

# HUITIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

421



NICE du 1<sup>er</sup> au 5 JUIN 1981

ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR)

Jalal ALMHANA\* - Claude André BOZZO\*\*  
Jean PAILLE\*

## RESUME

## SUMMARY

Les travaux présentés sont relatifs à la recherche d'une méthodologie de représentation et d'analyse des algorithmes intervenant dans les processus "temps réel". Cette représentation est basée sur les réseaux de PETRI et les réseaux P.E.R.T. dont les propriétés respectives pour cette nature d'applications sont étudiées et comparées. Les notions de macro et de micro-réseaux permettent de passer de la représentation d'un algorithme sous forme de réseau à la structure de traitement correspondante. Cette structure est par nature de type "multi-opérateurs" et "reproduit" le graphe associé à l'algorithme. Elle comporte donc plusieurs organes de mémoire, plusieurs processeurs et des dispositifs d'interconnexion banalisés permettant d'élaborer des architectures de type "Data Flow" (AFD : Architecture à Flux de Données) qui sont les mieux adaptées aux problèmes particuliers qui se posent en Automatique, Estimation, Détection, Traitement du Signal, etc...

On présente les résultats de l'analyse effectuée dans le cas du filtrage optimal (représentation linéaire) qui fait intervenir une équation matricielle de RICCATI. L'étude et le développement d'un opérateur de RICCATI Rapide (ORR) devrait permettre d'appliquer aux signaux "temporels" réels les résultats développés dans le domaine des représentations Markoviennes depuis une dizaine d'années.

L'analyse de l'algorithme de base et de un ou plusieurs algorithmes de résolutions associés par la méthode des réseaux devrait conduire à mettre en évidence les parallélismes "spatio-temporels" inhérents aux algorithmes et les aspects séquentiels ou combinatoires de certaines macro-opérations. Il est donc possible de déterminer une structure de traitement adaptée à l'identification d'un "Chemin Critique" pour l'algorithme considéré (minimisation du temps de traitement global).

The work described in this paper deals with the development of a methodology for representing and analysing the algorithms involved in "real-time" processes. Such a representation is based on PETRI and P.E.R.T networks whose respective properties for this type of application are analysed and compared. Macro and micro-network concepts make it possible to change from the representation of an algorithm as a network to the corresponding processing structure. By nature this structure belongs to the "multi-operator" type and it "reproduces" the graph associated with the algorithm. It therefore comprises several memory units several processors and multi-purpose interconnection devices which make it possible to elaborate architectures of the "Data Flow" type that are the most suitable for particular problems encountered in automatics, computation, detection, signal processing, etc...

This paper gives the results of the analysis made in the case of optimal filtering (linear representation) involving a RICCATI matrix equation. The study and development of a fast RICCATI operator (ORR) should make it possible to apply to actual "time" signal the results which have been developed in the field of markovian representations for ten years or so.

The analysis of basic algorithms together with one or more algorithms of associated resolutions by means of the network method should result in showing up the space-time parallelisms inherent in the algorithms and the sequential or combinative aspects of some macro-operations. It is therefore possible to determine a processing structure suitable for the identification of a "critical path" for the algorithm under consideration (total processing time minimization).

\* Laboratoire de Traitement du Signal Numérique  
ENS P - Rue Henri Poincaré - 13397 MARSEILLE CEDEX

\*\* Centre d'Analyse de Systèmes, de Programmation et  
de Calcul (CAPCA) Direction des Constructions et  
Armes Navales de TOULON - TOULON NAVAL 83000 -



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHERS ET DE RESEAUX DE PETRI

APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR)

## 1. - INTRODUCTION

### 1.1 - Réseaux de PETRI et structures de traitement de type "flux de Données"

Les premiers travaux concernant la synthèse des systèmes séquentiels sont dus à D.A. HUFFMAN. On les classe en général en deux groupes : les travaux relatifs aux machines de MOORE [A2], où les sorties sont indépendantes vis à vis des entrées et les travaux relatifs aux machines de MEALY [A3], où les sorties ne sont pas indépendantes vis à vis des entrées.

De nombreux développements ont suivi les travaux de HUFFMAN. Ils ont porté sur les différentes phases de sa méthode : écriture du cahier des charges [A4] [A5] [A6], minimisation de la table des phases primitives [A7] [A8] [A9], codage [A10 à A15] [A1] et écriture des équations des variables internes et de sortie [A15 à A17].

Malheureusement, la méthode de HUFFMAN et ses développements ne s'appliquent simplement qu'à des machines de très petite taille. Ils ne s'adaptent plus à la complexité des problèmes posés relatifs aux systèmes séquentiels en général et aux systèmes asynchrones en particulier.

Dans un certain nombre de travaux récents, on utilise des graphes qui constituent un outil d'analyse puissant et bien adapté à la nature des problèmes envisagés et en particulier les représentations par réseaux de PETRI [A18] [A30 à A36]. Cette représentation permet en effet de décrire de manière claire et rigoureuse, le fonctionnement de l'automate concerné. Bien entendu, l'utilisation à cette fin des réseaux de PETRI, comme moyen de description et d'analyse des systèmes à évolution parallèle, ne constitue qu'une application du concept de base qui avait, en fait, été introduit dans le cadre de travaux de recherche associés aux problèmes de langages formels [A19 à A28]. De nombreux chercheurs ont travaillé sur la synthèse et la mise en oeuvre des réseaux de PETRI. On peut distinguer trois modes de réalisation : câblées [A41 à A44], micro-programmées [A45 à A49] [A36] et programmées [A50 à A54] [A43] (en y rattachant, bien entendu, certains dispositifs "matériels" comme les PLA : Programmable Logic Array). On notera cependant que ces réalisations se limitent aux réseaux de PETRI saufs et correctement conçus c'est-à-dire tels que tous les problèmes de conflits et de déterminisme soient résolus.

Parmi les méthodes d'analyse exploitant des graphes "ordonnés", on étudiera plus particulièrement la méthode de PETRI et la méthode P.E.R.T. (Program Evaluation Research Task) et les relations entre ces deux méthodes dans le cadre de l'application particulière envisagée.

Si l'on considère que tout algorithme peut être décrit par un réseau (de type PETRI ou P.E.R.T) cette représentation pouvant être effectuée au niveau d'un macro-réseau (base de macro-opérations élémentaires : produit et somme de matrices, recherche de racines d'une équation etc...) ou à celui d'un macro-réseau (Opérateurs arithmétiques et logiques, opérateur de stockage, opérateur de transfert etc...), il est possible de déduire du réseau la structure de l'ensemble de traitement. Le réseau permet en effet de connaître :

- La nature des opérateurs élémentaires à utiliser,

- L'architecte du dispositif de traitement de données associé à l'algorithme. Cette architecture se déduit de façon naturelle de celle du réseau,

- L'ordonnancement des opérations défini par le graphe et par la détermination du "chemin critique" (qui permet également de déterminer le temps opératoire global).

Les architectures considérées sont de type "Data Flow". Elles ne comportent pas de processeur central. Celui-ci est remplacé par une tranche de modules de traitement élémentaires (unité arithmétique ou logique, processeur d'entrée - sortie etc...) Elles ne comportent pas également de mémoire à accès aléatoire RAM. Celle-ci est remplacée par une tranche de modules de mémoires (Mémoires d'informations totales etc...) contenant des codes opératoires, des opérands, des adresses etc... L'horloge de synchronisation classique, le compteur de programme et les registres de stockage n'existent pas également. Un module de gestion et de commutation de paquets (un réseau d'arbitrage) permet de relier la sortie d'une tranche de modules de mémoires à l'unité de traitement appropriée du système.

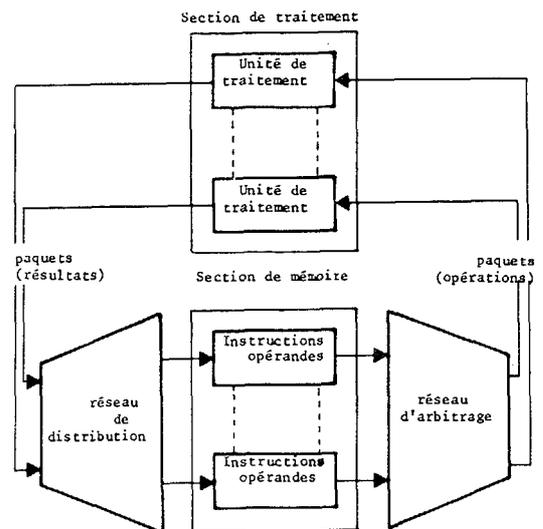


FIGURE 1 : ARCHITECTURE A FLUX DE DONNEES

Chaque unité de traitement envoie un paquet de (résultats du traitement, adresse, opérande) vers la Section de Mémoire (via un réseau de distribution).

### 1.2 - Filtrage Optimal - Etude d'un Opérateur de RICCATI Rapide

#### 1.2.1 - Généralités.

Les premiers travaux de WIENER portent sur l'utilisation d'une représentation externe des processus aléatoires stationnaires par une fonction de corrélation. KALMAN et BUCY [B1] ont exploité la représentation interne des processus Gaussiens-Markoviens sous forme d'équations différentielles (ou aux différences) stochastiques linéaires. Il existe, en fait, une infinité de réalisations Gaussiennes-Markoviennes d'un processus donné par sa fonction d'auto-corrélation (P. FAURRE) [B3].

Dans le cas discret, on peut écrire :

$$x_{k+1} = F(k) x_k + w_k$$

$$z_k = H(k) x_k + v_k$$

ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR).

La matrice de covariance du bruit blanc centré

$$\begin{bmatrix} w_k \\ v_k \end{bmatrix} \text{ étant :}$$

$$\begin{bmatrix} Q_K & S_K \\ S_K^T & R_K \end{bmatrix} \begin{cases} R \text{ et } F \text{ sont de dimension } (n \times n) \\ H \text{ et } S^T \text{ sont de dimension } (m \times n) \\ R \text{ est de dimension } (m \times m) \end{cases}$$

$x_k$  étant l'état à l'instant  $k$  (dimension  $n \times 1$ ),  
et  $z_k$  étant l'observation à l'instant  $k$  (dimension  $m \times 1$ ).

Dans la pratique les coefficients intervenant dans, le modèle mathématique ne sont pas connus et il faut résoudre l'un des deux problèmes suivants :

a) le doublet  $[F, H]$  (dynamique et observation) est connu, il s'agit alors d'estimer les matrices de covariance  $[Q, R, S]$  des bruits (de dynamique et de mesure).

b) le doublet  $[F, H]$  est lui-même inconnu, il faut alors identifier les cinq matrices  $F, H, Q, R, S$ .

Si le modèle est parfaitement connu, le filtrage optimal (au sens de KALMAN) donne la meilleure estimation de l'état.

$$\hat{x}_{k+1} = F(k) \hat{x}_k + F(k) K(k) \zeta_k$$

$\zeta(k)$  étant la suite innovation (bruit blanc) et  $K(k)$  le gain optimal.

1.22 - Estimation des covariances et du gain - Equation de RICCATI.

Que le modèle soit ou non parfaitement connu (du moins dans le cas a) quand  $F$  et  $H$  sont connus (voir ci-après) la détermination du gain de KALMAN conduit à résoudre une équation de RICCATI de la forme :

$$\begin{aligned} \Xi(k+1) &= F(k) [I - K(k) H(k)] \Xi(k) F^T(k) + Q(k) \\ (1) \quad &- S(k) [H(k) \Xi(k) H^T(k) + R(k)]^{-1} S^T(k) \\ &- F(k) K(k) S^T(k) - S(k) K^T(k) F^T(k) \end{aligned}$$

avec

$$(2) \quad K(k) = \Xi(k) H^T(k) [H(k) \Xi(k) H^T(k) + R(k)]^{-1}$$

dans le cas général.

Dans le cas particulier où  $S(k) = 0$  (ce qui veut dire que  $w$  et  $v$  ne sont pas corrélés), l'équation (1) se transforme en :

$$\begin{aligned} \Xi(k+1) &= F(k) \Xi(k) F^T(k) \\ &- F(k) \Xi(k) H^T(k) [H(k) \Xi(k) H^T(k) + R(k)]^{-1} H \Xi(k) F^T(k) + Q(k) \end{aligned}$$

Les équations, qui interviennent, dans les algorithmes de FAURRE, BELANGER et BOZZO ([B45] [B33] [B3] [B4]) et qui permettent d'identifier un processus Gaussien - Markovien de dynamique connue, ont également la même forme (à un changement de signe près).

Le problème de la conception d'un opérateur permettant de résoudre l'équation de RICCATI est donc tout particulièrement important et constitue un "point dur" pour l'application pratique des résultats du filtrage optimal.

Ce problème peut-être envisagé de la façon suivante :

- Détermination de la méthode mathématique (algorithme) permettant de résoudre l'équation (1) ou l'équation (3)
- synthèse d'un Opérateur de RICCATI Rapide (ORR).

On fait l'hypothèse que la méthode de résolution de l'équation de RICCATI a été étudiée et choisie par ailleurs. Il existe en effet de nombreux travaux dans ce domaine (cf en particulier les articles de LINDQUIST [B35] [B36] et [B37]) et on se reportera à la bibliographie analytique rédigée par A. GIULIERI et C. BOZZO [B47].

L'étude portera donc sur la réalisation de l'ORR. Cette résolution comportera les phases suivantes :

a) Analyse de l'algorithme de base (équation 1 ou équation 3) et de un ou plusieurs algorithmes de résolution (nous choisirons ici, par exemple, la méthode de LINDQUIST [B35] à [B37], la méthode quasi-linéarisation [B34] [B44] par la méthode des réseaux de PETRI. Cette analyse sera effectuée :

- Au niveau des opérations élémentaires (multiplications, décalages, etc...),
- au niveau des macro-opérations (calcul de covariances, extra-polations, etc...),

de façon à mettre en évidence les parallélismes spatio-temporels inhérents à l'algorithme et les aspects séquentiels ou combinatoires de certaines "macro-opérations".

Il est évident que certaines représentations (forme filtre, forme compagne, forme canonique de SALUT etc...) sont associées à la notion de représentation minimale et conduisent donc à un nombre minimal de coefficients et donc d'opérations élémentaires (additions ou multiplications)

Nous ferons donc l'hypothèse que la structure minimale "optimale" a été introduite et que l'analyse comparative entre les divers algorithmes est effectuée par le même doublet [0,4]. et sur la même forme canonique du système.

b) Choix d'un algorithme et d'une structure d'unité de traitement.

Ces deux aspects ne peuvent être séparés, la méthodologie adoptée devant être de même nature que celle déjà suivie, par exemple, dans le calcul de la Transformée de Fourier Discrète à l'aide de l'algorithme dit de Transformée de Fourier Rapide (calculs en papillon par exemple).

La structure de l'unité de traitement devra permettre d'obtenir au moindre coût un temps de traitement minimal (au sens du "transfert" global de l'opérateur).



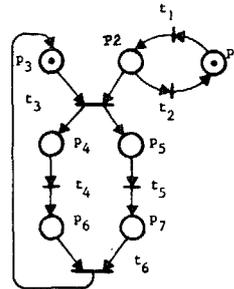
ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR).

2. - RAPPEL SUR LES RESEAUX DE PETRI

2.1 - Réseaux de PETRI ordinaires [21] [19]

2.11 - Définition :

Un réseau de PETRI, est un quadruplet  $R = (P, T, \alpha, \beta)$  tel que :  
 $P =$  soit un ensemble d'objets appelés places,  $P \neq \emptyset$   
 $T =$  ensemble d'objets appelés transitions,  $T \neq \emptyset$  et  $P \cap T = \emptyset$   
 $\alpha \subseteq P \times T =$  relation d'incidence avant  
 $\beta \subseteq T \times P =$  relation d'incidence arrière



$$M = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$t_1$  est validé par  $M$ .

2.12 - Représentation :

On représente un réseau de PETRI par un graphe biparti orienté. Les noeuds du graphe sont les places et les transitions qui sont représentées respectivement par des cercles et des barres (Fig.2.). Il existe un arc de la place  $P_r$  à la transition  $t_i$  si et seulement si  $(P_r, t_i) \in \alpha$ . De même il existe un arc de la transition  $t_i$  à la place  $P_s$  si et seulement si  $(t_i, P_s) \in \beta$ .

2.13 - Conventions de notations

On adopte les conventions suivantes :

- a)  $t \in T, t^* = \{p \in P \mid (p, t) \in \alpha\}$   
 $t^*$  est dit ensemble des places d'entrée de  $t$
- b)  $t \in T, t^* = \{p \in P \mid (t, p) \in \beta\}$   
 $t^*$  est dit ensemble des places de sortie de  $t$
- c)  $P \in P, P^* = \{t \in T \mid (t, P) \in \beta\}$   
 $P^*$  est dit ensemble des transitions d'entrée de  $P$
- d)  $P \in P, P^* = \{t \in T \mid (P, t) \in \alpha\}$   
 $P^*$  est dit ensemble des transitions de sortie de  $P$ .

2.2 - Propriétés d'un réseau de PETRI.

2.21 - Marquage : Un marquage  $M$  d'un réseau de PETRI  $R = (P, T, \alpha, \beta)$  est une application de  $P$  dans  $\mathbb{N}$ ,  $\mathbb{N}$  étant l'ensemble des entiers naturels.  
 Si  $n$  est le nombre des places de  $R$ , ( $|P| = n$ ), un marquage  $M$  de  $R$  peut-être défini par la donnée d'un vecteur de  $\mathbb{N}^n$ . On note :

$$M = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \quad \text{et } m_i = M(P_i)$$

Sur le graphe associé à  $R$ , on peut représenter un marquage  $M$  par une distribution, à l'intérieur des places, d'objets appelés marques. Une marque est représentée par un point. Chaque place  $P_i$  aura donc  $m_i$  marques (figure 2).

FIGURE 2 - EXEMPLE DE RESEAU DE PETRI

- Une transition  $t$  de  $T$  est dite tirable ou validée par le marquage  $M$  si et seulement si :  $\forall P \in t^*, \text{ on a } M(P) \geq 1$ . Autrement dit si et seulement si toute place d'entrée  $t$  contient au moins une marque.

Opération sur les marquages

Soit  $M$  l'ensemble des marquages de  $R$ , et de  $t$  une transition de  $R$ .  
 On définit la mise à feu de  $t$  ( $MAF_t$ ) comme étant l'opération qui, à un marquage  $M_i \in M$ , fait correspondre le marquage  $M_j$  tel que :

$$\forall p \in P, M_j(p) = \begin{cases} M_i(p) - 1 & \text{si } p \in t^* - (t^* - t^*) \\ M_i(p) & \text{si } p \in t^* - (t^* - t^*) \\ M_i(p) & \text{autrement} \end{cases}$$

On note  $MAF_t(M_i) = M_j$ . L'opération mise à feu est seulement définie pour les marquages  $M$  qui valident  $t$ .  
 Sur la représentation graphique associée à  $R$  avec un marquage  $M$ , l'opération  $MAF_t$  consiste à enlever une marque de toute place de sortie de  $t$ . Cette opération n'est possible que si chaque place d'entrée de  $t$  contient au moins une marque.

3. - REPRESENTATION DES ALGORITHMES PAR DES RESEAUX DE PETRI.

3.1 - Modèle de représentation.

Pour la représentation de nos algorithmes par des réseaux de PETRI, nous nous sommes inspirés d'un modèle décrit en [A56], appelé RdPI (réseaux de PETRI Interprétés), qui est fondé essentiellement sur des réseaux de PETRI temporisés. Dans ce modèle, le RdP,  $R$  décrit le schéma de contrôle de l'algorithme,

les applications  $\varphi$  et  $\psi$  définissent respectivement les actions associées aux places et les tests effectués (transitions), et les applications  $\mu$  et  $\nu$  traduisent le comportement dynamique que doit avoir le système dans ses interactions avec l'environnement extérieur. Tous ces éléments constituent un moyen important pour la description et la conception des systèmes parallèles synchrones. Cependant, un tel modèle n'est pas parfaitement adapté pour décrire une architecture de type "Data Flow" (Fig 1) qui nous intéresse pour la réalisation de l'ORR. Ceci nous a conduit à redéfinir le modèle précédent (RdPI) de la façon suivante :

3.11 - Réseaux de PETRI Temporisés (RdPT)

a) Définition : Un RdPT est défini par la donnée :

- d'un réseau  $R = (P, T, \alpha, \beta)$ ,
- d'une suite croissante de réels  $U = \{u_i\}_{i \in \mathbb{N}}$

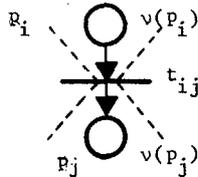
telle qu'à la mise à feu de chaque transition corresponde un élément de  $U$ ,

- d'un ensemble de réels  $\mathcal{E} = \{u_j - u_i\} = \{\tau_{ij}\}$ ,  $j > i \quad i \in \mathbb{N}, j \in \mathbb{N}$

ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
 APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE DE L'OPERATEUR DE RICCATI RAPIDE (ORR).

- d'une application  $v$  de  $P$  dans  $\mathcal{E}$  telle que :  $v(p_k) \leq (\mu_j - \mu_i)$ .
- $\mu_j$  : instant de tir de la transition  $t_j \in P_k$
- $\mu_i$  : instant de tir de la transition  $t_i \in P_k$

Représentation :



b) Opération sur le marquage

Définition : une marque dans une place  $P_i$  est dite "disponible" si le temps écoulé au marquage de cette dernière  $\Delta u \geq v(P_i)$

Une transition  $t_i$  est mise à feu dans un marquage  $M$  si elle est validée dans le réseau  $R$  et que les places  $P_i$  contiennent des marques disponibles.

c) Instants de mise à feu de la transition.

Pour une séquence de mise à feu  $S$  réalisable à partir d'un marquage initial  $M_0$ , dans un RdPT sauf, tel que  $M_0 \xrightarrow{S} M$ , on peut définir les instants de tirs (mise à feu) des transitions par la formule de récurrence suivante :

$$I_{MAF}^-(t_i) = \max_{P_k \in t_i} [ I_M^-(P_k) + v(P_k) ] \quad (4)$$

$$I_M^-(P_j) = I_{MAF}^-(t_i) \quad | \quad t_i \in P_j \quad (5)$$

On suppose que les instants de marquages des places marquées par  $M_0$  sont égaux à zéro.

$$\forall P \in P, M(P) \in M_0 \implies I_M^-(P) = 0$$

Une transition peut-être tirée plusieurs fois dans une séquence de mise à feu telle que :

$$I_{MAF}^+(t_i) - I_{MAF}^-(t_i) \geq \max_{P_k \in t_i} (v(P_k))$$

Le signe + indique la prochaine mise à feu de  $t_i$   
 Le signe - indique la mise à feu précédente de  $t_i$

Nous avons considéré dans la formule (5), que les opérations de tirs des transitions par elles mêmes n'impliquent pas des retards sur le marquage des places  $t_i$ . Nous rappelons que les actions ont été affectées aux places. Dans le cas inverse, la formule (5) devient :

$$I_M^-(P_k) = I_{MAF}^-(t_i) \quad | \quad t_i \in P_k \quad + \quad \Delta \bar{\tau}$$

$\Delta \bar{\tau}$  est le retard dû à l'opération de tir.

3.12 - Réseaux de PETRI Temporisés Interprétés (RdPTI)

Un RdPTI est défini par la donnée :

- a) d'un système opératif  $(X, OP, c)$  tel que :

-  $X = \{x_1, x_2, \dots, x_u\}$  est un ensemble fini de variables prenant leurs valeurs respectivement dans les domaines  $D_1, D_2, \dots, D_u$ .

-  $OP = \{OP_1, OP_2, \dots, OP_v\}$  est un ensemble fini d'opérateurs définis comme des applications internes de  $D_1 \times D_2 \times \dots \times D_u$

- $C = \{C_1, C_2, \dots, C_r\}$  est un ensemble de conditions (prédicats) sur les variables de  $x$  :
- b) d'un RdPT  $(R, U, \mathcal{E}, v)$
- c) d'une application  $\varphi : P \rightarrow OP$
- d) d'une application  $\psi : T \rightarrow C$

Finalement, on peut dire que les modifications apportées au modèle décrit en [A56] sont caractérisées par l'utilisation d'un réseau de PETRI temporisé non synchronisé. D'autre part, l'application  $v$  dans le nouveau modèle représente la durée de l'indisponibilité d'une marque dans une place, autrement dit la durée de l'action associée à la place; en revanche l'application  $v$  dans le premier modèle représente l'instant à partir duquel la marque devient disponible.

Ceci n'est qu'une nuance et ne constitue pas une modification fondamentale. En effet, le second modèle est semblable au premier si on considère tous les événements dans RdPS comme étant un événement toujours présent.

3.2 - Réseaux réalisés : nous avons adopté le modèle RdPTI pour représenter les algorithmes.

Sur les réseaux correspondants on définit les places et les transitions comme suit :

3.2.1 - Places : On distingue trois types de places

a) Places qui représentent des opérateurs (multiplieurs, additionneur, etc...). Celles-ci sont temporisées ( $v(P) \neq 0$ ), c'est-à-dire que l'on attribue une valeur  $\tau_i$  qui représente le temps de calcul de l'opérateur concerné.

b) Places qui représentent des mémoires "logiques". On notera que le temps d'accès à une mémoire peut-être considéré, en général, comme fixe (temps de lecture-écriture) ou, du moins, borné, et qu'il est inclus dans le temps opératoire des opérateurs correspondants. Dans la représentation adoptée, les mémoires sont donc considérées comme des opérateurs de stockage "parfaits".

c) Places de synchronisation ou d'ordonnancement (cercles hachurés). Celles-ci ont été ajoutées pour régler les problèmes de conflit (partage des ressources) ou tout problème qui peut rendre le réseau non-vivant. Ces places peuvent-être temporisées ( $v(P) = 0$ ), ceci dépend du rôle qui leur a été associé.

3.2.2 - Transitions :

On trouve sur les transitions des conditions (prédicats), des tests concernant les résultats issus des opérateurs etc...

4 - RESEAUX P.E.R.T. (PROGRAM EVALUATION RESEARCH TASK)

4.1 - Rappels :

Cette méthode est utilisée normalement en recherche opérationnelle pour établir le graphe d'un programme de recherche ou de développement. Par le calcul du chemin critique, il est possible d'analyser le temps de réalisation d'un ensemble de travaux.

Un programme, ensemble d'opérations concourant à la réalisation d'un objectif quelconque, peut-être représenté par un graphe, appelé graphe de programme ou graphe d'ordonnancement. On peut représenter ce dernier de deux manières : dans une première configuration, les sommets du graphe appelés EVENEMENTS et notés  $E_i$ , peuvent s'interpréter comme marquant la réalisation d'objectifs partiels (montage d'un sous-ensemble, par exemple). Les arcs représentent alors les opérations. La durée d'une opération  $OP_i$  qui est une des opérations contribuant à la réalisation de



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR)

l'évènement  $E_j$  est supposée connue.

Dans une seconde configuration, on représente, non plus un évènement par un sommet mais par une opération déterminée, et on utilise un arc pour représenter l'ordre dans lequel deux opérations  $OP_i$  et  $OP_j$  doivent se succéder de telle sorte qu'il soit possible de donner le délai entre le début de  $OP_i$  et celui de  $OP_j$ . Dans cette représentation, on convient d'appeler "tâches" les opérations précédemment considérées et "contraintes" les arcs auxquels les valeurs de délais sont affectées.

#### 4.2 - Représentation des algorithmes avec la méthode P.E.R.T.

Normalement, l'établissement du graphe ou du réseau correspondant à un programme quelconque nécessite un cahier des charges bien formulé et une analyse approfondie du problème. Du fait que les réseaux de PETRI étant formés à partir du cahier des charges après une analyse détaillée de ce dernier et compte tenu de l'existence pour les trois algorithmes étudiés (KALMAN, quasi-linéarisation et (LINDQUIST) à l'aide de réseaux de PETRI, il semble intéressant de passer directement du réseau de PETRI au réseau P.E.R.T. Pour cela on propose la méthode suivante :

a) - On élimine du réseau de PETRI toutes les places qui ne sont pas comprises dans l'ensemble défini par la relation suivante :

$$E = \{P \in \mathcal{P}, v(P) \in \mathcal{P} \mid v(P) \neq 0\}$$

$E$  est dit "ensemble des places temporisées du réseau" où  $v(P) \neq 0$

b) - On élimine du réseau de PETRI toutes les transitions sauf celles comprises dans l'ensemble  $\eta$ .

$$P \in E, \eta = \{t \in T \mid (t, p) \in \beta\}$$

c) - Les transitions restantes formeront les sommets ou les tâches du réseau P.E.R.T. En revanche, les places formeront les arcs ou les contraintes reliant les sommets ou les tâches.

Le temps  $T_i$  (temps associé à l'opération) attaché à la place dans le réseau de PETRI sera attribué à l'arc correspondant dans le réseau de P.E.R.T.

d) - Les arcs seront orientés sur le réseau P.E.R.T. dans le sens de l'évolution du marquage dans le réseau de PETRI.

Ainsi l'application de ces règles nous a conduit aux réseaux P.E.R.T. correspondant aux trois algorithmes étudiés. Ces réseaux sont représentés sur les graphes  $A_2$ ,  $B_2$  et  $C_2$ , respectivement. On précise que le délai entre deux tâches est supposé connu et que les durées des opérations ne sont pas aléatoires. Il est intéressant de noter que les places d'ordonnement sur les réseaux de PETRI apparaissent comme des contraintes supplémentaires sur les tâches dans les réseaux P.E.R.T. Celles-ci peuvent être ajoutées facilement, même après la construction du réseau, sans toutefois avoir besoin d'effectuer une nouvelle analyse du réseau (avantage de la représentation par tâches et contraintes dans les réseaux P.E.R.T.).

Bien entendu, lors du calcul du chemin critique, il faut tenir compte du nombre de marquages de places temporisées, autrement dit du nombre d'itérations demandé pour la réalisation du calcul. Pour déterminer ce dernier, on fait référence à l'article de A. GIULIERI ET C. BOZZO [B47].

#### 5. - APPLICATION A LA SYNTHESE D'UN OPERATEUR DE RICCATI RAPIDE.

On analysera à titre d'exemple, dans ce qui suit, le problème présenté dans le paragraphe 2. L'équation de RICCATI peut-être résolue :

a) à partir de l'algorithme de base (algorithme de KALMAN),

b) à partir d'un algorithme conduisant à la solution de façon itérative avec une approche indirecte :

- algorithme de quasi-linéarisation qui fait intervenir une équation de LIAPUNOV.
- algorithme de LINDQUIST.

On constate que, pour une même équation résolvant le problème théorique considéré (filtrage optimal - équation de RICCATI), les différents algorithmes considérés conduisent à des réseaux P.E.R.T. différents et donc à des chemins critiques différents. Si l'on compare ces chemins critiques, il existe un optimal au sens du temps de traitement et des ressources nécessaires (opérateurs, mémoires, etc...).

#### 5.1 - Méthode Directe - algorithme De KALMAN.

$$E + = \phi \Xi \phi^T - \phi \Xi H^T (H \Xi H^T + R)^{-1} H \Xi \phi^T + Q.$$

Cette équation correspond en fait aux trois équations récursives suivantes [ $\Xi^+ = \Xi(k+1)$  si  $\Xi = \Xi(k)$ ]

$$P_{k/k-1} = \phi P_{k-1/k-1} \phi^T + Q$$

$$P_{k/k} = P_{k/k-1} - K_k H P_{k/k-1}$$

$$K_k = P_{k/k-1} H^T [H P_{k/k-1} H^T + R]^{-1}$$

La solution asymptotique vérifie les relations :

$$P_{k+1/k} = P_{k/k-1} = \Xi \text{ et } P_{k+1/k+1} = P_{k/k}.$$

L'inconvénient de cette méthode provient, d'une part de l'inversion de la matrice :

$$[H P_{k/k-1} H^T + R] \text{ lorsque } m > 1, \text{ d'autre part du}$$

fait que les erreurs d'arrondi dans les calculateurs peuvent faire perdre le caractère symétrique défini positif de la matrice  $P$ .

Afin de garder la symétrie, on utilise parfois à chaque pas de calcul, une méthode qui consiste à faire la moyenne de termes :

$$P^* = \frac{1}{2} (P^* + P^{*T}) \text{ avec } P^* = P_k^{t-1}$$

Cette solution, qui améliore la précision sur  $P$  conduit à un nombre d'opérations arithmétiques beaucoup plus important.

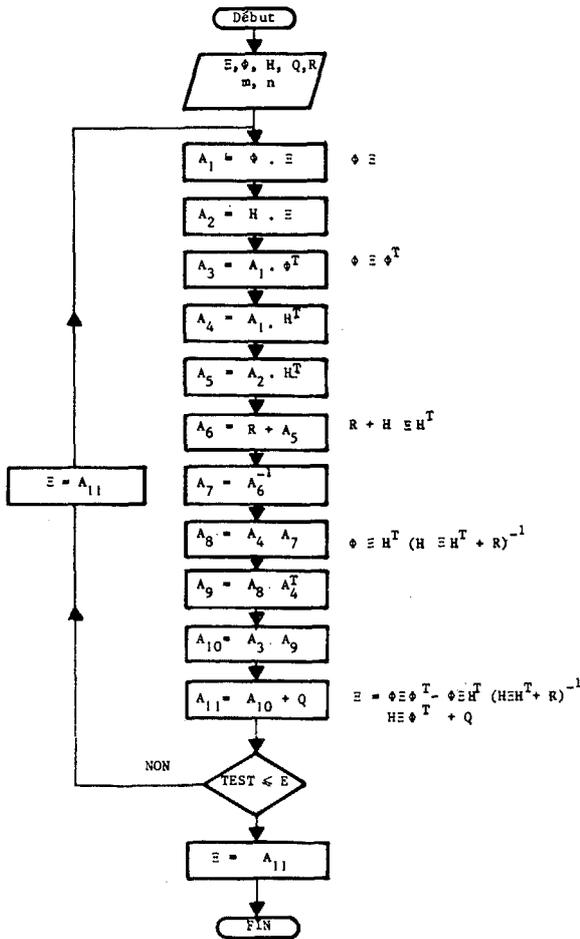
Les calculs de la propriété de la matrice  $P$  est conservée par la formulation de mise à jour de la variance de JOSEPH [B18, B19] :

$$P_{k+1/k+1} = (I - K_{k+1} H) P_{k+1/k} (I - K_{k+1} H)^T + K_{k+1} R K_{k+1}^T$$

Cette formule donne la valeur de la covariance en étant peu sensible à une erreur de gain, mais augmente énormément la taille des calculs.

On présente sommairement, dans ce qui suit, les graphes et tableaux correspondants à l'algorithme classique (organigramme I, tableau I, graphe A) les algorithmes de quasi-linéarisation et de LINDQUIST sont analysés de façon plus détaillée car il nous ont semblé plus caractéristiques vis à vis de la méthode proposée.

ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
 APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE DE L'OPERATEUR DE RICCATI RAPIDE (ORR)

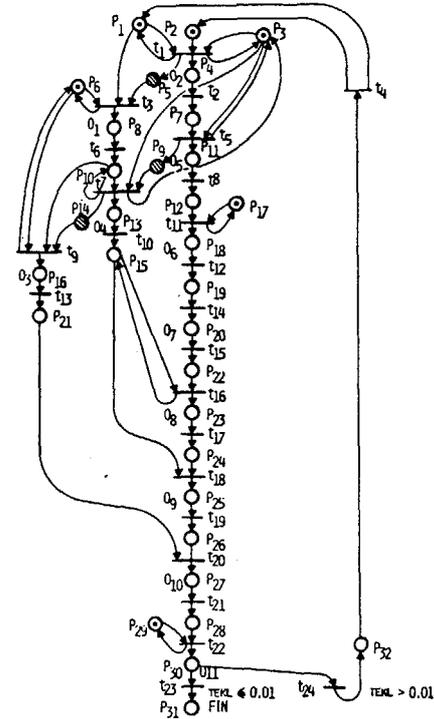


ORGANIGRAMME I.  
 Organigramme associé à la résolution de l'équation de RICCATI par la méthode traditionnelle (algorithme de KALMAN)

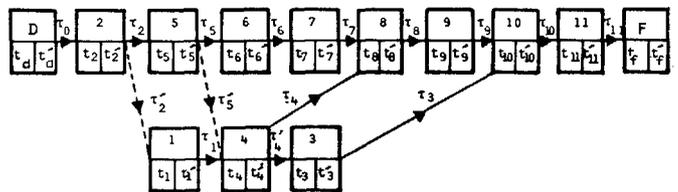
Dimensions des matrices :  $A_1 (n,n)$ ,  $A_2 (m,n)$ ,  
 $A_3 (n,n)$ ,  $A_4 (n,m)$ ,  $A_5 (m,m)$ ,  $A_6 (m,m)$ ,  $A_7 (m,m)$ ,  
 $A_8 (n,m)$ ,  $A_9 (n,n)$ ,  $A_{10} (n,n)$ ,  $A_{11} (n,n)$

Opérateurs et Places associées	Calcul effectué	Temps d'exécution	Nombre d'itérations	Conditions relatives aux opérateurs et transitions associées	Mémoires dans lesquelles les résultats sont chargés et placés associées	Modèle
$0_1 \rightarrow P_8$	$A_1 = \Phi \cdot \Xi$	$\tau_1$	k	$C_1 \rightarrow \tau_3$	$M_1 \rightarrow P_{10}$	$R + M_{16} \rightarrow P_{17}$
$0_2 \rightarrow P_4$	$A_2 = H \cdot \Xi$	$\tau_2$	k	$C_2 \rightarrow \tau_1$	$M_2 \rightarrow P_7$	$H + M_{15} \rightarrow P_3$
$0_3 \rightarrow P_{16}$	$A_3 = A_1 \cdot \Phi^T$	$\tau_3$	k	$C_3 \rightarrow \tau_9$	$M_3 \rightarrow P_{21}$	$\Xi + M_{14} \rightarrow P_1$
$0_4 \rightarrow P_{13}$	$A_4 = A_1 \cdot H^T$	$\tau_4$	k	$C_4 \rightarrow \tau_7$	$M_4 \rightarrow P_{15}$	$\Phi + M_{13} \rightarrow P_6$
$0_5 \rightarrow P_{11}$	$A_5 = A_2 \cdot H^T$	$\tau_5$	k	$C_5 \rightarrow \tau_5$	$M_5 \rightarrow P_{12}$	$Q + M_{12} \rightarrow P_{29}$
$0_6 \rightarrow P_{18}$	$A_6 = R + A_5$	$\tau_6$	k	$C_6 \rightarrow \tau_{11}$	$M_6 \rightarrow P_{19}$	
$0_7 \rightarrow P_{20}$	$A_7 = A_6^{-1}$	$\tau_7$	k	$C_7 \rightarrow \tau_{14}$	$M_7 \rightarrow P_{22}$	
$0_8 \rightarrow P_{23}$	$A_8 = A_4 \cdot A_7$	$\tau_8$	k	$C_8 \rightarrow \tau_{16}$	$M_8 \rightarrow P_{24}$	
$0_9 \rightarrow P_{25}$	$A_9 = A_8 \cdot A_4^T$	$\tau_9$	k	$C_9 \rightarrow \tau_{18}$	$M_9 \rightarrow P_{26}$	
$0_{10} \rightarrow P_{27}$	$A_{10} = A_3 \cdot A_9$	$\tau_{10}$	k	$C_{10} \rightarrow \tau_{20}$	$M_{10} \rightarrow P_{28}$	
$0_{11} \rightarrow P_{30}$	$A_{11} = A_{10} + Q$	$\tau_{11}$	k	$C_{11} \rightarrow \tau_{22}$	$M_{11} \rightarrow P_{32}$	

TABLEAU 1 : EQUATION DE RICCATI (méthode traditionnelle).  
 k: nombre d'itérations demandées pour la convergence



GRAPHE A1 : Réseau de PETRI associé à la méthode traditionnelle de résolution de l'équation de RICCATI.



GRAPHE A2 : Réseau P.E.R.T. de l'algorithme de la méthode traditionnelle (KALMAN).  
 Représentation par tâches et contraintes  
 $t_j$  : instant au plus tôt.  
 $t'_j$  : instant au plus tard.

5.2 - Algorithme de quasi-linéarisation.

On pose dans le cas discret :

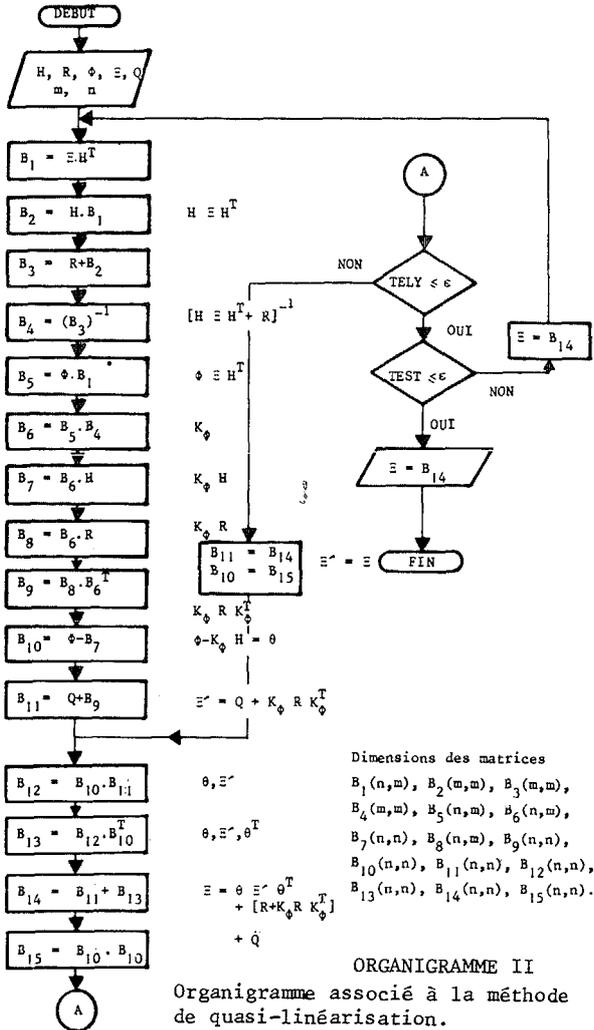
$$\begin{cases} \Theta_i = \Phi - K_\Phi(i) H & \text{avec } K_\Phi(i) = \Phi K(i) \\ P^* i - \Theta_i P^* i \Theta_i^T = K_\Phi(i) R K_\Phi^T(i) + Q & \text{avec } P^* \equiv \Xi \\ K_\Phi(i) = \Phi P^* i H^T (HP^* i H^T + R)^{-1} \end{cases}$$

Cet algorithme converge si  $\Phi - K_\Phi(0) H$  est stable. La deuxième équation est une équation de LIAPUNOV discrète que l'on résout de façon itérative.

La méthode de quasi-linéarisation est décrite par l'organigramme II, par le tableau 2 et par le graphe B.



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI  
 APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE DE L'OPERATEUR DE RICCATI RAPIDE (ORR)



Dimensions des matrices  
 $B_1(n,m), B_2(m,m), B_3(m,m),$   
 $B_4(m,m), B_5(n,m), B_6(n,m),$   
 $B_7(n,n), B_8(n,m), B_9(n,n),$   
 $B_{10}(n,n), B_{11}(n,n), B_{12}(n,n),$   
 $B_{13}(n,n), B_{14}(n,n), B_{15}(n,n).$

ORGANIGRAMME II

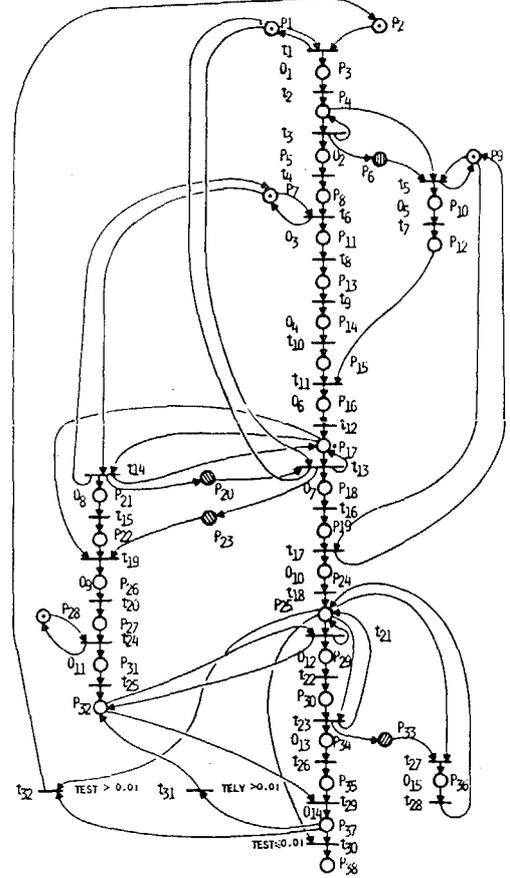
Organigramme associé à la méthode de quasi-linéarisation.

TELY: référence de convergence dans l'équation de LIAPUNOV.  
 TEST: référence de convergence dans la quasi-linéarisation.

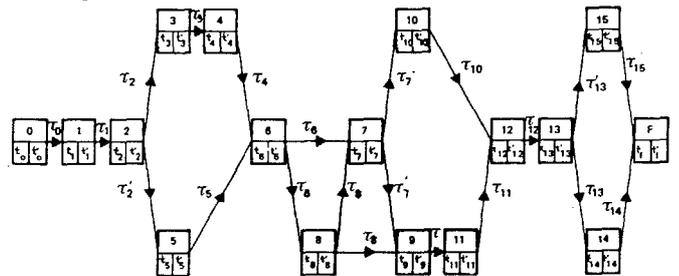
Opérateurs et places associées	Calcul effectué	Temps d'exécution	Nombre d'itérations	Conditions relatives aux opérateurs et transitions associées	Mémoires dans lesquelles les résultats sont chargés et places associées.	Modèle
$0_1 \rightarrow P_3$	$B_1 = \Xi \cdot H^T$	$\tau_1$	$k_1$	$C_1 \rightarrow \tau_1$	$M_1 \rightarrow P_4$	$Q \rightarrow H_{15}^T \cdot P_{28}$
$0_2 \rightarrow P_5$	$B_2 = H \cdot B_1$	$\tau_2$	$k_1$	$C_2 \rightarrow \tau_3$	$M_2 \rightarrow P_8$	$R \rightarrow H_{16}^T \cdot P_7$
$0_3 \rightarrow P_{11}$	$B_3 = R + B_2$	$\tau_3$	$k_1$	$C_3 \rightarrow \tau_6$	$M_3 \rightarrow P_{13}$	$H \rightarrow H_{17}^T \cdot P_1$
$0_4 \rightarrow P_{14}$	$B_4 = (B_3)^{-1}$	$\tau_4$	$k_1$	$C_4 \rightarrow \tau_9$	$M_4 \rightarrow P_{15}$	$\Xi \rightarrow H_{18}^T \cdot P_2$
$0_5 \rightarrow P_{10}$	$B_5 = \phi \cdot B_1$	$\tau_5$	$k_1$	$C_5 \rightarrow \tau_5$	$M_5 \rightarrow P_{12}$	$\phi \rightarrow H_{19}^T \cdot P_9$
$0_6 \rightarrow P_{16}$	$B_6 = B_5 \cdot B_4$	$\tau_6$	$k_1$	$C_6 \rightarrow \tau_{11}$	$M_6 \rightarrow P_{17}$	
$0_7 \rightarrow P_{18}$	$B_7 = B_6 \cdot H$	$\tau_7$	$k_1$	$C_7 \rightarrow \tau_{13}$	$M_7 \rightarrow P_{19}$	
$0_8 \rightarrow P_{21}$	$B_8 = B_6 \cdot R$	$\tau_8$	$k_1$	$C_8 \rightarrow \tau_{14}$	$M_8 \rightarrow P_{22}$	
$0_9 \rightarrow P_{26}$	$B_9 = B_8 \cdot B_5^T$	$\tau_9$	$k_1$	$C_9 \rightarrow \tau_{19}$	$M_9 \rightarrow P_{27}$	
$0_{10} \rightarrow P_{24}$	$B_{10} = \phi - B_7$	$\tau_{10}$	$k_1$	$C_{10} \rightarrow \tau_{17}$	$M_{10} \rightarrow P_{15}$	
$0_{11} \rightarrow P_{31}$	$B_{11} = Q + B_9$	$\tau_{11}$	$k_1$	$C_{11} \rightarrow \tau_{24}$	$M_{11} \rightarrow P_{32}$	
$0_{12} \rightarrow P_{29}$	$B_{12} = B_{10} \cdot B_{11}$	$\tau_{12}$	$k_1, k_2$	$C_{12} \rightarrow \tau_{21}$	$M_{12} \rightarrow P_{30}$	
$0_{13} \rightarrow P_{34}$	$B_{13} = B_{12} \cdot B_{10}^T$	$\tau_{13}$	$k_1, k_2$	$C_{13} \rightarrow \tau_{23}$	$M_{13} \rightarrow P_{35}$	
$0_{14} \rightarrow P_{37}$	$B_{14} = B_{11} + B_{13}$	$\tau_{14}$	$k_1, k_2$	$C_{14} \rightarrow \tau_{29}$	$M_{14} \rightarrow P_{32}$	
$0_{15} \rightarrow P_{36}$	$B_{15} = B_{10} \cdot B_{10}$	$\tau_{15}$	$k_1, k_2$	$C_{15} \rightarrow \tau_{27}$	$M_{15} \rightarrow P_{25}$	

TABEAU 2 : METHODE DE QUASI-LINEARISATION

$k_2$ : nombre d'itérations pour résoudre l'équation de LIAPUNOV.  
 $k_1$ : nombre d'itérations demandées pour la convergence



Graphe B1 Réseau de PETRI associé à l'algorithme de quasi-linéarisation.



Graphe B2 Réseau P.E.R.T. associé à l'algorithme de quasi-linéarisation. Représentation par tâches et contraintes.

5.3 - Algorithme de LINDQUIST.

Cet algorithme comporte trois équations principales

$$(1) S_{n+1} = S_n + S_n^* (R_n^*)^{-1} S_n^{*T} H^T$$

$$(2) S_{n+1}^* = \phi [S_n^* + S_n (H \cdot S_n + R) \cdot H \cdot S_n^*]$$

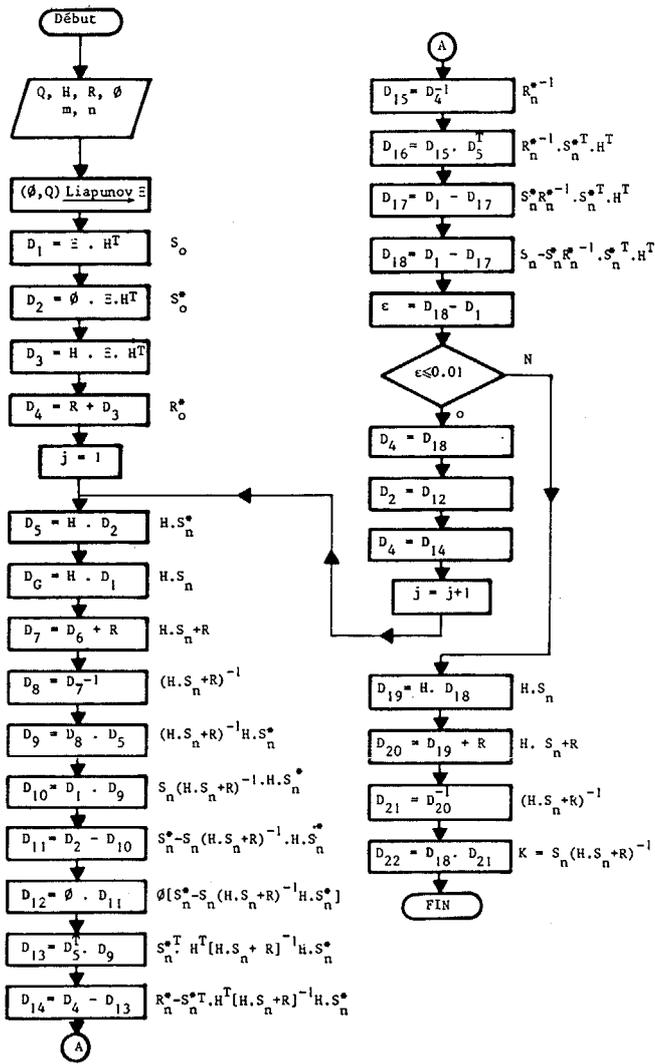
$$(3) R_{n+1}^* = R_n^* - S_n^{*T} \cdot H^T (H \cdot S_n + R)^{-1} \cdot H \cdot S_n^*$$

avec  $S_0 = \Xi \cdot H^T$   
 $S_0^* = \phi \cdot \Xi \cdot H^T$   
 $R_0^* = H \cdot \Xi \cdot H^T + R$

$\Xi$  est la solution de l'équation de LIAPUNOV associée au filtre. Le gain de KALMAN :  $K = S_n (H \cdot S_n + R)^{-1}$

Cet algorithme est décrit par l'organigramme III auquel sont associés, le tableau 3 et les graphes C1 et C2

ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHES ET DE RESEAUX DE PETRI APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE DE L'OPERATEUR DE RICCATI RAPIDE (ORR).



ORGANIGRAMME III  
- Algorithme de LINDQUIST.  
Dimensions des matrices.  
 $D_1(n,m), D_2(n,m), D_3(m,m), D_4(m,m), D_5(m,m),$   
 $D_6(m,m), D_7(m,m), D_8(m,m), D_9(m,m), D_{10}(n,m),$   
 $D_{11}(n,m), D_{12}(n,m), D_{13}(m,m), D_{14}(m,m), D_{15}(m,m),$   
 $D_{16}(m,m), D_{17}(n,m), D_{18}(n,m), D_{19}(m,m), D_{20}(m,m),$   
 $D_{21}(m,m), D_{22}(n,m).$

6 - ETUDE DES PROPRIETES DES GRAPHES ASSOCIES AUX ALGORITHMES. Notion de chemin critique - Exemple de l'algorithme de LINDQUIST.

Pour pouvoir exploiter les graphes associés aux algorithmes étudiés précédemment, nous avons été amenés à développer un logiciel qui comporte plusieurs programmes principaux :

6.1 - PROGRAMME DE DESCRIPTION D'UN RdP.

Ce programme permet la description logicielle des différents types de réseaux de PETRI, ordinaires, synchronisés, temporisés etc...Ceci est réalisé à partir de la matrice d'adjacence ou matrice d'incidence "sommets-sommets" sous la forme condensée qui permet, d'une part de décrire parfaitement la structure d'un RdP quel que soit son type et, d'autre part, de minimiser la place mémoire occupée par ce réseau.

Opérateurs et Places associées.	Calcul effectué	Nombre d'itérations	Conditions relatives aux opérateurs et transitions associées.	Temps d'exécution	Mémoires dans lesquelles les résultats sont chargés et placés associées
$O_{23} \rightarrow P_3$		1	$C_{23} \rightarrow t_1$	$\tau_{23}$	$M_{23} \rightarrow P_4$
$O_1 \rightarrow P_6$	$D_1 = \Xi.H^T$	1	$C_1 \rightarrow t_3$	$\tau_1$	$M_1 \rightarrow P_{18}, P_7$
$O_2 \rightarrow P_{10}$	$D_2 = \Phi.\Xi.H^T$	1	$C_2 \rightarrow t_6$	$\tau_2$	$M_2 \rightarrow P_{15}, P_{35}$
$O_3 \rightarrow P_8$	$D_3 = H.\Xi.H^T$	1	$C_3 \rightarrow t_5$	$\tau_3$	$M_3 \rightarrow P_{13}$
$O_4 \rightarrow P_{16}$	$D_4 = R+H.\Xi.H^T$	1	$C_4 \rightarrow t_9$	$\tau_4$	$M_4 \rightarrow P_{17}, P_{46}$
$O_5 \rightarrow P_{36}$	$D_5 = H.D_2$	j	$C_5 \rightarrow t_{27}$	$\tau_5$	$M_5 \rightarrow P_{37}$
$O_6 \rightarrow P_{19}$	$D_6 = H.D_1$	j	$C_6 \rightarrow t_{13}$	$\tau_6$	$M_6 \rightarrow P_{20}$
$O_7 \rightarrow P_{21}$	$D_7 = D_6 + R$	j	$C_7 \rightarrow t_{15}$	$\tau_7$	$M_7 \rightarrow P_{22}$
$O_8 \rightarrow P_{23}$	$D_8 = D_7^{-1}$	j	$C_8 \rightarrow t_{17}$	$\tau_8$	$M_8 \rightarrow P_{24}$
$O_9 \rightarrow P_{26}$	$D_9 = D_8.D_5$	j	$C_9 \rightarrow t_{19}$	$\tau_9$	$M_9 \rightarrow P_{65}$
$O_{10} \rightarrow P_{27}$	$D_{10} = D_1.D_9$	j	$C_{10} \rightarrow t_{21}$	$\tau_{10}$	$M_{10} \rightarrow P_{28}$
$O_{11} \rightarrow P_{30}$	$D_{11} = D_2.D_{10}$	j	$C_{11} \rightarrow t_{23}$	$\tau_{11}$	$M_{11} \rightarrow P_{31}$
$O_{12} \rightarrow P_{32}$	$D_{12} = \Phi.D_{11}$	j	$C_{12} \rightarrow t_{25}$	$\tau_{12}$	$M_{12} \rightarrow P_{33}$
$O_{13} \rightarrow P_{39}$	$D_{13} = D_3.D_9$	j	$C_{13} \rightarrow t_{29}$	$\tau_{13}$	$M_{13} \rightarrow P_{40}$
$O_{14} \rightarrow P_{41}$	$D_{14} = D_4.D_{13}$	j	$C_{14} \rightarrow t_{31}$	$\tau_{14}$	$M_{14} \rightarrow P_{42}$
$O_{15} \rightarrow P_{47}$	$D_{15} = D_4^{-1}$	j	$C_{15} \rightarrow t_{34}$	$\tau_{15}$	$M_{15} \rightarrow P_{48}$
$O_{16} \rightarrow P_{49}$	$D_{16} = D_{15}.D_5$	j	$C_{16} \rightarrow t_{36}$	$\tau_{16}$	$M_{16} \rightarrow P_{66}$
$O_{17} \rightarrow P_{50}$	$D_{17} = D_2.D_{16}$	j	$C_{17} \rightarrow t_{38}$	$\tau_{17}$	$M_{17} \rightarrow P_{52}$
$O_{18} \rightarrow P_{53}$	$D_{18} = D_1.D_{17}$	j	$C_{18} \rightarrow t_{39}$	$\tau_{18}$	$M_{18} \rightarrow P_{54}, P_{57}$
$O_{19} \rightarrow P_{58}$	$D_{19} = H.D_{18}$	1	$C_{19} \rightarrow t_{42}$	$\tau_{19}$	$M_{19} \rightarrow P_{59}$
$O_{20} \rightarrow P_{60}$	$D_{20} = D_{19} + R$	1	$C_{20} \rightarrow t_{44}$	$\tau_{20}$	$M_{20} \rightarrow P_{61}$
$O_{21} \rightarrow P_{62}$	$D_{21} = D_{20}^{-1}$	1	$C_{21} \rightarrow t_{46}$	$\tau_{21}$	$M_{21} \rightarrow P_{63}$
$O_{22} \rightarrow P_{64}$	$D_{22} = D_{18}.D_{21}$	1	$C_{22} \rightarrow t_{48}$	$\tau_{22}$	$M_{22} \rightarrow P_{67}$
$O_{24} \rightarrow P_{55}$	$\epsilon = D_{18}^{-1}.D_{18}$	j	$C_{23} \rightarrow t_{40}$	$\tau_{23}$	$M_{23} \rightarrow P_{56}$

TABLEAU 3  
ALGORITHME de LINDQUIST.

6.2 - PROGRAMMES DE SIMULATION.

Trois programmes de simulation ont été réalisés :

- Programme S1 : Dans ce programme on vérifie que le réseau étudié est sauf et vivant.

On vérifie que le réseau étudié est sauf et vivant. Ceci est réalisé d'une manière progressive sur des réseaux, non temporisés, temporisés, interprétés. On cherche également le vecteur de marquage atteint  $M_f$  à partir d'un marquage initial  $M_0$ .

Des modifications peuvent être ajoutées au cours de la simulation sur le réseau, pour atteindre le but recherché.

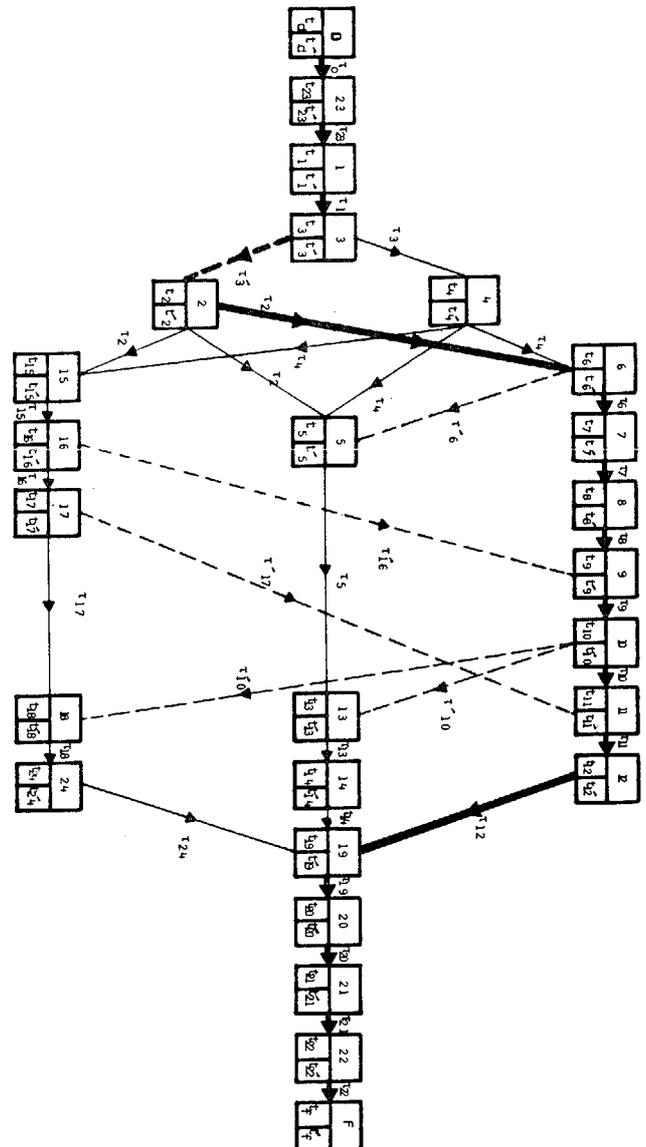
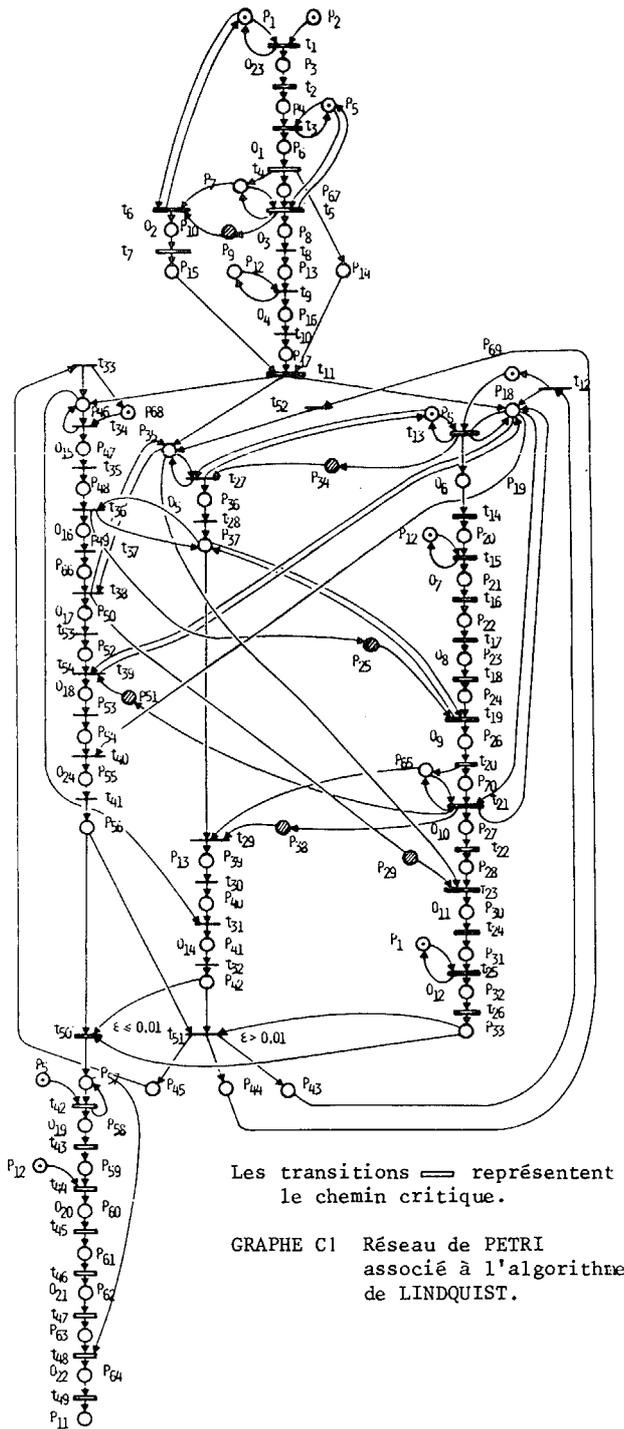
- Programme S2 : Recherche du chemin critique. Ce programme suppose que le réseau étudié a été simulé et corrigé par le programme précédent. On calcule dans ce programme, les instants de tir des transitions au plus tôt et au plus tard ; ceci permet de déterminer le chemin critique et la marge de flottement  $\Delta t = t_i - ti$  - ti sur les transitions ou les tâches.

- Programme S3 : Optimisation du vecteur d'opérateurs. Ce programme exploite les résultats du programme 2. En effet, dans ce programme on suppose qu'un vecteur d'opérateurs initial est déterminé ; ceci se traduit alors par des contraintes supplémentaires associées aux transitions du réseau étudié. A partir de ce vecteur on cherche le vecteur d'opérateurs optimal relatif au chemin critique.

A titre d'exemple, nous avons déterminé le chemin critique (lignes renforcées) sur les graphes correspondant à l'algorithme de LINDQUIST.



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHE ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR).



GRAPHE C2 Réseau P.E.R.T. associé à l'algorithme de LINDQUIST.

Représentation par tâches et contraintes. Le chemin critique correspond aux arcs en traits renforcés.

### 7. - CONCLUSION.

Les réseaux de PETRI étudiés dans cet article peuvent être considérés comme des macro-réseaux ou des micro-réseaux.

La méthode adoptée est caractérisée par l'approche suivante :

- a) - Elaboration des micro-réseaux de PETRI interprétés et simulation par programme afin de maîtriser le fonctionnement du système voulu.
- b) - Elaboration des réseaux P.E.R.T. correspondant aux micro-réseaux de PETRI interprétés et rangement ordonné de ces réseaux afin de déterminer les opérateurs en fonctionnement dans chaque séquence. On en déduit la structure du réseau et la synthèse de l'automate de commande.

c) - Détermination du chemin critique pour chaque réseau, et, en conséquence, du temps de calcul optimal pour chaque algorithme.

d) - Simulation par programme des macro-réseaux et comparaison de différents algorithmes étudiés. Ceci conduit à analyser les points suivants :

- Détermination de la structure générale du système adaptée à un ensemble d'algorithmes.
- Estimation du volume de mémoire nécessaire.
- Détermination du nombre optimal d'opérateurs (au sens du chemin critique).



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHE ET DE RESEAU DE PETRI

APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR)

BIBLIOGRAPHIE

A - OUVRAGES OU ARTICLES RELATIFS AUX RESEAUX DE PETRI  
ET AUX RESEAUX P.E.R.T.

- [A1] HUFFMAN D.A. "The Synthesis of Sequential Switching Circuits" Journal of the Franklin Institute, vol 257, n° 3-4 1954
- [A2] MOORE E.F. "Gedanken Experiments Machines" in "Automata Studies". Princeton University Press. Princeton, NJ, 1956
- [A3] MEALY G.H. "A Method for Synthesizing Sequential Circuits" Bell System Technical Journal, Vol.34, PP 1045-1079, Sept 1955
- [A4] KNUTH D.E. "Fundamental Algorithms" Vol.1, The Art of Computer Programming. Addison Wesley Publishing Company, 1968.
- [A5] COPII.M. ELGOT.C.C. - WRIGT J.B. "Realization of Events by Logical Nets" Journal of the A.C.M., Vol. 5, Avril 1958
- [A6] MC. NAUGHTON - YAMADA M. "Regular Expressions and State Graph for Automata" Trans. Cf the IRE, vol. ec. 9, n° 1, 1960
- [A7] GRASSELLI A. - LUCCIO F. "A Method for Minimizing the Number of Internal States in Incompletely Specified Sequential Machines". IEEE T.E.C., vol E.C. 14, n°3, PP. 350-359 juin 1965
- [A8] KLEENE S.C. "Realization of Events in Nerve Nets and Finite Automata in Automata Studies", C.E. Shannon and J. Mc Carthy Edts. Princeton, N, J 1956
- [A9] PAULL M.C. - UNGER S.H. "Minimizing the Number of States Incompletely Specified Sequential Switching Functions". IRE T.E.C, vol, E.C.8, PP. 356-367- Sept. 1959
- [A10] ARMSTRONG D.B. "A Programmed Algorithm for Assigning Internal Codes to Sequential Machines". IRE Trans. On E.C., vol. E.C. 11, PP.466-472, Août 1962
- [A11] DOLOTTA T.A. - MC CLUSKEY E.J. "The Coding of Internal States of Sequential Circuits". IEEE T.E.C., vol E.C. 13, PP. 549-562, Octobre 1964
- [A12] ELSEY J. "An Algorithm for the Synthesis of Large-Sequential Switching Circuits" PH.D University of Illinois, Urbana 1963
- [A13] HAZELTINE B. "Encoding of Asynchronous Sequential Circuits" IEEE T.E.C., vol.E.C. 14, PP 727-729 oct. 1965
- [A14] KOHAVI Z. "Secondary State Assignment for Sequential Machines" IEEE T.E.C., vol EC. 13, PP. 193-203, Juin 1964
- [A15] MAKI G.K. - TRACEY J.H. "A State Assignment Procedure for Asynchronous Sequential Circuits" IEEE T.C., vol. C. 20, n° 6, PP. 666-668 Juin 1971
- [A16] TORQUE J.C. "Contribution à la minimisation des fonctions logiques" These de spécialité EEA. Université Paul Sabatier, Toulouse n° 1378, 25 Avril 1973
- [A17] SMITH R.J. "Synthesis Heuristics for Large Asynchronous Sequential Circuits". PHD Disserta-tion, University of Missouri, Rolla, 1970
- [A18] PETRI C.A. "Kommunikation mit Automaten" Schriften des Rheinisch - Westfalischen Inst. Instrumentelle Math. an der Universtät. Bonn, HFT. 2, Bonn. 1962
- [A19] A.W. HOLT et F. COMMONER "Events and Conditions, Record of the Project MAC", Conference on concurrent systems and Parallel computation A.C.M., New York, 1970, P. 3-52
- [A20] M. HACK, "PETRI Net Languages, Project MAC", Computation Structures Group, Memo124, Juin 75
- [A21] M. HACK, "Analysis of Production Schemata by PETRI Nets", Master of Science, M.I.T., 1972
- [A22] M.HACK, "Decision Problem for PETRI Nets and Vector Addition" Systems MAC Technical Memorandum
- [A23] R. VALK, "Self Varying Nets", Journées d'études A.F.C.E.T. sur les réseaux de PETRI-Institut de Programmation de Paris VI Mars 77
- [A24] R. VALK et G. VIDAL-NAQUET "On the Rationality of PETRI Nets Languages" 3G1 Conference on Theoretical Comp. Science, Darmstadt, 1977
- [A25] J. SIFAKIS "Use of PETRI Nets For Performance Evaluation", 3rd Int. Symp. On Modelling and Performance Evaluation of Computer Systems, Bonn, Octobre 1977
- [A26] G. BERTHELOT et G. ROUCAIRAL "Reduction of PETRI Nets", Proc of the symp. On M.A.F.C.S. 76, Lecture notes in Comp Sciences, n° 45, Springer-verlag
- [A27] C. RAMCHANDANI "Analysis of Asynchronous Concurrent Systems by PETRI Nets", Ph.D. M.I.T. 1973
- [A28] L.H. LANDWEBER and E.L. ROBERTSON "Properties of Conflict-Free and Persistent PETRI Nets", Journal of the Association for computing Machinery, vol 25, n° 3 Juillet 1978. PP 352-364
- [A29] M. MOALLA, J PULOU et J. SIFAKIS "Réseaux de PETRI synchronisés" RAIRO. Automatique /Systems Analysis and Control, vol 12, n° 2, 1978 PP 103-130.
- [A30] J.L BAER, D.P BOUET and G. ESTRIN "Legality and other Properties of Graph Models of Computations" Journal of the A.C.M., vol 17, n° 3 Juillet 1970 P. 543-554
- [A31] C.W. ROSE "LOGOS And the Software Engineer", Proc. Fall joint comp. Conf. A.F.I.P.S. Press 1972, P, 311-323
- [A32] J.D. NOE AND G.J. NUTT "Macro E-Nets for Representation of Parallel Systems", IEEE Trans. On computers, C22, n° 8, Août 1973, P. 718-727
- [A33] M. MOALLA, G. SAUCIER, J. SIFAKIS AND M. ZACHARIADES. "A Design Tool for the Multilevel Description and Simulation of Systems of Interconnected Modules", 3rd Annual Symp. On comp Architecture, Tampa, Fla. Janvier 76
- [A34] R. VALETTE. "Sur la description, l'analyse et la validation des systèmes de commande parallèles" These d'état, Université Paul Sabatier, Toulouse Novembre 1976
- [A35] P. AZEMA, R. VALETTE et M. DIAZ, PETRI "Nets as a Common Tool for Design Verification and Hardware Simulation", 13th Design Automation Conference Palo Alto, Juin 76
- [A36] C. ANDRE. "Sur une méthode de conception assistée par ordinateur des systèmes logiques à évolutions simultanées", thèse de 3ème cycle, Nice Juin 75.
- [A37] M.C. MICHEL "Ensemble d'outils pour la conception assistée par ordinateur de systèmes numériques à haute performance", Colloque GRETSI PP - 42/1, 42/6 NICE 79.



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHE ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHESE  
DE L'OPERATION DE RICCATI RAPIDE (ORR)

- [A38] M. GHERBI, M. MICHEL  
"Programmes de conception des micro-systèmes numériques à haute performances". Colloque GRETSI PP. 38/1, 38/5 NICE 79
- [A39] M. CHAMBON, M. GHERBI, M. PAPACOSTA  
"Simulateur de Réseaux de PETRI". Colloque GRETSI PP - 37/1 - 37/6 NICE 79
- [A40] MOALLA (M) SIFAKIS (J) SILVA (M)  
"A la recherche d'une méthodologie de conception sûre des automatismes logiques basée sur l'utilisation des réseaux de PETRI" Math.appl. informat., F.R.A., (1978,n° 138, 143)
- [A41] R.DAVID, E. MITRANI, R. TELLEZ-GIRON  
"Emploi des CUSA pour la synthèse directe de systèmes asynchrones décrits par des graphes ou des réseaux de PETRI". Colloque AFCET-ADEPA : Automatismes logiques, Recherches et applications industrielles, PARIS, Déc 1976
- [A42] M. COURVOISIER. "Etude des systèmes logiques de commande asynchrone à évolutions simultanées" Thèse Doc.ès-Sciences, Univ. Paul Sabatier, Toulouse, Fév. 1974
- [A43] M. BLANCHARD, J.C. CAVARROC, J.GUILLON, G.THULLIER. "Conception modulaire d'automatismes séquentiels asynchrones" DERA - Télémeccanique Electrique, Rapport DGRST 71.7. 2912.01 Janvier 1976
- [A44] E.DACLIN, M. BLANCHARD  
"Synthèse des systèmes logiques", Ed. Cepadues Collection Sup-Aéro, Déc 76
- [A45] J.M AYACHE, P. LE DANOIS  
"Synthesis of Logic Systems with PLA'S ; Journées d'étude : Logique Calculée ou logique programmée, Lausanne, Mars 1977, PP. 89-95
- [A46] K.C. LEUNG C. MICHEL, P. LEBEUX  
"Logical Systems Design Using PLA'S and PETRI Nets - Programmable harwired systems", IFIP Congress Toronto 1977, PP. 607-611.
- [A47] C.R. CLARE "Designing Logic Systems Using State Machines" Ed.Mc. Graw-Hill, New-York, 1973
- [A48] H.A SHOLL "Direct Transition Memory its Application in Computer Design", IEEE trans. On comp, vol 23, n° 10, Oct. 1974, PP. 1048-1061
- [A49] H.A SHOLL "Design of Asynchronous Sequential Net work Using ROM" IEEE Trans. On comp, vol 24, n° 2, Fév. 1975, PP. 195-206
- [A50] L. TOURRES "Une méthode nouvelle d'étude des systèmes logiques et son application à la réalisation d'automatismes programmes". Revue générale de l'Electricité, T.85, n°3, Mars 1976
- [A51] M. BLANCHARD, J. GILLON  
"Réalisations logiques programmées des réseaux de PETRI" Journée d'Etude : logique câblée ou logique programmée Lausanne, Mars 77, PP.51-57
- [A52] M. SILVA "Contribution à la synthèse programmée des automatismes logiques", Thèse Docteur - Ingénieur LAG, GRENOBLE Juin 78
- [A53] J. PLEYBER, M. SILVA  
"Software Spécification for Sequential Processes" IFAC-IFIP Workshop on real time programming, Eindhoven, Juin 1977
- [A54] A. ALABEAU, J. FIGUERAS  
"Some Aspects of the Implementation of a Microcomputer of Sequential Systems", MMI-75, PP. 162-166
- [A55] A. KAUFMANN, G. DESBAZEILLE  
"La méthode du chemin critique" DUNOD Paris 1966
- [A56] M. MOALLA "L'approche fonctionnelle dans la vérification des systèmes informatiques. Proposition d'un ensemble de méthodologies" Thèse de Docteur Ingénieur, INPG-ENSIMAG, Grenoble, Déc 1976
- [A57] J. SIFAKIS "Use of PETRI Nets for Performance Evaluation" Measuring, Modelling and Evaluating Computer Systems, North Holland Publ. Co. 77, PP -75-93
- [A58] S. GHOSH "Some Comments, on Timed PETRI Nets", Journées d'étude AFCET Réseaux de PETRI, Mars 77, PP. 151-163

B - OUVRAGES OU ARTICLES RELATIFS AU FILTRAGE OPTIMAL ET A L'EQUATION DE RICCATI ASSOCIEE.

- [B1] R. KALMAN "A New Approach to linear Filtering and Prediction Problems" Journal of Basic Engineering, p. 34-35, Mars 1960
- [B2] J.C. RADIX "Introduction au Filtrage numérique" Ed. Eyrolles 1970
- [B3] P. FAURRE "Réalisations Markoviennes de processus stationnaires" Doctorat d'état Paris VI Déc 72
- [B4] C. BOZZO. "A Discrete Suboptimal Adaptive Estimation Scheme for Linear systems with Unknow Plant and Measurement Noise Covriances" IFAC, 6th Trienna World Congress Boston Mass. USA 1975.
- [B5] B. MOLINARI "The Stabilizing Solution of the Discrete Algebraic RICCATI Equation" - IEEE Trans. Aut. Control. P.396-399 Juin 1975
- [B6] T.KAILLATH, L. JUNG  
"The Asymptotic Behavior of Constant Coefficient RICCATI Differential Equations" IEEE Aut. Control. p. 385-388 Juin 1976
- [B7] A. HEWER. "Analysis of a Discrete Matrix RICCATI Equation of linear Control and KALMAN Filtering" Journal of Mathematical Analysis and applications, n° 42, p 226-236 - 1973
- [B8] S. KLEINBANOV, V. PRIVAL'SKH, I. TIME  
"Stabilization of Coefficients in Discrete KALMAN Filter" Automatica i telemekhanika n° 3 p. 76-82 Mars 1974
- [B9] D. RAPPAPORT, L. SILVERMAN  
"Structure and Stability of Discrete Time Optimal Systems" IEEE Aut. Control Vol 16, n° 3, p. 227-233 Juin 1971
- [B10] P. CAINES, D. MAYNE.  
"On the Discrete Matrix RICCATI Equation of Optimal Control" INT.J. Control. Vol 12, n° 5 p. 785-794, 1970
- [B11] J.M. RODRIGUEZ - CANABAL  
"The Geometry of the RICCATI Equations" Contrat AFSOR - 71-2141 reproduit par National Technical Information Service - Juin 1972
- [B12] R. BUCY, D. RAPPAPORT, L. SIVERMAN  
"Correlated Noise Filtering and invariant Directions for the RICCATI Equations" IEEE Aut Control. Vol 15 n°5, p.535-540, Octobre 1970
- [B13] D. RAPPAPORT. "Constant Directions of the RICCATI Equation" Automatica Vol. 8, p 175-186 Bergamon Press 1972
- [B14] A. BARRAUD. "Sur la résolution numérique des Equations de RICCATI - cas discret". RAIRO, Octobre 1974-J-3 p. 75 à 91
- [B15] M. LABARRERE, J.P. KRIEF, B. GIMONET  
"Le Filtrage et ses applications" CEPADUES Editions 1978



ANALYSE D'ALGORITHMES ET CONCEPTION DES STRUCTURES DE TRAITEMENT  
ASSOCIEES EXPLOITANT LES NOTIONS DE GRAPHE ET DE RESEAUX DE PETRI  
APPLICATIONS AU FILTRAGE OPTIMAL ET A LA SYNTHÈSE  
DE L'OPERATEUR DE RICCATI RAPIDE (ORR)

- [B16] C. BOZZO "Filtrage des processus stochastiques linéaires - Recherche de la solution d'équilibre de l'équation de RICCATI discrète associée au filtre de KALMAN-BUCY". Cours de DEA Marseille 1975.
- [B17] P. JOSEPH "Space Control Systems - Attitude, Rendez-vous and Docking". Course notes 1964 - Engineering Extension Course, UCLA Los Angeles, Californie.
- [B18] J. SWARTENGEN. "Fortran Subroutines to solve the LIAPUNOV and the RICCATI Equations". FOA 2 Rapport 1, n° C2655 E4. E5 Février 1974.
- [B19] R. BUCY, P. JOSEPH  
"Filtering for stochastic Processes with Applications to Guidance". Interscience Publishers 1968
- [B20] G. FAVIER, G. ALLENGRIN  
"Identification du gain d'un filtre linéaire optimal - Etude comparative d'algorithmes de filtrage adaptatif". Fiche technique S 10 324 GESTA/CAPCA, Juillet 1975
- [B21] M. CHIDAMBARA, A. SHIVAPRASAD  
"Digital Computer Study of the performance of a Digital Filter for Radar Tracking" Electro technology, p. 82-85 Septembre 1974
- [B22] G. BIERMAN. "A Comparison of Discrete Linear Filtering Algorithms" IEEE Aerospace and electronic systems vol. AES9, n°1, p. 28-37 Janvier 1973
- [B23] N. CARLSON. "Fast Triangular Formulation of the Square Root Filter" AIAA Journal Vol 11, n°9, Septembre 1973, p. 1259-1265
- [B24] J. SCHIESS. "Vectorization of linear Discrete Filtering Algorithms" NASA Technical Memorandum rapport n° TMX-3527, Juillet 1977
- [B25] C. CHOE, B. TAPLEY. "New Method for Propagating the square Root Covariance Matrix in Triangular Form". AIAA Journal Vol 13, n°5, Mai 1975
- [B26] P. KAMINSKY, A. BRYSON, S. SCHMIDT  
"Discrete Square Root Filtering : A Survey of Current Techniques". IEEE Aut. Control. Vol 16, n°6, Décembre 1971, p. 727-736
- [B27] C. THORNTON, G. BIERMAN  
"Givens Transformations Technics for KALMAN Filtering". Acta Astronautica Vol.4, p. 847-863, Bergamon Press 1977
- [B28] M. MORF, T. KAILATH  
"Square Root Algorithms for Discrete Sequential Estimation" IEEE Aut. Control Vol 20, n°4, 1975
- [B29] D. VAUGHAN. "A Non Recursive Algebraic Solution for the Discrete Time Equation". IEEE Aut. Cont p. 597-599, Octobre 1970
- [B30] R. MONZINGO. "A Non Recursive Algebraic Solution for the Discrete Smoothed Error Covariance Matrix". IEEE Aut, Control. p. 175, Avril 73
- [B31] D. CLEMENTS, B. ANDERSON  
"Polynomial Factorization via the RICCATI Equation" SIAM J. APPL. MATH. Vol 31, n°1, p. 179-205, Juillet 1976
- [B32] D. VAUGHAN. "A Negative Exponential Solution for the Matrix RICCATI Equation". IEEE Trans. Aut. Control Vol. AC 14, p. 72-75. Février 1969
- [B33] A. GUILBERT. "Etude et comparaison de quatre algorithmes de réalisation stochastique". Doctorat de 3ème cycle Marseille, Mai 1977.
- [B34] C. BOZZO. "Transformée de LAGUERRE d'un signal continu - Application à l'étude du régime asymptotique du filtre de KALMAN". CRAS t277, Juillet 1973
- [B35] A. LINDQUIST. "A New Algorithm for Optimal Filtering of Discrete Time Stationary Processes" SIAM J Control. Vol 12, n°4, p.736-746, Novembre 1974
- [B36] A. LINDQUIST. "Optimal Filtering of Continuous Time Stationary Processes"
- [B37] A. LINDQUIST. "Some Reduced-Order Non RICCATI Equations for linear Least-Square Estimation: the Stationary Single Output Case" Int. J. Control. 1976, vol. 24, n°6, p. 821-842
- [B38] A. BARRAUD. "A Numerical Algorithm to Solve  $A^T X A - X = Q$ " IEEE Aut. Control. Vol n°5, Octobre 1977, p. 883-885
- [B39] S. BARNETT. "Matrices, Polynomiales and linear Time Invariant Systems". IEEE Aut. Control. Vol 18, n°1, Février 1973
- [B40] R. BARTELS, G. STEWARD  
"Algorithm 432, Solution of the Matrix Equation  $A X + X B = C$ " Communications of the A.C.M. Vol 15, n°9, Septembre 1972
- [B41] I. SPACE, S. BARNETT  
"Comparison of Numerical Methods for Solving LIAPUNOV Matrix Equations" Int. Journal Control Vol. 15, n°5, p. 907-915, 1972
- [B42] W. HOSKINS, D. MEEK, D. WATSON.  
"The Solution of  $A^T Q + Q A = -C$ " IEEE Aut. Control. Vol 22, n°5, p. 882-883, Octobre 1977
- [B43] S. BARNETT. "Simplification of the LIAPUNOV Matrix Equation  $A^T P A - P = Q$ ". IEEE Aut. Control. p. 446-447, Août 1974
- [B44] D. ZILMER. "Rapport TP 5180 Naval Weapons Center", Mai 1971
- [B45] P. BELANGER - B. CAREW  
"Identification of Optimum Filter Steady State Gain for Systems with Unknown Noise Covariances" IEEE Vol. AC 18, n°6, Décembre 1973
- [B46] P. BELANGER. "Estimation Noise Covariance Matrices for a linear Time Varying Stochastic Process". Automatica Vol . 10, p. 267-275. 1974
- [B47] A. GIULIERI, C. BOZZO  
"Analyse des méthodes de résolution numérique de l'équation discrète de RICCATI - Application à l'opérateur de RICCATI Rapide". Septième colloque GRETSI - PP. 7411-7418, Nice 1979.
- C - ORGANISATION DE STRUCTURES DE TRAITEMENT
- [C1] OL BANDMAN  
"Synthèse de la commande microprogrammée asynchrone de processus parallèles" (en russe) Kibernetika n° 1 1980.

