



Soutenance de thèse :

**Quelques problèmes et solutions en ordonnancement
temps réel pour systèmes répartis**

15 septembre 1999, 15h00

Projet REFLECS

Jean-Francois.Hermant@inria.fr



Le domaine scientifique

En ordonnancement temps réel, il existe des problèmes fermés, qui sont tous établis pour des systèmes monoprocesseurs ou centralisés.

Même dans le cas où l'on se restreint aux systèmes centralisés, tous les problèmes d'ordonnancement temps réel dans lesquels on fait intervenir du partage d'objets rémanents sont des problèmes ouverts.

C'est a fortiori le cas pour les problèmes d'ordonnancement temps réel pour les systèmes répartis (tel le problème HRTDM) et en présence d'objets rémanents partagés (tel le problème HRTDT).



Plan (1/2)

Partie I : Ordonnancement temps réel pour ressources distribuées multiples

✧ Le problème HRTDT

- Les modèles : $\langle m.HRTDT \rangle$
- Les propriétés : $\langle p.HRTDT \rangle$

✧ La solution S du problème HRTDT

- L'architecture : le modèle clients/serveurs distribués
- Les algorithmes : EDF, FIFO et σ_τ (ordonnancement précalculé)
- L'oracle de dimensionnement Oracle.S pour la solution S
 - L'outil de placement
 - Les conditions de faisabilité



Plan (2/2)

Partie II : Ordonnancement temps réel pour ressource distribuée unique

- ✧ Le sous-problème HRTDM du problème HRTDT
 - Les modèles : $\langle m.HRTDM \rangle$
 - Les propriétés : $\langle p.HRTDM \rangle$

- ✧ La solution S du problème HRTDM
 - Les algorithmes : EDF (centralisé) et CSMA/DDCR (distribué)
 - L'oracle de dimensionnement Oracle.S pour la solution S
 - L'outil de placement
 - Les conditions de faisabilité



Introduction (1/3)

Domaine applicatif considéré : avionique modulaire.

Problèmes applicatifs examinés : combat aérien et problèmes connexes (problèmes réels décrits par Dassault Aviation dans le cadre du programme ASAAC de l'OTAN).



RAFALE B

DASSAULT
AVIATION



Introduction (2/3)

Ces problèmes ont été capturés en suivant une méthode d'ingénierie système prouvable, à savoir la méthode TRDF (G. Le Lann)

⇒ spécification du problème HRTDT
(the Hard Real-Time Distributed Transactions problem).



Introduction (3/3)

✧ Un ensemble de modèles (les hypothèses du problème HRTDT) :

$\langle m.HRTDT \rangle$

✧ Un ensemble de propriétés (les exigences du problème HRTDT) :

$\langle p.HRTDT \rangle$

– Les propriétés logiques : $\langle p\Lambda.HRTDT \rangle$

(e.g., atomicité des transactions distribuées).

– Les propriétés physiques : $\langle p\Phi.HRTDT \rangle$

(e.g., temps de réponse bornés supérieurement).



Les objectifs (1/2)

Fournir les spécifications suivantes :

✧ [S] = la spécification de la solution système S :

- L'architecture du système ;
- Les algorithmes.

✧ [Oracle.S] = la spécification de l'oracle de dimensionnement pour la solution S :

- L'outil de placement ;
- Les conditions de faisabilité.



Les objectifs (2/2)

L'oracle de dimensionnement :

✧ En entrée :

Une quantification $Q(\langle \text{HRTDT} \rangle)$.

✧ En sortie :

- Si $Q(\langle \text{HRTDT} \rangle)$ n'est pas faisable, une réponse négative est retournée par l'oracle ;
- Si $Q(\langle \text{HRTDT} \rangle)$ est faisable, la quantification $Q(\langle S \rangle)$ est retournée par l'oracle (e.g., tailles maximales des files d'attente, nombre minimum de processeurs, etc.).



Les preuves

Dassault Aviation exigeait les preuves suivantes :

- ✧ Les propriétés $\langle p\Lambda.HRTDT \rangle$ sont vérifiées
pour toute quantification $Q(\langle HRTDT \rangle)$ [même si la quantification $Q(\langle p\Phi.HRTDT \rangle)$ est déclarée non faisable par l'oracle]
(*e.g., les transactions distribuées sont sérialisables*).
- ✧ Les propriétés $\langle p\Phi.HRTDT \rangle$ sont vérifiées
pour toute quantification $Q(\langle HRTDT \rangle)$ qui est faisable
[la quantification $Q([S])$ de la solution S existe]
– $Q(\langle HRTDT \rangle)$ est déclarée faisable par l'oracle –
(*e.g., les échéances des transactions sont toujours satisfaites*).



Le problème HRTDT (1/2)

Le problème HRTDT est ouvert. Quelles sont les principales difficultés ?

✧ Les modèles $\langle m.HRTDT \rangle$:

- Le modèle de transactions : Graphes orientés finis
(ce modèle « domine » le modèle séquentiel)
- Les transactions manipulent des objets modifiables rémanents distribués
- Le modèle événementiel des arrivées (les demandes d'activation des transactions) : arbitraire unimodal.
 \forall événement e , au plus n_e arrivées dans toute fenêtre temporelle glissante de w_e unités de temps
(ce modèle « domine » le modèle périodique / sporadique)



Le problème HRTDT (2/2)

- ✧ Les propriétés < p.HRTDT > :
 - Sérialisabilité (transactions)
 - Ponctualité (échéances de terminaison strictes)
 - Propriétés de tolérance aux fautes
(omises dans cette présentation)

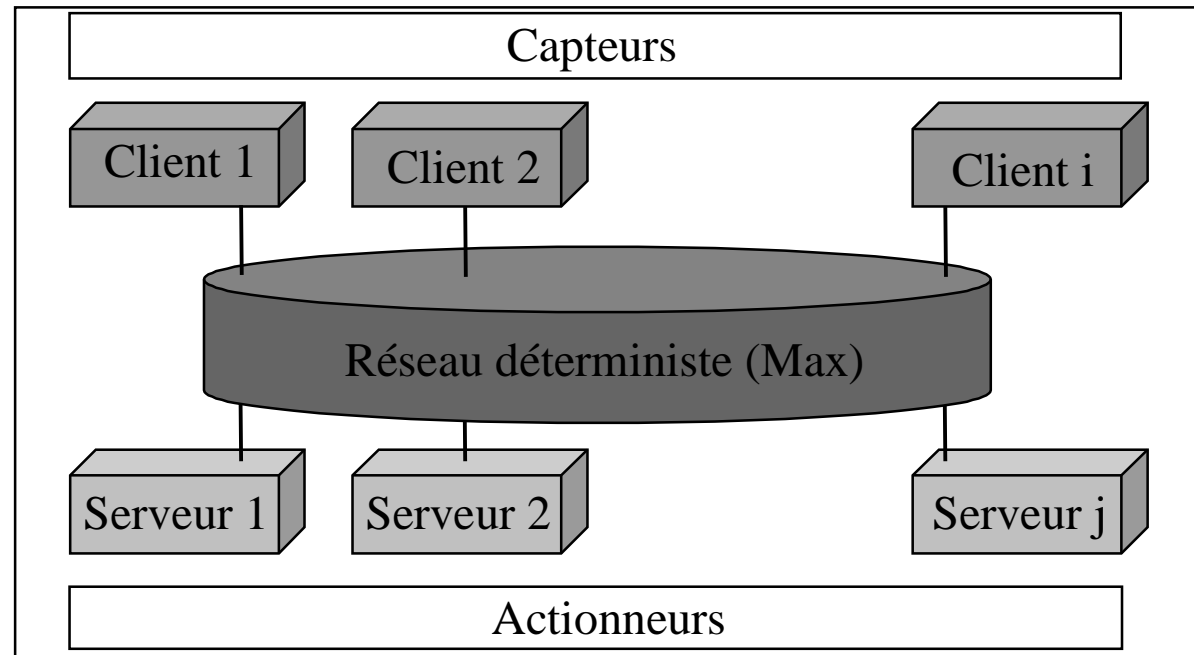
- ✧ La complexité de l'oracle de dimensionnement [Oracle.S] :
Faiblement polynomial en le nombre de transactions.



La solution du problème HRTDT (1/8)

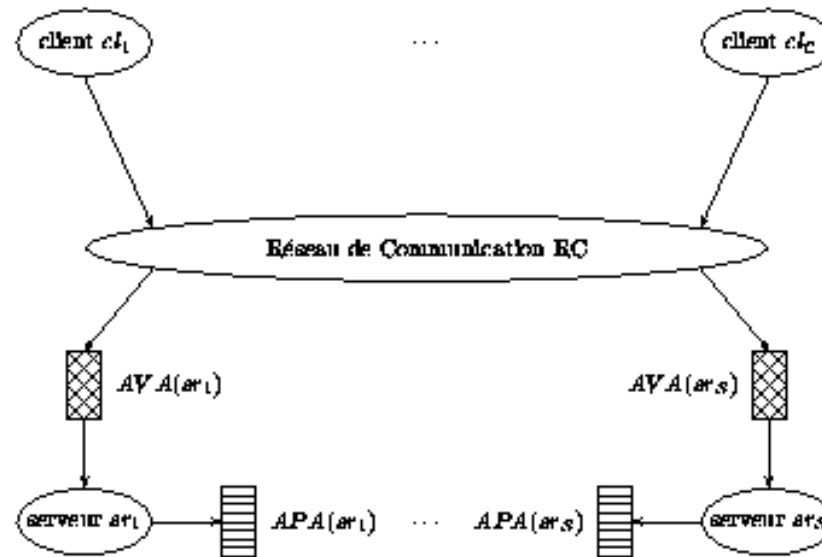
Spécification de la solution système [S] :

- ✧ L'architecture : une architecture embarquée distribuée de type clients/serveurs, exécutant des transactions distribuées, qui accèdent à des objets modifiables rémanents distribués sur les serveurs.



La solution du problème HRTDT (2/8)

Les transactions sont activées par des arrivées d'événements chez les clients.



Chaque client maintient une file d'attente.

Chaque serveur maintient deux files d'attente :

- AVA : Transactions en attente d'accord pour être ordonnancées ;
- APA : Transactions ordonnancées et exécutables.



La solution du problème HRTDT (3/8)

- ✧ Les algorithmes (deux composantes algorithmiques) :
 - Clients : pour les racines des transactions \Rightarrow FIFO ;
 - Serveurs : pour les séquences des transactions.

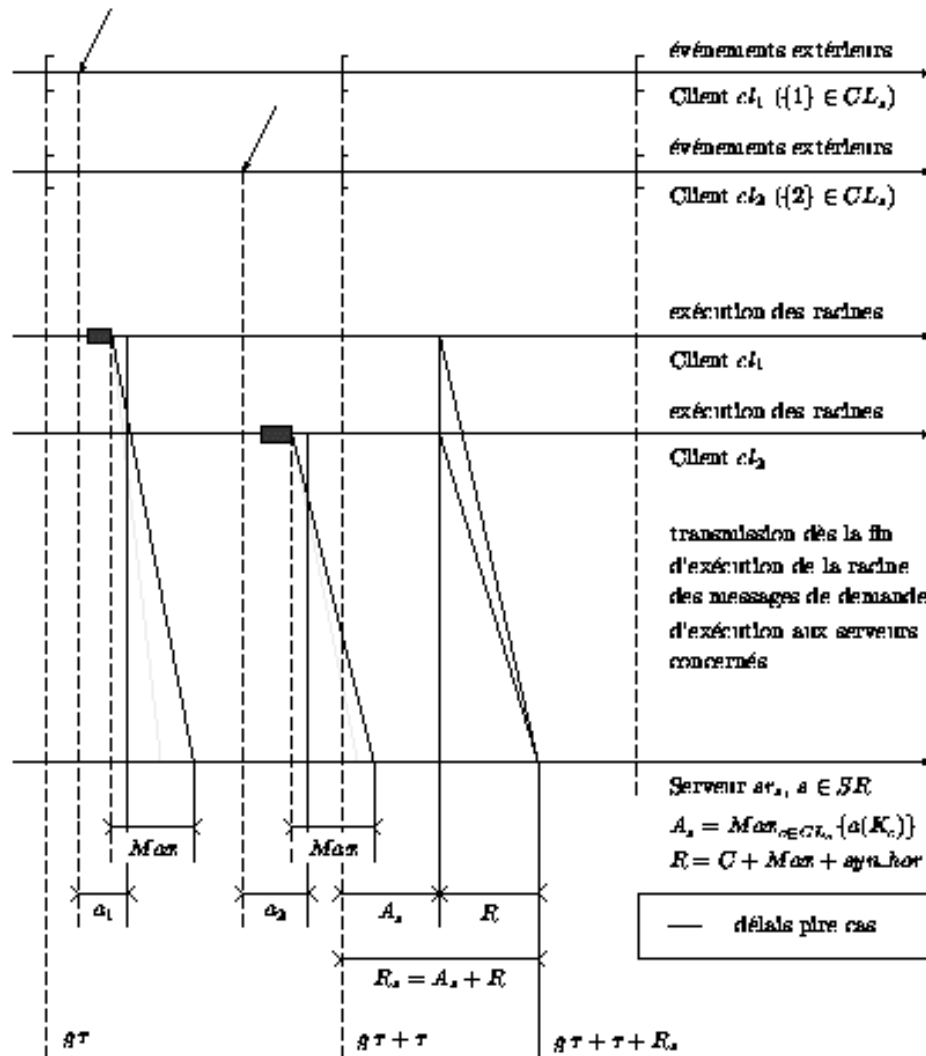
L'algorithme fonctionne de la manière suivante :

- obtention d'un accord entre serveurs portant sur l'ensemble des transactions (en attente) à exécuter ;
- ordonnancement générique précalculé σ , de type oisif et périodique (de période τ), par lots non modifiables, de l'ensemble des séquences de transactions ayant fait l'objet d'un accord ;
- exécution séquentielle de cet ensemble ordonnancé.

Les décisions d'ordonnancement sont prises **en-ligne**.



La solution du problème HRTDT (4/8)



La propriété de sérialisabilité est prouvée par le fait que chaque serveur $sr_s, s \in SR$, a la garantie qu'à l'heure locale $(g+1)\tau + R_s$ il a connaissance de toutes les demandes d'activation des séquences des transactions correspondant à toutes les arrivées s'étant produites dans l'intervalle $[g\tau, (g+1)\tau [$ sur tous les clients $cl_c, c \in CL_s$.

N.B. : Le temps global est obtenu par l'un des algorithmes connus.



La solution du problème HRTDT (5/8)

Spécification de l'oracle de dimensionnement [Oracle.S] :

- ✧ Analyse pire cas exacte : NP-Difficile
 - Modèle de transactions \Rightarrow les périodes occupées diffèrent dès que l'on permute deux transactions ;
 - Modèle d'arrivées arbitraires unimodal \Rightarrow plus dur que le modèle d'arrivées périodiques.

- ✧ Complexité imposée de l'oracle de dimensionnement [Oracle.S]
 - \Rightarrow conditions de faisabilité suffisantes.

- ✧ Les conditions de faisabilité : calcul matriciel en algèbre (max, +).

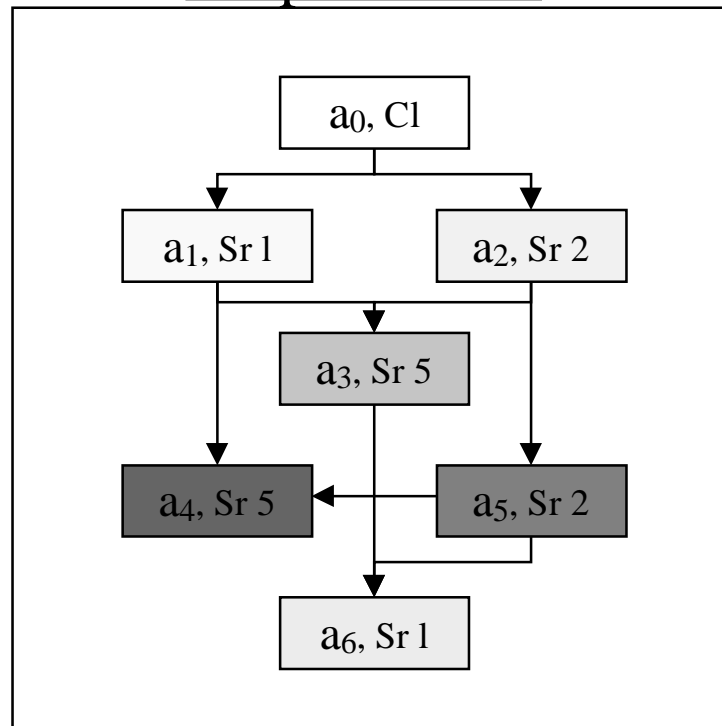


La solution du problème HRTDT (6/8)

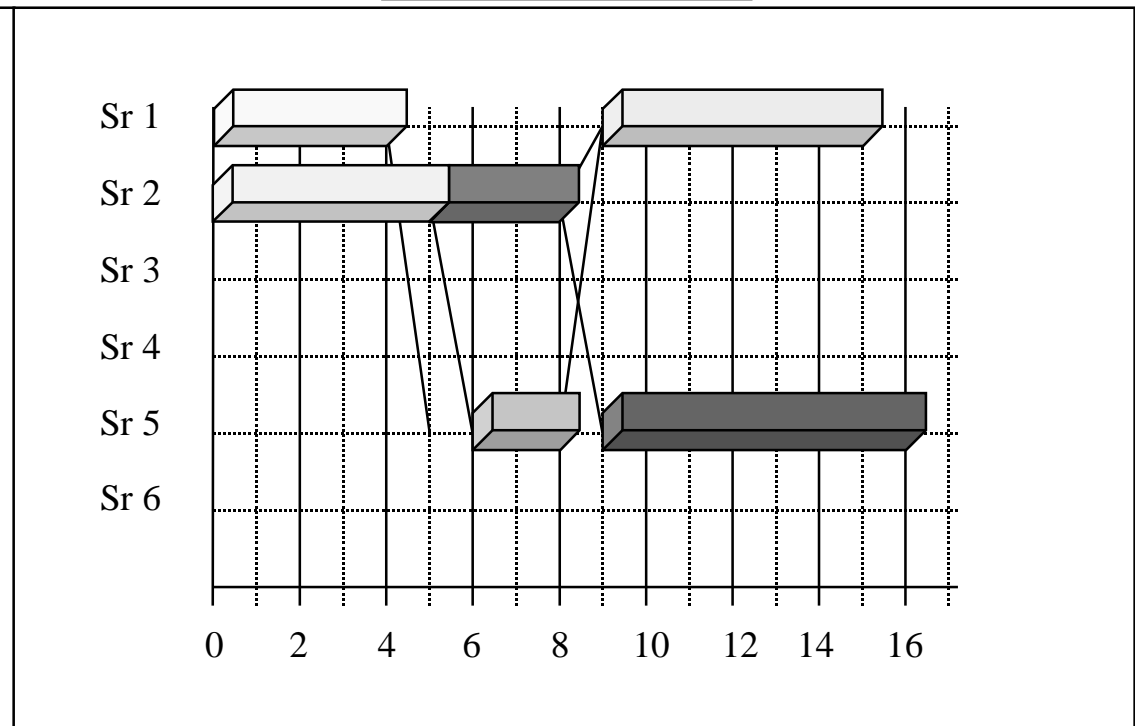
Notations : a_i = actions d'une transaction = méthodes (objets distribués) ;
 Sr_j = serveurs.

Durées : $a_1 = 4$, $a_2 = 5$, $a_3 = 2$, $a_4 = 7$, $a_5 = 3$, $a_6 = 6$; Délai Réseau Max = 1.

Un placement



Une exécution



La solution du problème HRTDT (7/8)

- VL^+ (resp. VL^-) est le vecteur des instants de libération des serveurs après (resp. avant) l'exécution d'une transaction.
- P est la matrice de transition en algèbre $(\max, +)$ correspondant au graphe d'une transaction.

$$VL^+ = P \otimes VL^- = \begin{bmatrix} 14 & 15 & -\infty & -\infty & 9 & -\infty \\ -\infty & 8 & -\infty & -\infty & -\infty & -\infty \\ -\infty & -\infty & 0 & -\infty & -\infty & -\infty \\ -\infty & -\infty & -\infty & 0 & -\infty & -\infty \\ 14 & 16 & -\infty & -\infty & 9 & -\infty \\ -\infty & -\infty & -\infty & -\infty & -\infty & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 15 \\ 8 \\ 0 \\ 0 \\ 16 \\ 0 \end{bmatrix}$$



La solution du problème HRTDT (8/8)

Dassault Aviation a implanté :

- ✧ Les spécifications [S] et [Oracle.S]
- ✧ Un simulateur événementiel de la solution S, dénoté $SIM([S])$

Les enseignements tirés :

- ✧ L'implantation des spécifications [S] et [Oracle.S] : facile
- ✧ $Oracle.S \approx 25$ fois plus rapide que $SIM([S])$



L'oracle de dimensionnement : l'outil de placement

- Matrice Transactions / Objets :

$$\begin{array}{c} O_1 \dots O_j \dots O_n \\ T_1 \\ \vdots \\ T_i \\ \vdots \\ T_r \end{array} \begin{bmatrix} 1 & \dots & 0 & \dots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \dots & 0 & \dots & 1 \end{bmatrix}$$

Cette matrice est spécifiée par les concepteurs du logiciel applicatif.

- Matrice Objets / Serveurs :

$$\begin{array}{c} s_{r1} \dots s_{rj} \dots s_{rs} \\ O_1 \\ \vdots \\ O_i \\ \vdots \\ O_n \end{array} \begin{bmatrix} 0 & \dots & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \dots & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 1 & \dots & 0 \end{bmatrix}$$

Cette matrice est obtenue par résolution d'un problème de placement (les concepteurs de la solution). Dans certains cas, le placement d'un objet sur un serveur est imposé par l'applicatif.



L'oracle de dimensionnement : les C.F. (1/5)

Lois événementielles : arbitraire unimodal.

\forall Transaction T_k , au plus n_k demandes d'activation dans toute fenêtre temporelle glissante de w_k unités de temps.

Hypothèse simplificatrice : Multifenêtrage restreint.

\forall Transaction T_k , $w_k =$ puissance de 2.

$$PPCM_k\{w_k\} = \text{Max}_k\{w_k\} = 2^\alpha.$$



L'oracle de dimensionnement : les C.F. (2/5)

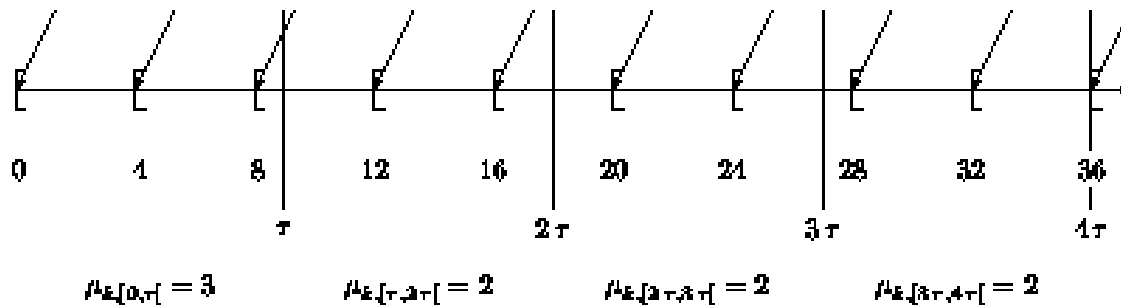
Motifs : motifs synchrones, motifs réguliers et irréguliers.

Motifs synchrones :

- ✧ Période d'invocation τ (de l'ordonnancement précalculé σ_τ).
- ✧ Densité d'arrivée maximale : n_k occurrences en w_k unités de temps.

On isole **1** occurrence (parmi n_k) $\Rightarrow \{ \mu_{k, [9n, 9(n+1)[} \}_{n \in \{0, \dots, 3\}} = \{3, 2, 2, 2\}$.

Exemple : $\tau = 9$; $n_k = 1$ et $w_k = 4$.



$$\Phi_k = \text{PPCM}\{w_k, \tau\} / \tau = 4.$$



L'oracle de dimensionnement : les C.F. (3/5)

Motifs réguliers et irréguliers :

✧ Inf_τ (occurrences dites régulières) :

hypothèses : période d'invocation τ et densité d'arrivée maximale

\Rightarrow au moins $\lfloor \tau/w_k \rfloor$ transactions T_k présentes dans chaque lot ;

✧ Sup_τ (occurrences dites irrégulières) :

hypothèses : période d'invocation τ et densité d'arrivée maximale

\Rightarrow au plus $\lceil \tau/w_k \rceil$ transactions T_k présentes dans chaque lot.

Exemple (suite) :

✧ motif régulier : $\{\mu'_{k, [9n, 9(n+1)[}}\}_{n \in \{0, \dots, 3\}} = \{2, 2, 2, 2\}$;

✧ motif irrégulier : $\{\mu''_{k, [9n, 9(n+1)[}}\}_{n \in \{0, \dots, 3\}} = \{1, 0, 0, 0\}$.



L'oracle de dimensionnement : les C.F. (4/5)

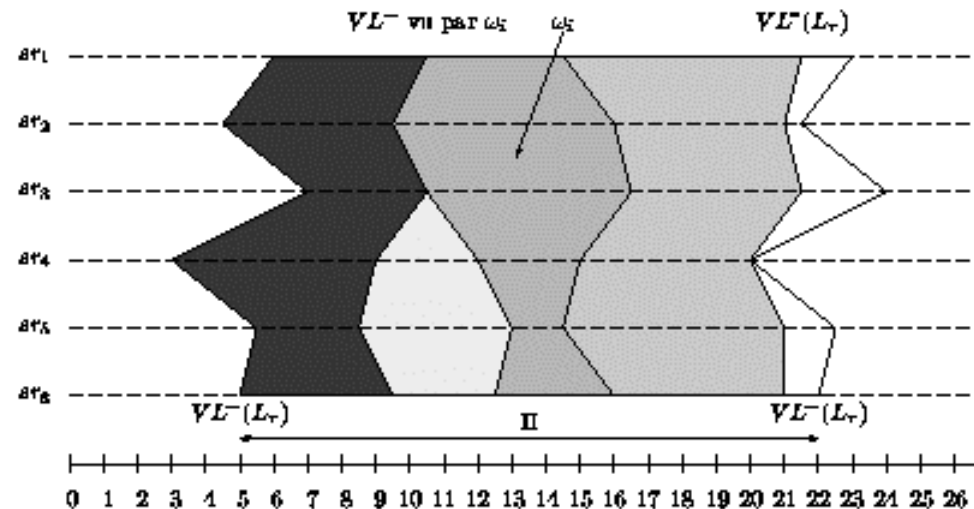
Occurrences dites régulières :

✧ calculs exacts en algèbre (max, +) ; semi-anneaux idempotents :

$(S, \oplus, \otimes) : (\mathbb{R} \cup \{-\infty\}, \max, +)$ et $(M_n(\mathbb{R} \cup \{-\infty\}), \max, +)$.

⇒ Concept d'« invariant de lot » Π

afin de raisonner sur les périodes occupées (théorie de l'ordonnancement)



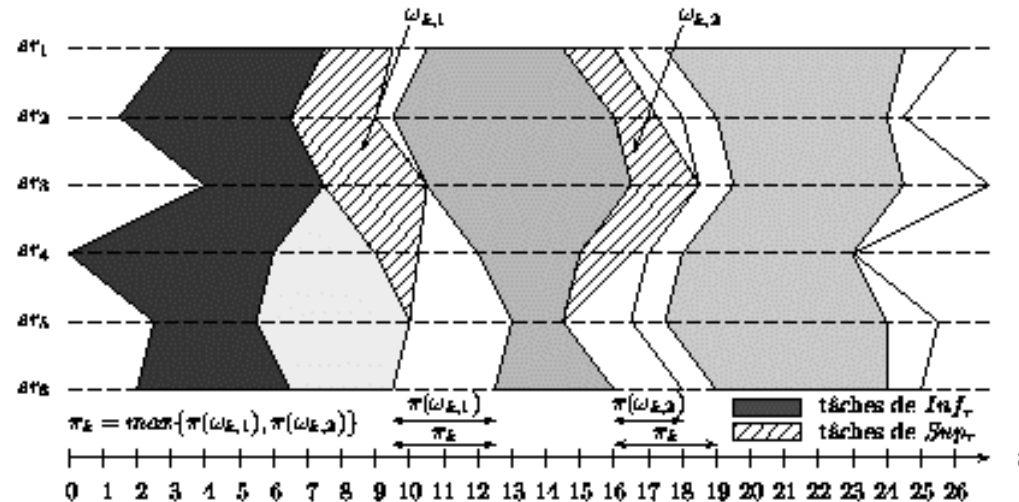
L'oracle de dimensionnement : les C.F. (5/5)

Occurrences dites irrégulières :

- ✧ calculs exacts en algèbre (max, +) ;
- ✧ approximations (maximisations).

⇒ Concept d'« invariant de transaction » π

afin de « rendre » localement le produit \otimes commutatif



Le problème HRTDM (1/3)

Les preuves que la solution [S] résout le problème $\langle \text{HRTDT} \rangle$ s'appuie sur l'hypothèse de l'existence d'une borne Max pour le réseau. Cette hypothèse doit être spécifiée comme un problème, lequel ayant une solution. Nous avons considéré les canaux à diffusion (bus, Ethernets, etc.).

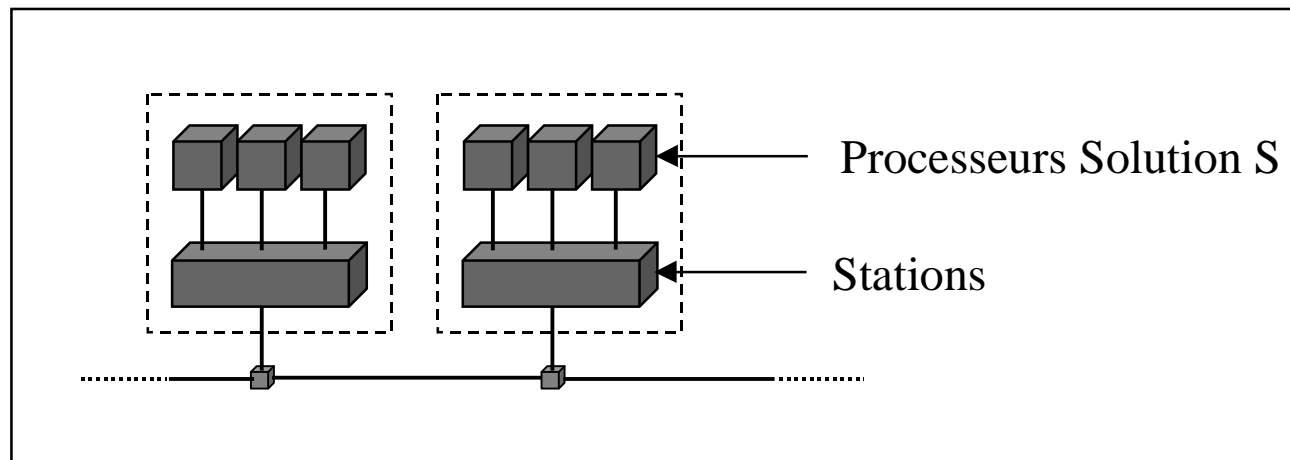
\Rightarrow spécification du problème HRTDM
(the Hard Real-Time Distributed Multiaccess problem).



Le problème HRTDM (2/3)

Les modèles < m.HRTDM > :

- ✧ L'architecture : des processeurs placés sur des stations et des stations interconnectées à un canal à diffusion.



- ✧ Modèle de stations : les stations en nombre fini quelconque multiplexent les files d'attente des messages générés par les processeurs.



Le problème HRTDM (3/3)

Les modèles $\langle m.HRTDM \rangle$ (suite) :

- ✧ Modèle de types de messages : nombre fini quelconque de types, longueurs des messages finies quelconques.
- ✧ Placement quelconque des messages / processeurs / stations.
- ✧ Modèle des arrivées des messages : arbitraires unimodal.

Les propriétés $\langle p.HRTDM \rangle$:

- ✧ Sûreté : les transmissions réussies doivent être mutuellement exclusives.
- ✧ Ponctualité : les échéances des messages sont toujours satisfaites.

Rem. : le problème $\langle HRTDM \rangle$ est « plus difficile » que l'hypothèse faite dans le problème $\langle HRTDT \rangle$ (délais \leq Max)



La solution du problème HRTDM (1/2)

Théorie de l'Information : Avec les canaux à diffusion, les algorithmes optimaux sont les algorithmes en arbre examinés par Gallager, Flajolet, Capetanakis, etc.

Théorie de l'Ordonnancement : Pour la version centralisée du problème $\langle \text{HRTDM} \rangle$, en arrivées périodiques/sporadiques, l'algorithme NP-EDF est optimal (Jeffay, etc.).

⇒ l'algorithme NP-EDF (local à chaque station)

⇒ le protocole CSMA/DDCR (distribué)

(DDCR : Deterministic Deadline-driven Collision Resolution)

- ✧ il émule l'algorithme d'ordonnancement NP-EDF distribué ;
- ✧ il utilise des parcours d'arbres m-aires pour résoudre les collisions.



La solution du problème HRTDM (2/2)

Principes de fonctionnement du protocole CSMA/DDCR :

- ✧ Lorsqu'une collision se produit alors que le canal était libre, chaque station concernée calcule un index temporel (« priorité dynamique ») qui dépend de l'échéance absolue D du message entré en collision.
- ✧ Chaque station exécute localement un algorithme de tri m -aire (TTS) qui progresse en fonction des états du canal.
- ✧ Un index temporel est un # de feuille d'arbre temporel, d'autant plus petit que la différence entre l'échéance D considérée et l'instant courant est petite (c = classe d'équivalence d'échéances).
- ✧ Si une collision se produit sur une feuille d'arbre temporel, un tri d'arbre statique (STS) est entrepris. Chaque station dispose exactement d'un index statique. Elle transmet tous les messages dont l'index temporel \leq l'index temporel courant.



Les conditions de faisabilité - la fonction ξ (1/5)

Afin d'établir les conditions de faisabilité, on doit connaître l'expression de la fonction ξ qui retourne le pire temps de parcours pour isoler k feuilles dans un arbre m -aire à t feuilles ($t = m^n$, $m \geq 2$, $n \geq 1$). On a :

- Tranche canal brouillée :

$$\xi_k^t = 1 + \max_{\substack{k_1 + \dots + k_m = k \\ (k_1, \dots, k_m) \in \{0, \dots, t/m\}^m}} \left\{ \xi_{k_1}^{t/m} + \dots + \xi_{k_m}^{t/m} \right\} \text{ si } k \in \{2, \dots, t\},$$

- Tranche canal occupée :

$$\xi_k^{t/m} = 0 \quad \text{si } k = 1,$$

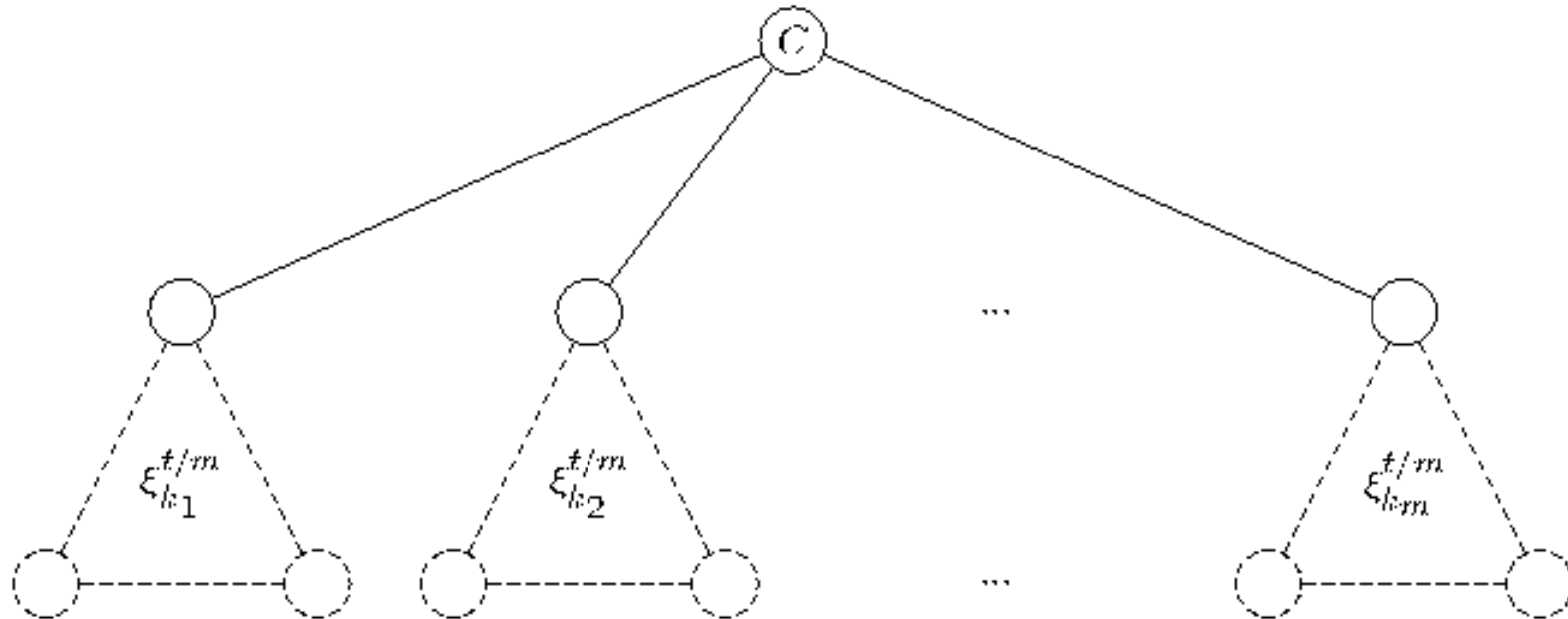
- Tranche canal libre :

$$\xi_k^{t/m} = 1 \quad \text{si } k = 0.$$



Les conditions de faisabilité - la fonction ξ (2/5)

Résolution déterministe de collision :



Arbre m-aire à t feuilles ($t = m^n$, $m \geq 2$, $n \geq 1$)



Les conditions de faisabilité - la fonction ξ (3/5)

La fonction ξ vérifie la relation de récurrence suivante, de type « diviser pour régner » :

- k pairs :

$$\xi_{2^p}^t = \begin{cases} 1 + \sum_{i=0}^{m-1} \xi_{2^{\lfloor \frac{\min(p, t/m) + i}{m} \rfloor}}^{t/m} - 2 \max(0, p - t/m) & \text{si } p \in \{1, \dots, \lfloor t/2 \rfloor\}, \\ 1 & \text{si } p = 0. \end{cases}$$

- k impairs :

$$\xi_{2^{p+1}}^t = \xi_{2^p}^t - 1, \quad p \in \{0, \dots, \lfloor t/2 \rfloor - 1\}.$$

$$t = m^n, \quad m \in \mathbb{N}^* \setminus \{1\}, \quad n \in \mathbb{N}^* \setminus \{1\}.$$



Les conditions de faisabilité - la fonction ξ (4/5)

La forme close de la fonction ξ :

$$\xi_k^t = \begin{cases} \frac{m \left\lceil \log_m \left(m \left\lfloor \frac{k}{2} \right\rfloor \right) \right\rceil - 1}{m - 1} + m \left\lfloor \frac{k}{2} \right\rfloor \left\lceil \log_m \left(\frac{t}{m \left\lfloor \frac{k}{2} \right\rfloor} \right) \right\rceil - \left(k - m \left\lfloor \frac{k}{2} \right\rfloor \right) & \text{si } k \in \{2, \dots, t\}, \\ 0 & \text{si } k = 1, \\ 1 & \text{si } k = 0. \end{cases}$$

$$t = m^n, \quad m \in \mathbb{N}^* \setminus \{1\}, \quad n \in \mathbb{N}^*.$$

Le comportement asymptotique (fonction concave - majorant serré) :

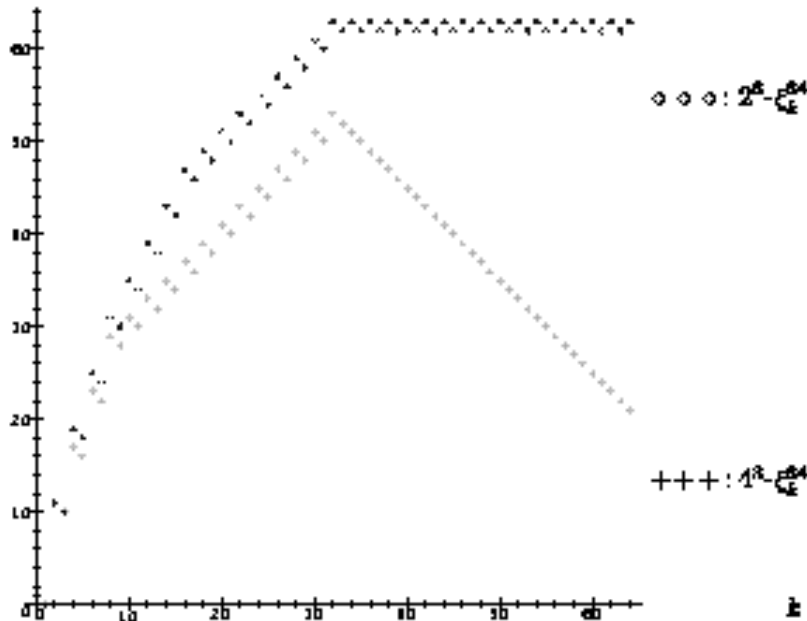
$$\bar{\xi}_k^t = \frac{m \frac{k}{2} - 1}{m - 1} + m \frac{k}{2} \log_m \left(\frac{2t}{k} \right) - k, \quad k \in [2, t].$$

$$\text{Max}_{k \in [2, 2t/m]} \left\{ \bar{\xi}_k^t - \xi_k^t \right\} \leq \left(\frac{\sqrt{\sqrt{3}}}{2 e \ln(3)} - \frac{1}{8} \right) t \leq (9.54\%) t.$$



Les conditions de faisabilité - la fonction ξ (5/5)

Efficacité algorithmique : arbre binaire versus arbre quaternaire.



- $t = 2^{2n} = 4^n$:
arbre quaternaire ;
- $t = 2^{2n+1} = 2 \cdot 4^n$:
arbre binaire.

Version probabiliste de l'algorithme en arbre [Mathys & Flajolet, 1985]
Résultat de l'optimalité du branchement ($m^* = e \Rightarrow$ binaire / ternaire)



Les conditions de faisabilité - la fonction Ξ (1/2)

Afin d'établir les conditions de faisabilité, on doit connaître l'expression de la fonction Ξ qui retourne le pire temps de parcours pour isoler u feuilles dans une suite de v arbres m -aires consécutifs à t feuilles ($t = m^n$, $m \geq 2$, $n \geq 1$).

$$\Xi_{u,v}^t = \max_{\substack{k_1 + \dots + k_v = u \\ (k_1, \dots, k_v) \in \{2, \dots, t\}^v}} \{ \xi_{k_1}^t + \dots + \xi_{k_v}^t \}, \quad u \in \{2v, \dots, tv\}, \quad v \in \mathbb{N}^*.$$

La fonction Ξ vérifie l'équation suivante :

$$\Xi_{u,v}^t = \sum_{i=0}^{v-1} \xi_{2 \left\lfloor \frac{\lfloor u/2 \rfloor + i}{v} \right\rfloor}^t - (u - 2 \lfloor u/2 \rfloor); \quad \Xi_{u,1}^t = \xi_{2 \lfloor u/2 \rfloor}^t - (u - 2 \lfloor u/2 \rfloor).$$

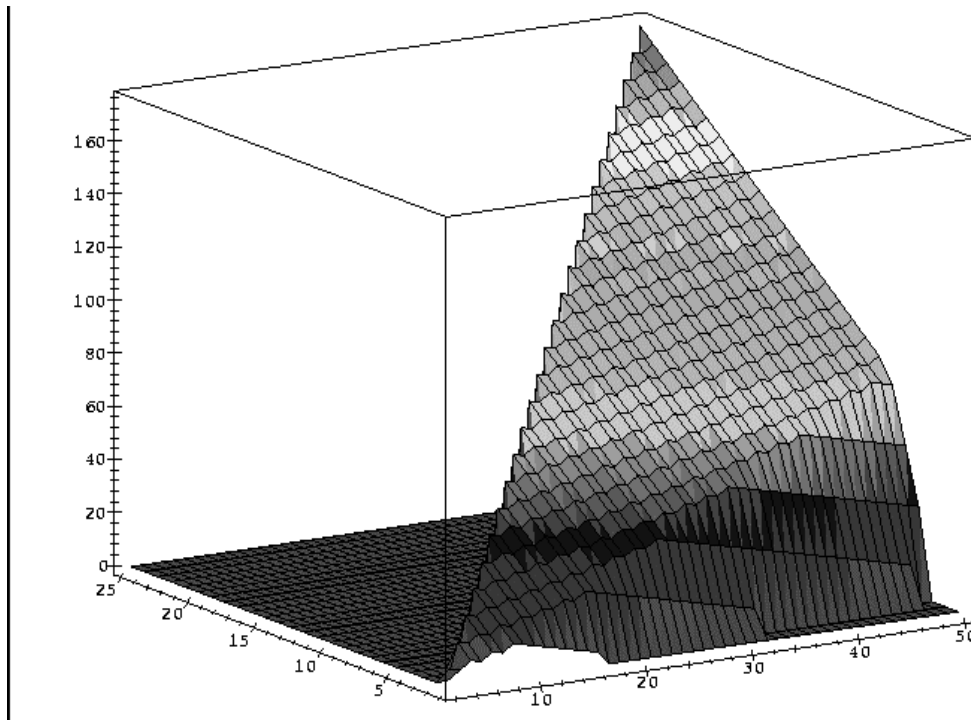
$$\Xi_{u,v}^t = v \tilde{\xi}_{u/v}^t = \tilde{\xi}_u^{tv} - \frac{v-1}{m-1}.$$



Les conditions de faisabilité - la fonction Ξ (2/2)

La forme close de la fonction Ξ :

$$\Xi_{u,v}^t = v \frac{m^{\lceil \log_m(\frac{m}{v} \lfloor \frac{u}{2} \rceil) \rceil} - 1}{m - 1} + m \lfloor \frac{u}{2} \rfloor \left[\log_m \left(\frac{t}{\frac{m}{v} \lfloor \frac{u}{2} \rfloor} \right) \right] - \left(u - m \lfloor \frac{u}{2} \rfloor \right).$$



- $t = 16 = 4^2$:
arbre quaternaire ;
- $u \in \{2, \dots, 50\}$,
 $v \in \{1, \dots, 25\}$.



Conclusions (1/3)

A l'issue de ce travail, j'ai acquis la conviction d'avoir étudié les paradigmes qui correspondent à de nombreuses réalités opérationnelles, notamment :

- ✧ les modèles de transactions distribuées, manipulant des objets modifiables rémanents distribués sur l'ensemble desquels sont définis des invariants globaux ;
- ✧ les échéances strictes de terminaison ;
- ✧ les modèles événementiels arbitraires (unimodal, multimodal).

La composabilité (de modules de traitement, de modules de communication) en présence de dépendances globales, qui est chère aux industriels, est rendue difficile par le choix de ces paradigmes.



Conclusions (2/3)

J'ai dû examiner des problèmes analytiques difficiles, notamment ceux posés par l'établissement de conditions de faisabilité en pire cas. Mon travail est, je pense, novateur. Il permet de mieux voir comment on peut poser et résoudre des problèmes relevant du domaine combiné TR/TD/TF, encore très peu exploré.

Les algorithmes étudiés sont hybrides : combinaison d'algorithmes (en-ligne) d'ordonnancement temps réel centralisés ou distribués, d'algorithmes de consensus / de contrôle de concurrence distribués, d'algorithmes pour la tolérance aux fautes. Le choix des « bonnes » combinaisons reste un problème ouvert.



Conclusions (3/3)

Les travaux de recherche à venir sont multiples. Les principaux me semblent être l'identification de combinaisons algorithmiques optimales, l'analyse de combinaisons algorithmiques valides dans les modèles de calcul partiellement synchrones, l'établissement de conditions de faisabilité suffisantes optimales pour une complexité donnée.



Publications

Les travaux consignés dans cette thèse ont donné lieu à cinq publications (actes des conférences internationales ETFA, PDCS, ICDCS et ISORC).

Objectifs à court terme (publications dans des journaux) :

- ✧ une version détaillée du papier ISORC '99
(co-auteurs : P. Carrère, J.-F. Hermant et G. Le Lann)
The Journal of Real-Time Systems
- ✧ une version détaillée du papier ICDCS '98
(co-auteurs : Jean-François Hermant et Gérard Le Lann)
The Journal of Real-Time Systems
- ✧ une publication sur l'ordonnancement temps réel centralisé
(co-auteurs : J.-F. Hermant, L. Leboucher et N. Rivierre)

