : Exemple de tests de tendance sur la série
SERIE-A

TEST DU COEFFICIENT DE RANG DE KENDALL

NOMBRE DE MESURES UTILISEES : 36 VALEUR DU COEFFICIENT A : 26

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 0.368 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 0.368 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 0.368 - HYPOTHESE H0 ACCEPTEE

TEST DE COX-STUART POIDS OPTIMAUX -ECH.TOTAL-

NOMBRE DE MESURES UTILISEES : 36 VALEUR DU COEFFICIENT A : 164

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 0.045 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 0.045 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 0.045 - HYPOTHESE H0 ACCEPTEE

TEST DE COX-STUART POIDS EGaux -ECH.TOTAL-

NOMBRE DE MESURES UTILISEES : 36 VALEUR DU COEFFICIENT A : 9

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 0.000 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 0.000 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 0.000 - HYPOTHESE H0 ACCEPTEE

TEST DE COX-STUART POIDS EGaux -ECH.INCOMPLET-

NOMBRE DE MESURES UTILISEES : 36 VALEUR DU COEFFICIENT A : 7

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 0.577 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 0.577 - HYPOTHESE H0 ACCEPTEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 0.577 - HYPOTHESE H0 ACCEPTEE

Fig. 5.6 : Exemple de tests de tendance sur la série
SERIE-A (suite)

L'hypothèse H0 étant acceptée par la presque totalité des tests, on peut raisonnablement conclure à l'absence de tendance dans SERIE-A.

H0 : IL N EXISTE PAS DE PERIODICITE
H1 : IL EXISTE UNE PERIODICITE

TEST DES LONGUEURS DE PHASES

VALEUR DU QUI2 CALCULE : 81.378 NOMBRE DE DEGRES DE LIBERTES : 2.50

SEUIL ALPHA : 0.010 QUI2TH : 0.067 - HYPOTHESE H0 REJETEE
SEUIL ALPHA : 0.050 QUI2TH : 0.227 - HYPOTHESE H0 REJETEE
SEUIL ALPHA : 0.100 QUI2TH : 0.399 - HYPOTHESE H0 REJETEE

TEST D ANALYSE DE VARIANCE

VALEUR DU RAPPORT : 24.288

DECISION : CONSULTER LA TABLE DE LA LOI F(11, 24)

au seuil $\alpha = 5\%$ $F_{\alpha}(11,24) = 2.22$

décision : l'hypothèse H0 est rejetée au seuil $\alpha = 5\%$

TEST DU NOMBRE DE POINTS CRITIQUES

NOMBRE DE MESURES UTILISEES : 36
NOMBRE DE POINTS CRITIQUES : 11

DECISION : CONSULTER LA TABLE DU NOMBRE DE POINTS CRITIQUES
au seuil $\alpha = 5\%$, l'hypothèse H0 est rejetée

TEST DU PERIODOGRAMME CUMULE

VALEUR DE C : 0.9412
PERIODE ASSOCIEE : 12

DECISION : CONSULTER LA TABLE CORRESPONDANTE

au seuil $\alpha = 5\%$ $C_{\alpha} = 0.256$

décision : l'hypothèse H0 est rejetée au seuil $\alpha = 5\%$

Fig 5.7 : Exemple de tests de recherche de périodicité sur la série SERIE-A

L'hypothèse H0 étant rejetée par ces tests, nous pouvons conclure à l'existence d'une périodicité dans SERIE-A.

H0 : IL N EXISTE PAS DE DEPENDANCE
H1 : IL EXISTE UNE DEPENDANCE

TEST DU RAPPORT DE VON NEUMANN

NOMBRE DE MESURES UTILISEES : 36 VALEUR DU RAPPORT : 0.612

décision : ce rapport est très éloigné de 2, l'hypothèse H0 est rejetée au seuil $\alpha = 5\%$ en consultant la table correspondante.

TEST DU COEFFICIENT DE CORRELATION SERIALE D ORDRE 1

VALEUR DE R1 : 0.7044

UTILISATION DE LA Z-TRANSFORMATION DE FISHER
VALEUR DE Z : 0.9335

DECISION : UTILISATION DE LA LOI LIMITE NORMALE

SEUIL ALPHA : 0.010 TALPHA : 2.570 T : 5.601 - HYPOTHESE H0 REJETEE
SEUIL ALPHA : 0.050 TALPHA : 1.960 T : 5.601 - HYPOTHESE H0 REJETEE
SEUIL ALPHA : 0.100 TALPHA : 1.645 T : 5.601 - HYPOTHESE H0 REJETEE

Fig 5.8 : Exemple de tests de recherche de dépendance sur la série SERIE-A

A partir de ces résultats, on peut accepter l'existence d'une dépendance temporelle entre deux observations successives de SERIE-A.

II-3.- Recherche de modèles probabilistes

Dans la recherche de modèles probabilistes, nous nous sommes limités pour diverses raisons (entre autres techniques) à la détermination de modèles faisant intervenir des variables aléatoires mutuellement indépendantes (III-4.6.1).

La procédure d'identification retenue (III-4.6.1) comporte trois étapes :

- choix d'un modèle,
- estimation des paramètres,
- vérification des hypothèses faites.

Nous considérons successivement ces trois points.

II-3.1.- Choix d'un modèle

Le programme DESCRIPT permet d'obtenir des caractéristiques sur la distribution statistique en donnant diverses estimations sur la tendance centrale, la dispersion et la forme de la distribution empirique (caractéristiques résumées dans le tableau de la figure 3.8 dans le chapitre III).

Exemple

Nous résumons les caractéristiques des modèles probabilistes concernant :

- le nombre de commandes écrites par jour (série NB-COM-ECR-J),
- le nombre de lignes par commande écrite (série LG-COM-ECR),
- le nombre de lignes par commande téléphonique (série LG-COM-TEL).

Les histogrammes correspondants figurent dans le chapitre IV (fig 4.3, 4.4, 4.5).

| caractéristique \ série | NB-COM-ECR-J | LG-COM-ECR | LG-COM-TEL |
|-------------------------|--------------|------------|------------|
| moyenne | 8.46 | 5.25 | 3.14 |
| médiane | 9 | 4 | 3 |
| mode | 9 | 1 | 1 |
| écart-type | 7.98 | 32.62 | 21 |
| écart moyen abs/moy. | 3.76 | 3.99 | 2.49 |
| coef. de variation | 0.89 | 38.62 | 45.20 |
| coef. de dissymétrie | -0.6 | -16.43 | -15 |
| coef. d'aplatissement | 4.2 | 327.75 | 260.38 |

Fig 5.9 : Caractéristiques descriptives

L'observation des histogrammes correspondant à ces trois séries nous amène à rechercher des modèles probabilistes du type combinaison linéaire de lois connues : lois exponentielles pour les processus stochastiques sous-jacents aux séries LG-COM-TEL et LG-COM-ECR et lois normales en ce qui concerne la série NB-COM-ECR-J. Pour des raisons de complexité d'algorithme, nous n'avons pas approfondi cette recherche.

II-3.2.- Estimation des paramètres

Lorsqu'on a retenu un modèle probabiliste, il est possible d'estimer les paramètres du modèle soit à partir des estimations fournies par la statistique descriptive soit en les calculant soi-même ou par programme.

Dans le paramétrage de la maquette exemple, les modèles probabilistes n'étant pas simples, nous n'en dirons pas plus, n'ayant pas mis en oeuvre l'algorithme proposé par A. SCHROEDER (III-4.6.1) (vu sa complexité) pour rechercher les composants d'un mélange hétérogène.

II-3.3.- Vérification

Le programme E3-SERIE a pour but de vérifier l'adéquation du modèle retenu. Il met en oeuvre les tests présentés en (III-4.6.1.c) test de Kolmogoroff, test de Von Mises et test du χ^2 .

Si les modèles théoriques retenus sont des distributions exponentielles, poissonniennes ou normales, la fonction de répartition théorique est tabulée automatiquement. Dans le cas contraire, l'utilisateur doit fournir lui-même la tabulation.

Exemple

Nous montrons des résultats fournis par ce programme à partir de deux échantillons utilisés pour le tester.

- échantillon 1 : échantillon de 10 mesures (ordonnées)

| | | | | |
|-------|-------|-------|-------|-------|
| 0.203 | 0.329 | 0.382 | 0.477 | 0.480 |
| 0.503 | 0.554 | 0.581 | 0.621 | 0.710 |

On fait l'hypothèse que la loi théorique est une loi uniforme sur l'intervalle $[0,1]$.

L'application des tests de Kolmogoroff et de Von Mises sur cet échantillon nous conduit à accepter cette hypothèse (fig 5.10).

TEST DE VON MISES *****

HYPOTHESES : XO V.A.R CONTINUE
ECHANTILLON ORDONNE

DECISION :

| | | | |
|---------------------|----------------|-----------|-------------------------|
| SEUIL ALPHA : 0.100 | TALPHA : 0.347 | T : 0.240 | - HYPOTHESE HO ACCEPTEE |
| SEUIL ALPHA : 0.050 | TALPHA : 0.461 | T : 0.248 | - HYPOTHESE HO ACCEPTEE |
| SEUIL ALPHA : 0.010 | TALPHA : 0.743 | T : 0.248 | - HYPOTHESE HO ACCEPTEE |

TEST DE KOLMOGOROFF SMIRNOV *****

HYPOTHESE : XO V.A.R. CONTINUE

DECISION :

| | | | |
|---------------------|----------------|-----------|-------------------------|
| SEUIL ALPHA : 0.100 | TALPHA : 0.380 | T : 0.290 | - HYPOTHESE HO ACCEPTEE |
| SEUIL ALPHA : 0.050 | TALPHA : 0.430 | T : 0.290 | - HYPOTHESE HO ACCEPTEE |
| SEUIL ALPHA : 0.010 | TALPHA : 0.515 | T : 0.290 | - HYPOTHESE HO ACCEPTEE |

Fig 5.10 : Application des tests de Kolmogoroff et de Von Mises

- échantillon 2 : échantillon de 576 mesures dont l'histogramme est donné sur la figure 5.11 et la répartition des mesures ci-dessous :

| valeur | 0 | 1 | 2 | 3 | 4 | 5 |
|------------------------|-----|-----|----|----|---|---|
| Nombre de réalisations | 229 | 211 | 93 | 35 | 7 | 1 |

CHAPITRE VI

OUTILS DE MESURE ET D'ANALYSE
DES SORTIES

Dans ce chapitre, nous présentons les outils de mesure et d'analyse des sorties réalisés dans le cadre du projet MAESTRO. Ils mettent en oeuvre des méthodes exposées dans les chapitres III et IV et permettent d'obtenir des résultats comme ceux présentés au chapitre IV.

Les outils de mesure réalisés sont intégrés dans le logiciel MAESTRO (voir DUF 80a) pour ce qui concerne l'analyse des sorties tandis que les programmes d'analyse statistique sont disjoints.

Nous nous limitons à une présentation générale pour ne pas encombrer le texte par des détails de programmation.

I - SPECIFICATIONS FONCTIONNELLES

I-1.- Objectifs

Après avoir construit une maquette, on peut évaluer les performances du système décrit en la simulant (IV-1). Ceci permet d'obtenir des résultats sur (IV.2.5.1) :

- les volumes de données traitées,
- l'utilisation de certaines ressources,
- les temps de réponse relatifs à certaines parties du système.

Désirant les obtenir en une seule simulation, il est nécessaire que le système soit en état stationnaire pour que les résultats soient significatifs (IV.2.5).

I-2.- Démarche

Pour satisfaire ces objectifs d'évaluation (chapitre IV), nous avons conçu divers outils qui mettent en oeuvre les méthodes choisies et exposées auparavant dans le chapitre III pour les problèmes statistiques (III.2 et III.3) et dans le chapitre IV pour les problèmes purement informatiques (IV.2).

Les mesures prélevées au cours de la simulation sont utilisées directement dans des calculs (valeurs moyenne, maximale et minimale...) ou ultérieurement dans une analyse statistique plus approfondie (recherche de caractéristiques sur les séries de mesures, calcul d'inter-

valles de confiance).

I-3.- Caractéristiques générales

Pour avoir un mécanisme relativement standard et général, nous avons été amenés à définir différents types d'objets pour collecter des résultats : transactions, régions, structures de données et ressources (IV.2.5.1).

Dans cette partie, nous donnons leur description et leur utilisation.

I-3.1.- Description des objets utilisés

a) Les transactions

C'est une classe d'objets qui regroupe les entités mobiles du système : ce sont des objets de ce type qui traversent les régions ou sont contenus dans les structures de données.

b) Les régions

Elles servent à définir des parties du système pour lesquelles on désire connaître un temps de réponse moyen (IV.2.5.1.c).

Une région est définie par un point d'entrée et de sortie entre lesquels circulent des transactions.

Ce type d'objets permet de collecter des résultats sur :

- le nombre d'entrées et de sorties,
- le contenu (maximal, minimal, moyen),
- le temps de séjour (maximal, minimal, moyen).

Remarque : Pour se faire une idée sur la distribution des temps de réponse, l'utilisateur peut obtenir un histogramme.

c) Les structures de données

Ce type d'objets permet de représenter les structures de données du système modélisé (II.1.1.1.a) et de collecter des résultats sur

l'évolution des volumes de données dans le temps : volume moyen, maximal ou minimal (IV.2.5.2.a) et intervalle de confiance sur la moyenne du volume de données éventuellement.

Par ailleurs, il est possible d'obtenir des résultats plus spécifiques au type de structure de données considéré. Jusqu'à présent, nous n'avons mis en oeuvre ce procédé que pour les files d'attente, ce qui fournit des résultats sur :

- le nombre de transactions entrées et sorties,
- le nombre de transactions présentes (maximal, minimal, moyen),
- le nombre de fois où la file est vide.

d) Les ressources

Ce type d'objets a été introduit pour obtenir des résultats concernant l'utilisation des ressources décrites dans une maquette (II.1.1.2). Les mesures collectées permettent de calculer un taux d'utilisation de la ressource et d'autres résultats qui dépendent du type de ressources modélisées :

- ressource critique : • nombre de processus utilisateurs,
• durée de disponibilité de la ressource ;
- ressource à accès multiple : • nombre total d'accès disponibles,
• nombre d'accès utilisés (moyen, maximal, final),
• taux d'utilisation par accès ;
- ressource "processeur partagé" : • nombre total de processus utilisateurs,
• nombre de processus utilisateurs simultanément, (moyen, maximal, final) ;
- ressource à nombre infini d'accès : résultats identiques aux précédents.

Remarque : Pour chaque ressource, il nous semble intéressant de connaître la fréquence d'utilisation des accès. A cet effet, nous avons ajou-

té un diagramme d'utilisation des accès en nous limitant aux fréquences d'utilisation du nombre d'accès inférieur ou égal à 10.

I-3.2.- Description et utilisation des outils

Les outils de mesure que nous avons réalisés concernent :

- la prise des mesures durant la simulation,
- l'édition des résultats après la simulation,
- l'analyse statistique de séries chronologiques pour obtenir des précisions sur les résultats (taux d'occupation moyen, temps de réponse moyen, volume moyen de données) et sur l'état du système.

Dans l'utilisation de ces outils, nous distinguons trois étapes :

a) Pendant la simulation

Les différents types d'objets définis ci-dessus servent à collecter des résultats relatifs aux volumes de données, à l'utilisation des ressources et aux temps de réponse :

- régions : les mesures ont lieu lors des mouvements de transactions,
- structures de données : idem aux régions et éventuellement à intervalles réguliers,
- ressources : les mesures s'effectuent lorsqu'il y a acquisition ou restitution d'un ou plusieurs accès et à chaque début et fin de période de disponibilité si la ressource possède un calendrier.

Nous ne détaillons pas plus en renvoyant à (THA 79).

b) A la fin de la simulation

Les résultats généraux (présentés dans le paragraphe précédent) concernant les régions, les structures de données (et les files), les ressources sont enregistrés dans un fichier pour être édités ultérieurement.

c) Après la simulation

Les mesures effectuées pendant la simulation peuvent être exploitées à des fins diverses :

- caractérisation de l'état du système en analysant les séries chronologiques relatives aux mesures sur les volumes de données, l'utilisation des accès des ressources et les temps de réponse,
- édition des résultats généraux qui ne sont significatifs que si le système est en équilibre,
- calcul d'intervalle de confiance sur les volumes de données moyens, les taux d'occupation moyens et les temps de réponse moyens.

II - SPECIFICATIONS TECHNIQUES

Nous donnons les caractéristiques techniques générales, la forme syntaxique et l'utilisation des outils créés. Ils sont intégrés à la classe MAESTRO dont on peut trouver une version dans (DUF 80a).

II-1.- Caractéristiques techniques générales

II.1.1.- Langage

Nous utilisons SIMULA 67 (CII 72) comme langage d'écriture des maquettes (DUF80a). Ce dernier est intéressant notamment pour

- son puissant concept de classes hiérarchisées favorisant la structuration des programmes,
- la simulation à événements discrets.

Cependant comparé à d'autres langages de simulation comme SIMSCRIPT (MAR 66), GPSS (KIV 69), il est relativement pauvre en outils statistiques, ce qui nous a amenés à développer nos propres outils.

Pour une bonne compréhension de notre travail, nous faisons un bref rappel sur les concepts qui nous sont utiles : classes hiérarchisées et processus.

a) Classes hiérarchisées

La déclaration d'une classe en SIMULA crée un nouveau type paramétré d'objets contenant des attributs locaux et des (fonctions-) procédures d'accès et de modification. Nous n'en donnons pas une description exhaustive (CII 72) mais seulement une forme simplifiée :

```
class nomclasse (liste des paramètres formels);...;
begin
  ... déclaration des attributs locaux ... ;
  ... instructions ... ;
end ;
```

La création d'un objet de type nomclasse repéré par nc s'exprime ainsi :

```
ref (nomclasse) nc ;
nc : -new nomclasse (liste des paramètres effectifs) ;...;
```

Ceci permet de générer un nouvel objet de type nomclasse repéré par nc, d'effectuer le passage des paramètres et d'exécuter les instructions simples du corps de classe pour cet objet.

L'accès aux attributs locaux d'un objet d'une classe s'écrit en notation pointée. Ainsi, l'appel d'une (fonction -) procédure nomproc attribut de l'objet de type nomclasse, repéré par nc s'exprime par :

```
nc.nomproc (liste des paramètres effectifs) ;
```

A ce concept de classe s'ajoute celui de la hiérarchisation. Il est possible de préfixer une classe par le nom d'une autre classe : on dit qu'on déclare une sous-classe de la classe préfixée. Ceci permet la concaténation des définitions de deux classes : un objet d'une classe préfixée possède outre les attributs locaux à sa classe tous ceux de la classe dont le nom figure en préfixe (DAH 70).

b) Processus

SIMULA 67 possède une classe prédéfinie de processus, la classe process qui peut être utilisée pour préfixer des classes de processus paramétrées.

Les processus sont des coroutines (COW 63) s'exécutant en quasi-parallélisme grâce à un échéancier standard.

II-1.2.- Utilisation de fichiers

Afin de séparer nettement la prise des mesures, l'édition des résultats généraux et l'analyse statistique, nous utilisons divers fichiers séquentiels (ce sont les seuls autorisés dans la version actuelle de SIMULA 67 disponible à l'IUCAL). Ils sont au nombre de quatre :

- fichier_FGEN : il contient les résultats généraux concernant les régions, les ressources, les structures de données (et les files) ; il est créé à la fin de la simulation et permet ainsi une édition séparée.
- fichier_EVOL : il contient les séries chronologiques des mesures concernant la valuation de certaines structures de données (mesures effectuées à intervalles réguliers au cours de la simulation).
- fichier_FOCC : il contient les séries chronologiques relatives au nombre d'accès utilisés.
- fichier_FREP : il contient les séries chronologiques relatives aux temps de séjour des transactions dans chaque région.

Ces trois derniers fichiers sont constitués au cours de la simulation et servent à deux fins : déterminer l'état du système et calculer un intervalle de confiance sur la moyenne.

Nous ne détaillons ni leur forme ni leur contenu : ceux-ci sont présentés dans (THA 79).

La création et l'obtention de ces fichiers sont obtenues en valant divers paramètres du programme maquette concernant les objets décrits et le lancement de la simulation (voir 2.4 et 2.5).

II-2.- Description des objets spécifiques aux mesures

II-2.1.- Les transactions

Les objets (au sens de SIMULA) qui entrent dans les structures de données ou les régions appartiennent à la classe transaction ou à une de ses sous-classes.

Pour diverses raisons exposées dans (KLE 79, DUF 80 a), la classe transaction est une sous classe de la classe message :

- ```
(1) link class message (no) ; integer no ; null ;
(2) message class transaction (pds) ; real pds ;
 begin
(3) real hdebrg ;
(4) procédure modifierpoids (p) ; real p ;...;
 end ; $ transaction $
```

- (1) entête de la classe message  
no est un entier permettant de repérer le message
- (2) entête de la classe transaction  
pds est un réel définissant le poids associé à la transaction
- (3) hdebrg est une variable réelle donnant l'heure d'entrée de la transaction dans une région
- (4) l'appel de la procédure modifierpoids permet de donner un nouveau poids p à la transaction

### II-2.2.- Les régions

Un objet de cette classe est essentiellement un compteur modifié à chaque mouvement de transaction.

Une région est un objet d'une classe prédéfinie trégion dont la structure est la suivante :

- ```
(1) link class trégion (nom) ; text nom ;
    begin
(2)     boolean precis, bhisto ;
(3)     ref (histogramme) h ;
```

- ```
(4) integer rgent, rgsort, rgmax, noreg ;
(5) real rgtpsmin, rgtpsmax, rgmoy, rgtpsatsat, hmoy ;
(6) procédure entrer (t) ; ref (transaction) t ;...;
(7) procédure sortir (t) ; ref (transaction) t ;...;
(8) procédure reinit ;...;
(9) noreg := numreg ;
(10) hmoy := time ;
 precis := false ;
 bhisto := false ;
 rgent := rgsort := rgmax := 0
 rgtpsmin := rgtpsmax := rgmoy := rgtpsatsat := 0 ;
(11) into (regionq) ;
 end ; $ tregion $
```

- (1) entête de la classe tregion  
nom est un libellé permettant d'identifier la région
- (2) bhisto est un booléen vrai si on veut un histogramme sur la distribution des temps de séjour des transactions dans cette région  
bprecis est un booléen vrai si on veut calculer un intervalle de confiance sur le temps de séjour moyen.
- (3) h est une variable repérant un objet de type histogramme qui est utilisé lorsque bhisto est vrai
- (4) ces variables entières servent à cumuler le nombre d'entrées (rgent), de sorties (rgsort) et tenir à jour le contenu maximal (rgmax) dans cette région dont le numéro est noreg
- (5) ces variables réelles servent à cumuler le temps de séjour des transactions sorties (rgtpsatsat) en tenant à jour les temps de séjour minimal (rgtpsmin), maximal (rgtpsmax), le contenu moyen (rgmoy)  
hmoy est une variable réelle contenant l'heure simulée du dernier mouvement de transaction
- (6) l'appel de la procédure entrer assure la mise à jour des variables locales et celle de l'heure d'entrée simulée pour la transaction concernée

- (7) l'appel de la procédure sortir assure elle aussi la mise à jour de variables locales ; elle permet, si l'utilisateur l'a précisé de créer un nouvel enregistrement dans le fichier FREP.
- (8) l'appel de la procédure reinit a lieu lorsqu'une nouvelle simulation commence pour réinitialiser les variables locales.
- (9) numreg est une fonction-procédure qui fournit un numéro noreg.
- (10) initialisation des variables locales.
- (11) regionq est une liste de régions qui chaîne les différents objets de type tregion.

#### Exemple

Dans la maquette prototype, nous avons défini deux régions : 'administratif' et 'physique'. Leur déclaration et création se sont effectuées de la manière suivante :

```
ref (tregion) adm, phy ;
...
adm :- new tregion ('ADMINISTRATIF') ;
phy :- new tregion ('PHYSIQUE') ;
```

#### II-2.3.- Les structures de données

Dans le logiciel MAESTRO, jusqu'à présent les seuls types de structure de données réalisées sont des files d'attente où peuvent s'effectuer des blocages. Elles sont gérées selon une politique définie par l'utilisateur avec différentes possibilités d'accès ou de modifications.

Etant donné qu'une file d'attente est un cas particulier de structure de données, la classe tfile est une sous-classe de la classe tstructdonnee.

Nous donnons successivement les structures de ces deux classes :

- (1) monitor class tstructdonnee (mesure, equilibre, inter, dmes) ;  
real inter, dmes ;  
boolean mesure, equilibre ;  
begin

- (2) ref (condition) nonvide ;  
(3) integer sdpre, sdmax, sdnul, sdsort, nbmes, nosd ;  
(4) real tvol, sdtpsat, volmin, volmax, hmod, sdpds, sdmoy ;  
(5) procédure sdajouter (t) ; ref (transaction) t ;... ;  
(6) procédure sdmodifier (t,p) ; ref (transaction) t ;  
real p ;... ;  
(7) procédure sdsupprimer (t) ; ref (transaction)t ;... ;  
(8) procédure sddetruire ;... ;  
(9) procédure sdmesure ;... ;  
(10) real procédure volume ;... ;  
(11) procédure sdreinit ;... ;  
...  
hmod := time ;  
if equilibre or mesure then  
begin  
(12) new refs (this tstructdonnee).into (sdq) ;  
(13) nosd := numsd ;  
(14) if equilibre then activate new mesurevol (this  
tstructdonnee)  
qua process delay inter ;  
end ;  
(15) nonvide :- new condition (this monitor) ;  
end ; \$ tstructdonnee \$

- (1) entête de la classe définissant le type tstructdonnee : cette classe est préfixée par la classe monitor présentée dans (DUF 80a). Ceci signifie que tous les accès se font en exclusion mutuelle (II.1.1.1).  
mesure est un booléen vrai si et seulement si on veut obtenir des résultats généraux sur les volumes de données.  
equilibre est un booléen vrai si et seulement si on veut obtenir des résultats détaillés sur la série chronologique des valuations.  
inter est un nombre réel indiquant la périodicité des mesures sur le volume des données contenues dans cette structure de données.

- dmes donne l'heure simulée du début des mesures.
- (2) déclaration d'une variable nonvide repérant la condition associée au moniteur.
  - (3) déclaration de variables entières :
    - sdpre : nombre de transactions présentes.
    - sdmax : nombre de transactions maximal.
    - sdnul : nombre de fois où la structure de données est vide.
    - sdsort : nombre de transactions sorties.
    - nbmes : nombre de mesures sur le volume.
    - nosd : numéro associé à la structure de données.
  - (4) déclaration de variables réelles :
    - sdtpsat : cumul des temps de séjour des transactions sorties.
    - tvol : cumul des mesures sur les volumes.
    - volmin : volume minimal.
    - volmax : volume maximal.
    - hmod : heure simulée de la dernière modification.
    - sdpds : volume courant.
    - sdmoy : volume moyen.
  - (5) L'appel de la procédure sdajouter assure la mise à jour de variables locales lors de l'arrivée d'une transaction.
  - (6) L'appel de la procédure sdmodifier assure la mise à jour de variables locales lorsque le poids associé à la transaction repérée par t est modifié.
  - (7) L'appel de la procédure sdsupprimer permet de mettre à jour les variables locales lorsque la transaction concernée est sortie.
  - (8) L'appel de la procédure sddétruire assure la mise à jour de variables locales lorsqu'on vide la structure de données.
  - (9) L'appel de la procédure sdmesure permet de mesurer le volume de la structure de données.
  - (10) L'appel de la fonction-procédure volume fournit le volume courant.
  - (11) L'appel de la procédure reinit a lieu au début d'une nouvelle simulation pour réinitialiser les variables locales.

- (12) toutes les structures de données sont associées à des objets de la classe refsd qui sont chaînés dans une liste sdq.
- (13) numds est une fonction-procédure qui fournit un numéro nods.
- (14) si le booléen equilibre est vrai, il y a création d'un processus cyclique de mesure du volume qui est activé tous les inter unités de temps à partir de dmes.
- (15) création d'une condition nonvide associée au moniteur protégeant cette structure.

Considérons maintenant la structure de la classe tfile :

```
(1) tstructdonnee class tfile ;
 begin
(2) ref (head) q ;
(3) integer nofile ;
(4) procedure entrer (t) ; ref (transaction) t ;...;
(5) procedure modifier (t,p) ; ref (transaction) t ;
 real p ;...;
(6) ref (transaction) procedure sortir ;...;
(7) procedure enlever (t), ref (transaction) t ;...;
(8) ref (transaction) procedure suivant (t) ;
 ref (transaction) t ;...;
(9) ref (transaction) procedure premier ;...;
(10) integer procedure longueur ;...;
(11) procedure detruire ;...;
(12) procedure reinit ;...;
(13) q := new head ;
 if mesure or equilibre then begin
(14) nofile := numfile ;
(15) into (fileq) ;
 end ;
 end ; $ tfile $
```

- (1) entête de la classe tfile préfixée par tstructdonnee.
- (2) q est une liste contenant des transactions.
- (3) nofile est un entier donnant un numéro à la file.

- (4) l'appel de la procédure entrer a pour effet de ranger la transaction repérée par t en fin de liste.
- (5) l'appel de la procédure modifier assure la modification du poids associé à la transaction repérée par t.
- (6) l'appel de la fonction-procédure sortir a pour effet de délivrer la première transaction de q si elle existe (sinon le processus appelant est bloqué sur la condition nonvide).
- (7) l'appel de la procédure enlever permet de sortir de la liste q la transaction repérée par t.
- (8) l'appel de la fonction-procédure suivant permet d'accéder à la transaction suivant celle repérée par t, dans la liste q (si elle existe).
- (9) l'appel de la fonction-procédure premier permet d'accéder à la transaction de tête de liste q.
- (10) l'appel de la fonction-procédure longueur a pour effet de donner le nombre de transactions contenues dans la liste q.
- (11) l'appel de la procédure détruire assure le vidage de la file.
- (12) l'appel de la procédure reinit a pour effet de reinitialiser les variables locales au début d'une nouvelle simulation.
- (13) création d'une liste de transactions q.
- (14) numfile est une fonction-procédure qui délivre un numéro nofile.
- (15) fileq est une liste chaînant les objets de type tfile.

L'exécution des (fonction-)procédures est réalisée en exclusion mutuelle. On a ainsi la gestion d'un "tampon de taille infinie".

#### Exemple

Dans la maquette exemple, nous avons défini plusieurs structures de données de type tfile : 'courrier', 'ordres', 'bons1', 'bons2', qui sont déclarées et créées de la façon suivante :

```
ref (tfile) courrier, ordre, bons1, bons2 ;
```

```

```

```
courrier : - new tfile ('COURRIER', true, true, 24, 9.5) ;
```

```
ordres : - new tfile ('ORDRE', true, false, 24, 20) ;
```

```
bons1 : - new tfile ('BONS1', true, true, 24, 20) ;
```

```
bons2 : - new tfile ('BONS2', false, false, 0, 0) ;
```

La signification des paramètres effectifs est la suivante : on désire des résultats synthétiques sur les trois premières files et des résultats détaillés sur les volumes des files 'courrier' et 'bons1', qui sont évalués tous les jours, l'une à 9 H 30 et l'autre à 20 H.

#### II-2.4.- Les ressources

Pour collecter des résultats sur l'utilisation des ressources, nous avons défini la classe tressource qui sert à préfixer les classes concernant les différents types de ressources définies (KLE 79, DUF 80a).

La structure de cette classe prédéfinie est la suivante :

```
(1) monitor class tressource ;
 begin
(2) boolean precision ;
(3) real hdispo, hdeb, tpsdispo, tpsutil, tpsinutil, acces,
 rpxmaxdispo, rpxmaxutil, array util (1:10) ;
(4) integer rsort, nbutil, nbmaxutil, nores, type ;
(5) procedure statacquerir (n) ; real (n) ;...;
(6) procedure statliberer (n) ; real (n) ;...;
(7) procedure statutilisateur (nb) ; integer nb ;...;
(8) procedure statservis (nb) ; integer nb ;...;
(9) procedure statinit (n, typ) ; integer n ; typ ;...;
(10) procedure statreinit ;...;
(11) new refres (this tressource).into (ressourceq) ;
 end ;
```

- (1) entête de la classe tressource qui est préfixée par monitor.
- (2) precision est un booléen vrai si et seulement si on désire des résultats détaillés sur l'utilisation de la ressource.
- (3) déclaration de variables réelles :  
hdispo : date de début de disponibilité de la ressource  
hdeb : date de dernière modification sur l'utilisation des accès

- tpsdispo : cumul des durées de disponibilité  
tpsutil : cumul des durées d'utilisation des accès  
tpsutil : cumul des périodes d'inutilisation  
util (1:10) : tableau servant à cumuler les durées d'utilisation par nombre d'accès utilisés
- acces : nombre courant d'accès utilisés  
rpxmaxdispo : nombre maximal d'accès disponibles  
rpxmaxutil : nombre maximal d'accès utilisés
- (4) déclaration de variables entières :  
rsort : nombre de processus ayant utilisés la ressource  
nbutil : nombre de processus utilisateurs de la ressource  
nbmaxutil : nombre maximal de processus utilisateurs  
nores : numéro de la ressource  
type : type de la ressource
- (5) l'appel de la procédure statacquerir permet la mise à jour de variables locales lorsqu'il y a acquisition de certains accès de la ressource et crée un nouvel enregistrement sur le nombre d'accès utilisés dans le fichier Foccc si l'utilisateur l'a demandé.
- (6) l'appel de la procédure statliberer a un rôle similaire lors de la restitution d'accès.
- (7) l'appel de la procédure statutilisateur assure la mise à jour du nombre de processus utilisant en même temps des accès pour s'exécuter.
- (8) l'appel de la procédure statservis assure la mise à jour du nombre de processus ayant requis des accès de cette ressource.
- (9) l'appel de la procédure statinit assure l'initialisation des variables locales.
- (10) l'appel de la procédure stareinit assure la réinitialisation des variables locales lorsque commence une nouvelle simulation.
- (11) toutes les ressources sont associées à des objets de la classe refres qui sont chaînés dans la liste ressourceq.

Remarque : Comme la classe tressource sert à préfixer les autres classes de ressources (DUF 80a), elle est donc complètement transparente à l'utilisateur.

#### Exemple

Dans la maquette exemple, nous avons les ressources suivantes :

- x, z et p : ressources critiques avec calendrier modélisant chacune une personne.
- y : ressource critique préemptible avec priorité possédant un calendrier (modélisation d'une personne).
- ord : ressource à nombre infini d'accès (ordinateur).

On a alors les déclarations et créations suivantes :

```
ref (calrc) x, z, p ;
ref (calrcpp) y ;
ref (raminf) ord ;
...
x : - new calrc ('X', true, true, 24,0) ;
z : - new calrc ('Z', true, true, 24,0) ;
p : - new calrc ('P', true, false, 24,0) ;
y : - new calrcpp ('Y', true, false, 24,0) ;
ord : - new raminf ('ORD', false, false, 24,0) ;
```

Pour ces différentes classes (DUF 80a), la signification des trois premiers paramètres concerne le nom donné à la ressource, l'existence d'un calendrier et la possibilité pour un processus utilisateur de terminer en fin de période de disponibilité. Les deux derniers paramètres sont spécifiques au mécanisme de calendrier (DUF 80a), (cycle du calendrier et heure de début).

#### II-3.- Processus prise de mesure

Si on désire obtenir des résultats statistiques approfondis, relatifs à l'évolution de volume de données contenues dans une structu-

re, il y a alors création d'un processus cyclique qui mesure le volume à intervalles réguliers. Ce processus appartient à la classe mesurevol dont la structure est la suivante :

```
(1) process class mesurevol (sd) ; ref (tstructdonnée) sd ;
 begin
 while true do inspect sd do
 begin
(2) ...
(2) sdmeasure ;
(3) if rechequil then ecrivvol (nbpds, nosd, volume) ;
(4) reactivate this process delay inter ;
 end ;
 end ; $ mesurevol $
```

- (1) entête de la classe mesurevol préfixée par process sd est un paramètre repérant la structure de données considérée.
- (2) l'appel de la procédure sdmeasure (voir 2.2.3) assure la mise à jour des variables locales à la structure de données repérée par sd.
- (3) Si on veut obtenir des résultats détaillés sur l'évolution des volumes de données contenus dans certaines structures, le booléen rechequil est positionné à vrai. Dans ces conditions, l'appel de la procédure ecrivvol a pour effet de créer un nouvel enregistrement concernant cette mesure de volume dans le fichier FVOL.
- (4) ce processus cyclique activé pour la première fois à la date dmes (voir 2.2.3) est réactivé tous les inter unités de temps.

#### II-4.- Exécution d'une simulation

Après avoir écrit une maquette avec les outils de base (KLE 79, DUF 80a), assuré la création des objets correspondants et l'activation des processus, on peut fixer les conditions de déroulement de la simulation. A cet effet, l'utilisateur doit paramétrer la procédure simul qui a la structure suivante :

```
(1) procédure simul (dureesimul, resgen, equilibre, brep, bocc) ;
 boolean resgen, equilibre, brep, bocc ;
 real dureesimul ;
 begin
 ...
(2) hold (dureesimul) ;
 ...
 end ; $ simul $
```

- (1) entête de la procédure simul  
la signification des paramètres est la suivante :  
dureesimul : fixe la durée de la simulation.  
resgen : booléen vrai si et seulement si on veut l'édition des résultats généraux (utilisation du fichier FGEN)..  
equilibre : booléen vrai si et seulement si on désire des résultats détaillés sur les volumes de données contenues dans certaines structures (utilisation du fichier FVOL).  
brep : booléen vrai si et seulement si on veut des résultats détaillés sur les temps de réponse (utilisation du fichier FREP).  
bocc : booléen vrai si et seulement si on veut obtenir des résultats détaillés sur l'utilisation des ressources (utilisation du fichier FOCC).
- (2) la durée de cette simulation est dureesimul.

Plusieurs appels consécutifs de la procédure simul permettent d'effectuer plusieurs simulations consécutives. Ceci permet notamment de faire exécuter le modèle sans tenir compte des mesures (phase transitoire) en donnant différentes valeurs aux paramètres effectifs.

#### Exemple

Dans la maquette exemple dont le texte est donné en annexe, on y trouve deux appels successifs de la procédure simul :

```

simul (120, false, false, false, false) ;
...
simul (1440, true, true, true, true) ;

```

Le premier appel correspond à une simulation de la maquette pendant 120 heures simulées (5 jours) de manière à ce que le système se trouve en "équilibre". De cette phase transitoire, on ne conserve aucun résultat.

Le second appel permet l'exécution de la maquette pendant une durée simulée de 60 jours en enregistrant tous les résultats.

#### II-5.- Procédures relatives à l'obtention de résultats détaillés

Nous avons écrit diverses procédures permettant de spécifier quels sont les objets sur lesquels on désire d'autres renseignements :

- (1) procedure preciseressource (res) ; ref (tressource) res ;  
...
- (2) procedure preciseregion (rg) ; ref (tregion) rg ;...
- (3) procedure historegion (rg, n, inf, pas, larg, haut) ;  
ref (tregion)rg ; real inf, pas ;  
integer n, larg, haut ;  
... ;

- (1) l'appel de la procedure preciseressource permet d'obtenir des résultats détaillés sur l'occupation de la ressource repérée par res en utilisant le fichier FOCC.
- (2) l'appel de la procedure preciseregion a un rôle identique sur les temps de réponse concernant la région repérée par rg en utilisant le fichier FREP.
- (3) l'appel de la procedure historegion permet d'obtenir un histogramme sur les temps de séjour dans la région repérée par rg.

Les paramètres de cette procédure ont la signification suivante :

n : nombre de classes de l'histogramme  
inf : borne inférieure

pas : taille d'un intervalle.  
larg : paramètre de mise en page (largeur d'une classe).  
haut : paramètre de mise en page (hauteur de la classe d'effectif max.).

En outre, l'utilisateur peut obtenir des histogrammes sur tout ensemble de valeurs qu'il fournit. Pour ce faire, nous avons défini une classe d'objets histogramme qui a la structure suivante :

- (1) link class histogramme (ident, n, inf, pas, larg, haut) ;  
text ident ;  
integer n, larg, haut ;  
real inf, pas ;  
begin  
...  
(2) procedure ajout (val) ; real val ;... ;  
(3) procedure reinit ;... ;  
...  
(4) into (histoq) ;  
end ; \$ histogramme \$

#### (1) entête de la classe histogramme

Les attributs ont la signification suivante :

ident : nom associé à l'histogramme.  
n : nombre de classes de l'histogramme.  
inf : borne inférieure.  
pas : taille d'une classe.  
larg : paramètre d'édition en largeur.  
haut : paramètre d'édition en hauteur.

- (2) l'appel de la procedure ajout permet d'ajouter la valeur val dans l'histogramme.
- (3) l'appel de la procedure reinit a lieu au début d'une nouvelle simulation pour réinitialiser les variables locales.
- (4) histoq est une liste chaînant les objets de type histogramme.

Exemple

Dans la maquette exemple (annexe ), les résultats détaillés sur les régions 'administratif' et 'physique' (adm et phy) et sur les ressources z et p sont obtenus par les instructions suivantes :

```
preciseregion (adm) ;
preciseregion (phy) ;
preciseressource (z) ;
preciseressource (p) ;
```

Désirant compléter les résultats sur la région 'administratif' (adm) par un histogramme, nous avons l'instruction suivante :

```
historegion (adm, 15,1,24,1,0) ;
```

Par ailleurs, ne pouvant créer une région système (régions disjointes) concernant tout le système décrit ; nous avons recours à un histogramme 'système' pour connaître la distribution des durées de traitements des commandes :

```
ref (histogramme) st ;
```

```
...
```

```
st : -new histogramme ('SYSTEME' 15,0,24,1,35) ;
```

Ceci nous permet d'obtenir un histogramme comportant 15 classes, de borne inférieure 0 (heure), de pas : 24 (heures) qui sera édité avec les paramètres de mise en page suivants en largeur (1) et en hauteur (35).

Cet histogramme est mis à jour chaque fois qu'une commande est soldée administrativement. Si tps est sa durée de traitement, cette mise à jour s'effectue par l'appel de la procédure ajou :

```
st.ajou (tps) ;
```

II-6.- Obtention des résultats

Lorsque la simulation est terminée, il est possible d'exploiter les fichiers constitués durant son exécution. Les programmes que nous avons écrits servent à :

- l'analyse de séries chronologiques,
- l'édition des résultats généraux,
- le calcul d'intervalles de confiance de moyenne,

II-6.1.- Analyse de séries chronologiques

Le calcul d'intervalle de confiance de moyenne et l'obtention de résultats de type moyenne supposent que le système soit en "équilibre".

D'après la définition retenue (IV.2.5), la vérification de cet état s'obtient en analysant plusieurs séries chronologiques portant sur :

- des volumes de données,
- l'utilisation de ressources,
- des temps de réponse.

Nous admettons que le système est en "équilibre" si les processus stochastiques sous-jacents aux séries chronologiques sont quasi-stationnaires.

Pour ce faire, nous avons écrit un programme EQUILIBRE dont le but est de rechercher l'existence de composantes caractéristiques dans une chronique (tendances, cycles). Sa structure et son fonctionnement sont identiques aux programmes E1-SERIE et E2-SERIE (voir V.2).

II-6.2.- Edition des résultats généraux

Le programme STATGEN permet d'obtenir des résultats synthétiques (voir 1.3.1) sur les volumes de données, les ressources et les régions en utilisant le fichier FGEN.

Comme sa structure et son fonctionnement sont transparents à l'utilisateur, nous n'en parlerons pas plus en renvoyant à (THA 79).

II-6.3.- Calcul d'intervalle de confiance

C'est un programme paramétré PRECISION qui permet d'obtenir un intervalle de confiance sur les volumes de données moyens, les

taux d'occupation moyens et les temps de réponse moyens : il met en oeuvre la méthode des blocs (III.2.2.2) pour estimer des intervalles de confiance de moyennes de processus stochastiques sous-jacents aux séries chronologiques des mesures enregistrées dans les fichiers FVOL, FOCC et FREP suivant le paramétrage.

Nous ne détaillons ni la structure ni le fonctionnement de ces programmes qui sont transparents à l'utilisateur (THA 79).

### III - CONCLUSION

Dans ce chapitre, nous avons présenté les outils de mesure et d'analyse conçus et réalisés dans le cadre du projet MAESTRO.

Afin de bien séparer la prise des mesures et leur analyse, nous avons choisi de donner à l'utilisateur la possibilité de constituer plusieurs fichiers (FGEN, FVOL, FOCC, FREP). Ceux-ci peuvent être exploités par programmes (2.6). Ceci permet entre autres d'obtenir des résultats comme ceux montrés dans le chapitre IV (2.5) à partir de la maquette exemple.

Cependant, il est possible d'exploiter différemment ces fichiers en effectuant par exemple des analyses de chroniques plus approfondies. Ceci peut se faire en utilisant des méthodes étudiées dans le cadre de ce projet (chapitre III) et mises en oeuvre en partie (chapitre V). Mais ceci n'aura vraiment d'intérêt qu'en améliorant en autres le paramétrage de la maquette pour que les résultats soient significatifs.

CONCLUSION

Nous avons présenté les outils conçus et réalisés pour évaluer quantitativement des maquettes de système d'information en fonctionnement.

Pour ce faire, la méthodologie adoptée est celle de la simulation à événements discrets. Nous y retrouvons les problèmes inhérents à ce type d'approche. Pour chaque difficulté rencontrée, nous avons présenté plusieurs solutions pour expliquer et justifier celles que nous avons retenues. Dans cet exposé d'ordre méthodologique, nous avons élaboré une synthèse, tout au moins en évaluation de performances de maquettes, sur le paramétrage, l'analyse des entrées et des sorties, la précision statistique des résultats, la validation et la robustesse, tout ceci d'un point de vue utilisateur.

La résolution des difficultés rencontrées passe très souvent par l'utilisation de techniques statistiques (analyse de séries chronologiques, estimations, calcul d'intervalle de confiance entre autres).

Bien que très intéressant, cette évaluation de performances par maquettes exécutées selon une technique de simulation discrète doit être complétée par d'autres méthodes. Les résultats obtenus doivent être recoupés ou comparés avec d'autres obtenus soit à partir de modèles analytiques soit à partir de mesures. C'est un problème de validation pour lequel notre expérience est encore insuffisante.

En outre, il nous paraît important d'acquérir une meilleure connaissance sur la validité d'une maquette (influence des hypothèses de réduction et simplification implicites ou explicites dans la description) et d'approfondir les notions de niveau de détail ou de robustesse d'une maquette, compte tenu des résultats souhaités.

Le dépassement des difficultés rencontrées passe par le développement d'études expérimentales afin de dégager les facteurs essentiels influant sur les performances, en fonction des objectifs fixés.

Par ailleurs, il nous semble que l'approche proposée et les outils réalisés peuvent être utilisés dans un domaine plus large que celui des systèmes d'information en considérant par exemple d'autres types de processus dans l'entreprise.

BIBLIOGRAPHIE

- ABA 79 ABADIE J. et MESLIER F. - "Etude de l'utilisation des modèles ARIMA pour la prévision à très court terme de l'énergie journalière produite par EDF" - RAIRO - Vol. 13, N° 1 - février 1979.
- AGR 70 AGRAWALA A.K. - "Learning with a probabilistic teacher" - IEEE Trans. on Information Theory - Vol. IT 16, N° 4 - mai 1970.
- AIV 70 AIVAZIAN S. - "Etudes statistiques des dépendances" - MIR - Moscou 1970.
- AND 71 ANDERSON T.W. - "The Statistical Analysis of Time Series" - Wiley - 1971.
- AND 75 ANDERSON O.D. - "Time Series Analysis and Forecasting" - Butterworths - 1975.
- BAR 56 BARTLETT M.S. - "STOCHASTIC PROCESSES Methods & Applications" - Cambridge - University Press - 1956.
- BAR 71 BARBUT M., FOURGEAUD D. - "Elements d'analyse mathématique des chroniques" - Hachette Université - 1971.
- BAS 75 BASKETT F., CHANDY K.M., MUNTZ R.R., PALACIOS F.G. - "Open, closed and mixed networks of queues with different classes of customers" - JACM - Vol. 22, N° 2 - avril 1975.
- BEN 66 BENDAT J., PIERSOL G. - "Measurement and Analysis of Random Data" - Wiley - 1966.
- BOD 79 BODARD F., PIGNEUR Y. - "A model and a language for functional specifications and evaluation of information system dynamics, Formal Models and Practical Tools for Information System Design" - H.J. Schneider, North-Holland-1979.
- BOX 70 BOX G.E.P., JENKINS G.M. - "Time Series Analysis Forecasting and Control" - Holden-Day - 1970.
- CAR 69 CARNAHAN B., LUTHER H.A., WILKES J.A. - "Applied Numerical Methods" - Wiley - 1969.

- CHA 74 CHANG A., LAVENBERG S.S. - "Work - rates in closed queueing network model" - Operations Research - Vol. 22, N° 4 - 1974.
- CHA 75a CHANDY K.M., HERZOG U., WOO L. - "Parametric analysis of queueing network model" - IBM Journal of Research and Development - Janvier 1975.
- CHA 75b CHANDY K.M., HERZOG U., WOO L. - "Approximate analysis of general queueing networks" - IBM Journal of Research and Development - Janvier 1975.
- CHA 77 CHATFIELD C. - "Some Recent Developments in Time - series Analysis" - JRSA A - Vol. 140, N° 4 - 1977.
- CII 72 CII - "SIMULA sous SIRIS 7/8 - Manuel de présentation et d'utilisation" - CII - CIDOC - Louveciennes - 1972.
- CON 63 CONWAY R.W. - "Some tactical problems in digital simulation" - Management Sciences - Vol. 10, N° 1 - Octobre 1963.
- CON 71 CONOVER W.J. - "Practical non parametric Statistics" - Willey - 1971.
- COU 77 COURTOIS P.J. - "Decomposability, Queueing and Computer System Application" - ACM Monograph Series - Academic Press - 1975.
- COW 63 CONWAY M.E. - "Design of a Separable Transition - Diagram Compiler" - CACM - Vol. 6, N° 7 - 1963.
- COX 69 COX D.R., LEWIS P.A.W. - "L'analyse statistique des séries d'événements" - Dunod - 1969.
- CRA 74 CRANE M.A., IGLEHART D.L. - "Simulating stable stochastic systems I : General multiserver queues - II : Markov chains" - JACM - Vol. 21, N° 1 - January 1974.
- CRA 75a CRANE M.A., IGLEHART D.L. - "Simulating Stochastic Systems III : Regenerative process and Discrete Event Simulations" - Operations Research - Vol. 23, N° 1 - February 1975.

- CRA 75b CRANE M.A., IGLEHART D.L. - "Simulating stable stochastic systems IV : approximation techniques" - Management Sciences - Vol. 21, N° 11 - July 1975.
- DAH 68 DAHL O.J. - "Discrete event simulation languages, Programming Languages" - Ed. F. Genuys - ACADEMIC PRESS - 1968.
- DAH 70 DAHL O.J., MIHRHAUG B., NYGAARD K. - "SIMULA 67 common base language" - Pub. S-22 - Norwegian Comp. Center - 1970.
- DAV 79 DAVIES N., NEWBOLD P. - "Some power studies of a portman-teau test of time series model specification" - Biometrika - Vol. 66, N° 1 - 1979.
- DAY 69 DAY N.E. - "Estimating the components of a mixture of normal distributions" - Biometrika - Vol. 56, N° 3 - 1969.
- DER 79 DERNIAME J.C., FINANCE J.P. - "Types abstraits de données : spécification, utilisation, réalisation" - Cours Ecole d'été d'informatique - Monastir - juillet 1979.
- DOO 53 DOOB J.L. - "Stochastic Processes" - Wiley - 1953.
- DUF 80a DUFORD J.F. - "Maquettes pour évaluer les systèmes d'information des organisations" - Thèse d'état - Nancy - janvier 1980.
- DUF 80b DUFORD J.F. - "Description et évaluation par simulation des systèmes d'information" - Séminaire INFORSID groupe 2 Aix en Provence - janvier 1980.
- DUR 50 DURBIN J., WATSON G.S. - "Testing for serial correlation in least square regression" - Biometrika - Vol. 37 - 1950.
- DUR 60 DURBIN J. - "The fitting of time series models" - Rev. Inst. 28 - 1960.
- FER 78 FERRARI D. - "Computer Systems Performance evaluation" - Prentice - Hall - 1978.

- FIS 67 FISHMAN G.S. - "Problems in the statistical analysis of simulation experiments: the comparison of means and the length of sample records" - CACM - Vol. 10, N° 2 - Februa-ry 1967.
- FIS 73 FISHMAN G.S. - "Concepts and Methods in Discrete Event Digital Simulation" - Wiley - 1973.
- FOR 61 FORRESTER J.W. - "Industrial Dynamics" - MIT Press- 1961.
- FOU 67 FOURGEAUD D.C., FUCHS A. - "Statistique" - Dunod - 1967.
- GEK 76 GELENBE E., KURINCKS A. - "Random injection control of multiprogramming in virtual memory" - Proc. 2 nd Interna-tional Workshop on Modelling and performance evaluation of Computer Systems - E. Gelenbe ed. North-Holland - Publis-hing Company - 1976.
- GEL 75a GELENBE E., LEROUDIER J. - "Cours de modèles de systèmes informatiques" - Ecole internationale d'été informatique AFCET - Rabat - Juillet 1975.
- GEL 75b GELENBE E., PUJOLLE G. - "Probabilistic models of compu-ter systems, part II" - Rapport IRIA-Laboria - N° 147 - décembre 1975.
- GEL 77 GELENBE E., PUJOLLE G., - "A diffusion model for multiple class queueing networks" - Rapport IRIA - Laboria - N° 242 - août 1977.
- GRA 77 GRANGER C.W.J., NEWBOLD P. - "Forecasting Economic Time Series" - Academic Press - 1977.
- GUT 78 GUTTAG J.V., HOROWITZ P., MUSSER D.R. - "Abstract Data Types and software Validation" - CACM - Vol. 21, N° 12 - 1978.
- HAN 60 HANNAN E.J. - "Times Series Analysis" - Methuen's mono-graphs on applied probability and statistics - Londres - 1960.

- HOA 74 HOARE C.A.R. - "Monitors : An Operating System Structuring Concept" - CACM - Vol. 17, N° 10 - Octobre 1974.
- IBM 70 IBM - "General Purpose Simulation System/360 User's Manual" - GH 20-0326 - 1970.
- JAC 63 JACKSON J.R. - "Jobshop - like queueing systems" - Manage-ment Sciences - Vol. 10, N° 1 - Octobre 1963.
- JAI 78 JAIN A.K. - "A guideline to statistical approaches in com-puter performance evaluation studies" - ACM SIGMETRICS PER-Vol. 7, N° 1,2 - 1978.
- JOM 78 JOMIER G., KURINCKS A. - "Application de la méthode de dé-composition au traitement de modèles de systèmes informati-ques" - Rapport IRIA - Laboria - N° 319 - Juillet 1978.
- KIV 69 KIVIAT P.J., VILLANUEVA R., MARKOWITZ H. - "The SIMSCRIPT II Programming language" - Prentice Hall - Englewood Cliffs - N.J. - 1969.
- KLE 74 KLEIJNEN J.P.C. - "Statistical Techniques in Simulation" - part I et II - Dekker - 1974.
- KLE 76 KLEINROCK L. - "Queueing Systems : computer applications" - Wiley - 1976.
- KLE 79 KLEIN P. - "Concepts et outils pour l'écriture de Maquettes dans le projet MAESTRO" - Thèse de 3ème cycle - Nancy - Mai 1979.
- KNU 68 KNUTH D.E. - "The Art of Computer Programming" - Addison - Wesley - 1968.
- KOB 74a KOBAYASHI H. - "Application of the diffusion approximation to queueing networks I : Equilibrium distributions" - JACM - Vol. 21, N° 2 - April 1974.
- KOB 74b KOBAYASHI H. - "Application of the diffusion approximation to queueing networks II : non-équilibre distributions and applications to computer system modelling" - JACM - Vol. 21, N° 3 - July 1974.

- KOB 78 KOBAYASHI H. - "Modelling and Analysis : An Introduction to System Performance Evaluation Methodology" - Addison - Wesley - 1978.
- LAV 75 LAVENBERG S.S., SLUTZ D.R. - "Introduction to regenerative simulation" - IBM Journal of Research and Development - september - 1975.
- LAW 79 LAW A.M., CARSON J.S. - "A Sequential Procedure for Determining the Length of a Steady-State Simulation" - Operations Research - Vol. 27, N° 5 - Sept., Oct. 1979.
- LER 77 LEROUDIER J. - "Systèmes adaptatifs à mémoire virtuelle" - Thèse d'état - Grenoble - mai 1977.
- LEW 76 LEWIS P.A.W. - "SASE IV and the statistical analysis of series of events in computer systems" - Rapport IRIA - Laboria - N° 198 - Novembre 1976.
- LIN 63 LINNIK Y.V. - "Méthode des moindres carrés" - Dunod - 1963.
- LIN 78 LING R.F. - "A study of the Accuracy of some Approximations for  $t$ ,  $\chi^2$  and F Tail Probabilities" - JASA - Vol. 73, N° 362 - June 1978.
- MAR 66 MARKOWITZ H.M. - "Simulating with SIMSCRIPT" - Management Sciences - Vol. 12, N° 10, June 1966.
- MAR 76 MARIE R. - "Approximation et Application aux réseaux de files d'attente" - Rapport IRISA - N° 65 - Université de Rennes - 1976.
- McC 78 Mac CLAVE J.T. - "Estimating the Order of Autoregressive Models: the Max  $\chi^2$  Method" - JASA - Vol. 73, N° 361 - mars 1978.
- MEP 78 MERLE D., POTIER D., VERAN M. - "A tool for computer system performance analysis" - International Conférence on Performance of Computer Installation - Gardone - juin 1978.

- MER 78 MERLE D. - "Algorithmes de calcul des probabilités stationnaires d'un réseau de files d'attente" - Rapport IRIA - Laboria - N° 279 - mars 1978.
- PAT 66 PATRICK E.A., HANCOCK J.C. - "Non supervised sequential classification and recognition of patterns" - IEEE Trans. on Information theory - Vol. IT 12, N° 3 - July 1966.
- PHL 79 PHILIPS L., BLOMME R. - "Analyse chronologique" - CABAY - Louvain la Neuve - 1979.
- POT 77 POTIER D. - "Modèles à files d'attente et gestion des ressources dans les systèmes informatiques" - Thèse d'état - Grenoble - janvier 1977.
- PRS 69 PRIESLEY M.B., SUBBARAO T. - "A test for non stationarity of time series" - JRSS - Vol. B 31, N° 1 - 1969.
- PUJ 78 PUJOLLE G. - "Modélisation et évaluation de performances des réseaux à commutation de paquets" - Thèse d'état - Paris 1978.
- QUE 79 Mac QUEEN D.B. - "Models for distributed computing" - Rapport IRIA - Laboria - N° 351 - avril 1979.
- REI 74 REISER M., KOBAYASHI H. - "Accuracy of the diffusion approximation for some queueing systems" - IBM Journal of Research and development 18,2 - mars 1974.
- SCH 75 SCHROEDER A. - "Reconnaissance des composants d'un mélange de distributions" - Rapport IRIA - Laboria - N° 104 - février 1975.
- SIE 56 SIEGEL S. - "Non parametric Statistics for the behavioral Sciences" - Mac Graw Hill - New-York - 1956.
- SIM 61 SIMON H., ANDO A. - "Aggregation of variables in dynamic systems" - Econometrika - Vol. 29, N° 2 - April 1961.
- SKI 67 SKINNER C.E. - "Priority queueing systems with Server-walking time" - Operations Research - Vol. 15, N° 2 - Mars-Avril 1967.

- STE 76 STEWARD W.J. - "MARCA, Markov Chain Analyser" - Rapport IRISA - Rennes 1976.
- THA 79 THAUREL M., DUFOURD J.F. - "Evaluation et outils statistiques dans le projet MAESTRO" - 3eme rapport partiel ATP - octobre 1979 - (CRIN 79-R-076).
- TSP 76 TSP - "Time Series Processor - User's Manuel" - BRONWYN H. HALU - 1976.
- VAU 77 VAUCHER J., BRATLEY P. - "Cours de Langages de Simulation" - Ecole internationale d'été informatique AFCET - Montreal - 1977.
- WAH 71 WAHBOR - "A polynomial algorithm for density estimation" - Ann. Math. Stat. - 1971.
- WAL 62 WALSH J.E. - "Handbook of non parametric Statistics" - SDC - 1962.

ANNEXE

EXEMPLE DE PROGRAMME MAQUETTE  
(GESTION DES COMMANDES-CLIENTS)

V75SIMULA SI,INTR  
 BEGIN

```

***** MODELE REDUIT *****


```

UTILISATION DE LA CLASSE MAESTRO

\*\*\*\*\*\$

MAESTRO BEGIN

```

TRANSACTION CLASS BC (DAT,TYPE,NL,NLL) ; REAL DAT ; CHARACTER TYPE ;
INTEGER NL,NLL ; NULL ;
CLASS TABHISTO (N,INF,PAS) ; INTEGER N ; REAL PAS,INF ;
BEGIN

```

```

REAL VX ,VF ;
INTEGER I ;
REAL ARRAY FN (1:N) ;
INTEGER ARRAY XI (1:N) ;
VX := INF ;
VF := 0 ;
FOR I := 1 STEP 1 UNTIL N DO
 BEGIN
 XI (I) := VX ;
 VF := VF + ININT ;
 FN (I) := VF ;
 VX := VX + PAS ;
 END ;
FOR I := 1 STEP 1 UNTIL N DO FN (I) := FN (I) / VF ;

```

```

END ; $ TABHISTO $
REAL PROCEDURE EXPONENT (DUR) ; VALUE DUR ; REAL DUR ;
BEGIN

```

```

REAL X ;
X := - LOG (UNIFORM (0,1,U)) / DUR ;
EXPONENT := X ;

```

```

END ; $ EXPONENT $
INTEGER PROCEDURE NORMALE (A,B,C) ; VALUE A,B,C ;
REAL A,B ;
INTEGER C ;
BEGIN

```

```

REAL NOR ;
NOR := C - 1 ;
WHILE NOR < C DO
 NOR := NORMAL (A,B,U) + 0.5 ;
 NORMALE := ENTIER (NOR) ;

```

```

END ; $ NORMALE $
REAL PROCEDURE REELNORMALE (A,B,C) ; VALUE A,B,C ;
REAL A,B,C ;
BEGIN

```

```

REAL NOR ;
NOR := C - 1 ;
WHILE NOR < C DO NOR := NORMAL (A,B,U) ;
 REELNORMALE := NOR ;

```

```

END ; $ REELNORMALE $
INTEGER PROCEDURE BINOMIALE (NB,P) ; VALUE NB , P ;
INTEGER NB ; REAL P ;
BEGIN

```

```

INTEGER K,PR;
PR := 0 ;
FOR K := 1 STEP 1 UNTIL NB DO
 IF DRAW (P,U) THEN PR := PR + 1 ;
 BINOMIALE := PR ;

```

```

END ; $BINOMIALE $

```

```

INTEGER PROCEDURE GENHISTO (TH) ; REF (TABHISTO) TH ;
BEGIN
 REAL V ;
 INTEGER X,I;
 V := UNIFORM (0,1,U) ;
 I := 1 ;
 WHILE TH.FN (I) <= V AND I < TH.N DO I := I + 1 ;
 X := TH.XI (I) ;
 GENHISTO := X ;
END ;
TFILE CLASS CCOMMANDES ;
BEGIN
 PROCEDURE MODIFIERCOM (COM,QTTELIVREE);VALUE QTTELIVREE;
 REF (BC) COM ;
 INTEGER QTTELIVREE ;
 BEGIN
 THIS MONITOR. ENTER ;
 COM.NL := COM.NL - QTTELIVREE ;
 COM.NLL := QTTELIVREE ;
 THIS MONITOR.LEAVE ;
 END ; * MODIFIERCOM *
 PROCEDURE PROPOSERLIV (COM,QTTEPROP) ; VALUE QTTEPROP ;
 REF (BC) COM ;

 INTEGER QTTEPROP ;
 BEGIN
 THIS MONITOR. ENTER ;
 COM.NLL := QTTEPROP ;
 THIS MONITOR.LEAVE ;
 END ; * PROPOSERLIV *
END ; * CCOMMANDES *
PROCESSUS CLASS GENERATEUR ;
BEGIN
 WHILE TRUE DO
 BEGIN
 INTEGER I,NBC; * NOMBRE DE COMMANDES *
 REF (MESSAGE) MESS ;
 ACTIVATIONS.ACQUERIR (MESS) ;
 MESS := NONE ;
 NBC := GENHISTO (NCE) ;
 HNCE.AJOU (NBC) ;
 FOR I := 1 STEP 1 UNTIL NBC DO
 BEGIN
 REF (BC) B ;
 INTEGER NBL ;
 NBL := GENHISTO (LCE) ;
 HLCE.AJOU (NBL) ;
 B := NEW BC (NUMEROTATION,NBL,TIME,"E",NBL,0) ;
 COURRIER.ENTRER (B) ;
 ECR.ENTRER (B) ;
 END ;
 HOLD (0)
 END ;
END ; * GENERATEUR *

```

```

PROCESSUS CLASS GENETEL ;
BEGIN
 WHILE TRUE DO
 BEGIN
 CHARACTER C ;
 INTEGER NLL ; * NOMBRE DE LIGNES *
 REF (BC) B ;
 IF Y.HEURETRAVAIL
 THEN BEGIN
 IF DRAW (1/3,U) THEN C := "T"
 ELSE C := "D" ;
 NLL := GENHISTO (LCT) ;
 B := NEW BC (NUMERO,NLL,TIME,C,NLL,0) ;
 BDITE. ENVOYER (B)
 END ;
 HOLD (NEGEXP (17/8,U)) ;
 END ;
END ; * GENETEL *
PROCESSUS CLASS ENREGISTREMENT ;
BEGIN
 WHILE TRUE DO
 BEGIN
 REF (HEAD) L ; * LISTE INTERMEDIAIRE *
 REF (MESSAGE) MESS ;
 REF (BC) B ;
 REAL DUREE ;
 TEXT LIB ;
 LIB := 'ENREGISTREMENT1' ;
 ACTIVATION1.ACQUERIR (MESS) ;
 MESS := NONE ;
 L := NEW HEAD ;
 WHILE COURRIER.LONGUEUR > 0 DO
 BEGIN
 B := COURRIER.SORTIR ;
 X.ACQUERIR ;
 DUREE := 0.026 + (B.NL * 0.012) ;
 HOLD (DUREE) ;
 X.LIBERER ;
 B.INTO (L) ;
 END ;
 WHILE L.CARDINAL > 0 DO
 BEGIN
 B := L.FIRST QUA BC ;
 B.OUT ;
 X.ACQUERIR ;
 HOLD (0.012 + 0.012 * B.NL) ;
 ORD.ACQUERIR ;
 HOLD (0.002 + 0.001 * B.NL) ;
 ORD.LIBERER ;
 X.LIBERER ;
 COMMANDES.ENTRER (B) ;
 ECR.SORTIR (B) ;
 ADM.ENTRER (B) ;
 END ;
 END ;
END ; * ENREGISTREMENT *

```

```

PROCESSUS CLASS SAISIE ;
BEGIN
 WHILE TRUE DO
 BEGIN
 REF (MESSAGE) MESS ;
 REAL DUREE ;
 BOITE.ACQUERIR (MESS) ;

 Y.ACQUERIR ;
 IF MESS QUA BC.TYPE = "D"
 THEN DUREE := 0.08
 ELSE BEGIN
 DUREE := 0.005 + 0.008 + (MESS QUA BC.NL * 0.008) ;
 MESS QUA BC.NO := NUMEROTATION ;
 END ;
 HOLD (4 * DUREE / 5) ;
 ORD.ACQUERIR ;
 HOLD (DUREE / 5) ;
 ORD.LIBERER ;

 TEL.ENTRER(MESS QUA BC) ;

 IF MESS QUA BC.TYPE = "T"
 THEN BEGIN
 HLCT.AJOU (MESS QUA BC.NL) ;
 COMMANDES.ENTRER (MESS QUA BC) ;
 ADM.ENTRER(MESS QUA BC) ;
 END ;
 MESS := NONE ;
 Y.LIBERER ;
 END
END ; $ SAISIE $
PROCESSUS CLASS PREPARATION ;
BEGIN
 WHILE TRUE DO
 BEGIN
 REF (BC) B,B1 ;
 REAL T1,T2,T3 ;
 B := ORDRES.SORTIR ;
 P.ACQUERIR ;
 T1 := B.NLL * 0.07 ;
 T2 := B.NLL * 0.01 ;
 T3 := B.NLL * 0.05 ;
 HOLD (REELNORMALE (T1,T2,T3)) ;
 P.LIBERER ;
 BONS1.ENTRER (B) ;
 INSPECT B DO B1 := NEW BC (NO,PDS,DAT,TYPE,NL,NLL) ;
 BONS2.ENTRER (B1) ;

 PHY.SORTIR (B) ;
 END ;
END ; $ PREPARATION $
PROCESSUS CLASS VIDEUR (BONS,ACT) ; REF (TFILE) BONS ;
REF (ACTMONITOR) ACT ;
BEGIN
 WHILE TRUE DO
 BEGIN
 REF (MESSAGE) MESS ;
 ACT.ACQUERIR (MESS) ;
 BONS.DETRUIRE ;
 END ;
END ; $ VIDEUR $

```

```

PROCESSUS CLASS DECISION ;
BEGIN
 WHILE TRUE DO
 BEGIN
 TEXT LIB ;
 REF (MESSAGE) MESS ;
 REF (BC) B,B1 ;
 INTEGER I,NC,NLLT ; $ NC NB COMMANDES NLLT NB TOT LIGNES $
 REAL LIV ; $ % LIGNES A LIVRER $
 LIB := 'DECISION1' ; $ EDITION PROPOSITIONS LIVRAISON $
 ACTIVATION2.ACQUERIR (MESS) ;
 ORD.ACQUERIR ;
 NC := COMMANDES.LONGUEUR ;
 NLA := COMMANDES.VOLUME ; $ NB LIGNES EN ATTENTE $
 NLLT := 0 ; $ NB LIGNES A LIVRER $
 IF NLA > 0 THEN LIV := (NLA - NLREST)/NLA ELSE LIV := 0 ;
 B := COMMANDES.PREMIER ;
 FOR I := 1 STEP 1 UNTIL NC DO
 BEGIN
 INTEGER QTE ;
 QTE := BINOMIALE (B.NL,LIV) ;
 NLLT := NLLT + QTE ;
 COMMANDES.PROPOSERLIV (B,QTE) ;
 B := COMMANDES.SUIVANT (B) ;
 END ;
 HOLD (0.25) ;
 ORD.LIBERER ;
 LIB := 'DECISION2' ; $ PRISE DE DECISION $
 Y.ACQUERIR ;
 HOLD (NC * 0.007) ;
 Y.LIBERER ;
 LIB := 'DECISION3' ; $ SAISIE DES MODIFICATIONS $
 Z.ACQUERIR ;
 HOLD (NLA * 0.040 / 10) ;
 ORD.ACQUERIR ;
 HOLD (NLA * 0.020 / 10) ;
 ORD.LIBERER ;
 Z.LIBERER ;
 LIB := 'DECISION4' ;
 $ EDITION DES BL MAJ DES COMMANDES $
 ORD.ACQUERIR ;
 B := COMMANDES.PREMIER ;
 FOR I := 1 STEP 1 UNTIL NC DO
 IF B.NLL > 0 THEN
 BEGIN
 $ LIGNES A LIVRER $
 IF B.NL = B.NLL
 THEN BEGIN
 $ COMMANDE SOLDEE $
 B1 := B ;
 B := COMMANDES.SUIVANT (B1) ;
 COMMANDES.ENLEVER (B1) ;
 OUTIMAGE ;
 ECRI (TIME) ;
 OUTTEXT (' LA FACTURE NUMERO ') ;
 OUTINT (B1.NO,4) ;
 OUTTEXT(' EST ENTIEREMENT LIVREE ');
 OUTTEXT ('*****');
 END ;
 END ;
 END ;
END ;

```

```

IF B1.DAT > 0 THEN
BEGIN
 REAL TPS ;
 TPS := TIME - B1.DAT ;
 OUTTEXT (' TEMPS PASSE DANS') ;
 OUTTEXT (' L ENTREPRISE') ;
 ECRI (TPS) ;
 ST.AJOU (TPS) ;
END ;
OUTIMAGE ; OUTIMAGE ;
ADM.SORTIR(B1) ;
END
ELSE BEGIN $ LIVRAISON PARTIELLE $
 INSPECT B DO
 B1 :- NEW BC (NO,NLL,DAT,TYPE,NL,NLL) ;
 COMMANDES.MODIFIERCOM(B,B.NLL) ;
 B :- COMMANDES.SUIVANT (B) ;
END ;
ORDRES.ENTRER (B1) ;
PHY.ENTRER (B1) ;
END ;
NLAP := NLA - NLLT ; $ NB LIGNES EN ATTENTE $
HOLD (0.25) ;
ORD.LIBERER ;
END ;
END ; $ CDECISION $
INTEGER PROCEDURE NUMEROTATION ;
BEGIN
 ORDRE := ORDRE + 1 ;
 NUMEROTATION := ORDRE
END ; $ NUMEROTATION $
INTEGER PROCEDURE NUMERO ;
BEGIN
 NUMAPP := NUMAPP + 1 ;
 NUMERO := NUMAPP
END ; $ NUMERO $
INTEGER LIMCE,LIMCT,NLA,NLAP,NLREST,
 ORDRE , I , NUMAPP ;
REF (TABHISTO) NCE,LCE,LCT,ILCT ;
REF (CCOMMANDES) COMMANDES ;
REF (TFILE) COURRIER,ORDRES,BONS1,BONS2 ;
REF (TREGION) ECR,TEL,ADM,PHY ;
REF (CALRAMINF) ORD ;
REF (CALRC) X,Z,P ;
REF (CALRCPP) Y ;
REF (HISTOGRAMME) ST,HNCE,HLCT,HLCE ;
REF (ACTMONITOR) ACTIVATION1 , ACTIVATION2 , ACTIVATION3,
 ACTIVATION4 , ACTIVATIONS , BOITE ;
REF (ECHEANCIER) ECHE ;
LIMCE := 10 ;
LIMCT := 5 ;
ORDRE := 0 ;
NUMAPP := 0 ;
ECHE :- NEW ECHEANCIER (24,0) ;
ECR :- NEW TREGION ('ECRITES') ;
TEL :- NEW TREGION ('TELEPHONEES') ;
ADM :- NEW TREGION ('ADMINISTRATIF') ;
PHY :- NEW TREGION ('PHYSIQUE') ;

```

```

COMMANDES :- NEW CCOMMANDES ('COMMANDES',TRUE,TRUE,24,20) ;
COURRIER :- NEW TFILE ('COURRIER',TRUE,TRUE,24,8,50) ;
ORDRES :- NEW TFILE ('ORDRES',TRUE,TRUE,24,20) ;
BONS1 :- NEW TFILE ('BONS1',TRUE,TRUE,24,20) ;
BONS2 :- NEW TFILE ('BONS2',FALSE,FALSE,24,20) ;
ORD :- NEW CALRAMINF ('ORDINATEUR',FALSE,TRUE,0,0) ;
X :- NEW CALRC ('X',TRUE,TRUE,24,0) ;
Y :- NEW CALRCPP ('Y',TRUE,TRUE,24,0) ;
Z :- NEW CALRC ('Z',TRUE,TRUE,24,0) ;
P :- NEW CALRC ('P',TRUE,FALSE,24,0) ;
ACTIVATIONS :- NEW ACTMONITOR ('GENERATEUR',3) ;
ACTIVATION1 :- NEW ACTMONITOR ('ENREGISTREMENT',3) ;
ACTIVATION2 :- NEW ACTMONITOR ('DECISION',3) ;
ACTIVATION3 :- NEW ACTMONITOR ('VIDEUR1',3) ;
ACTIVATION4 :- NEW ACTMONITOR ('VIDEUR2',3) ;
BOITE :- NEW ACTMONITOR ('BOITE',1) ;
ECHE.MODIFECH (ACTIVATION1) ;
ECHE.MODIFECH (ACTIVATION2) ;
ECHE.MODIFECH (ACTIVATION3) ;
ECHE.MODIFECH (ACTIVATION4) ;
ECHE.MODIFECH (ACTIVATIONS) ;
NCE :- NEW TABHISTO (21,1,1) ;
LCE :- NEW TABHISTO (33,1,1) ;
LCT :- NEW TABHISTO (187,1) ;
HNCE :- NEW HISTOGRAMME ('NB COM ECR/J',21,1,1,1,30) ;
HLCE :- NEW HISTOGRAMME ('LG COM ECR',33,1,1,1,30) ;
HLCT :- NEW HISTOGRAMME ('LG COM TEL',18,1,1,1,30) ;
ILCT :- NEW TABHISTO (5,1,1) ;
FOR I := 1 STEP 1 UNTIL LIMCT DO
BEGIN
 REF (BC) B ;
 INTEGER NBLR ;
 NBLR := GENHISTO (ILCT) ;
 B :- NEW BC (NUMEROTATION,NBLR,TIME,"A",NBLR,NBLR) ;
 ADM.ENTRER (B) ;
COMMANDES.ENTRER (B) ;
END ;
FOR I := 1 STEP 1 UNTIL LIMCE DO
BEGIN
 REF (BC) B ;
 INTEGER NBLR ;
 NBLR := IMAX (GENHISTO (LCE) // 4 , 1) ;
 B :- NEW BC (NUMEROTATION,NBLR,TIME,"A",NBLR,NBLR) ;
 ADM.ENTRER(B) ;
COMMANDES.ENTRER(B) ;
END ;
NLAP := COMMANDES.VOLUME ;
NLREST := NLAP ;
ACTIVATE NEW GENERATEUR ('GENERATEUR',1,0) QUA PROCESS ;
ACTIVATE NEW GENETEL ('GENETEL',1,0) QUA PROCESS ;
ACTIVATE NEW ENREGISTREMENT ('ENREGISTREMENT',1,0) QUA PROCESS ;
ACTIVATE NEW SAISIE ('SAISIE',1,0) QUA PROCESS ;
ACTIVATE NEW DECISION ('DECISION',1,0) QUA PROCESS ;
ACTIVATE NEW PREPARATION ('PREPARATION',1,0) QUA PROCESS ;
ACTIVATE NEW VIDEUR ('VIDEUR1',1,0,BONS1,ACTIVATION3) QUA PROCESS ;
ACTIVATE NEW VIDEUR ('VIDEUR2',1,0,BONS2,ACTIVATION4) QUA PROCESS ;

```

```

ST :- NEW HISTOGRAMME ('SYSTEME',15,1,24,1,15) ;
SIMUL (120,FALSE,FALSE,FALSE,FALSE) ;
HISTOREGION (ADM,10,1,24,1,20) ;
PRECISEREGION (ADM) ;
PRECISEREGION (PHY) ;
PRECISERESSOURCE (P) ;
PRECISERESSOURCE (Z) ;
SIMUL (1440,TRUE,TRUE,TRUE,TRUE) ;
END $ BLOC MAESTRO $
END $ PROGRAMME $ #
!EOD
0 10 0 0 12 0 0 14 0 0 15 0**
0 8 0 0 12 0 0 14 0 0 18 0**
0 15 30 0 17 0**
0 8 0 0 12 0 0 14 0 0 18 0**
0 10 0*
0 15 0*
0 21 0*
0 21 0*
0 9 0*
4 2 2 4 8 7 3 6 9 1 3 2 5 2 2 1 1 1 0 2 1
421 210 135 139 102 96 83 67 46 47 32 31 31 14 16 9
11 12 6 3 5 4 8 1 6 4 4 6 2 1 1 1 2
171 49 32 19 14 11 15 5 2 4 4 2 5 1 5 2 0 1
97 20 4 4 3
!EOD
EOF HIT

```

NOM DE L'ETUDIANT : Mademoiselle THAUREL Michèle

NATURE DE LA THESE : Doctorat de 3ème cycle : INFORMATIQUE

VU, APPROUVE

et PERMIS D'IMPRIMER

NANCY LE 30.MAI/1980 05990

LE PRESIDENT DE L'UNIVERSITE DE NANCY I



M. BOULANCE