L'approche trajectorielle pour les systèmes temps-réel

Alain Jean-Marie
LIRMM

Journées AS01
Automatique et réseaux de communication
18 octobre 2002

Plan

1. Introduction: Temps réel, réseaux et automatique
2. L'approche «Network Calculus»
3. Le modèle trajectoriel pour le calcul de temps de réponse
4. Directions de recherche à explorer

Introduction: Temps réel, réseaux et automatique

Le lien entre les réseaux de communication et les systèmes temps-réel

- LAN: il a toujours existé (réseaux de terrain)

- WAN: il est plus fort depuis la confluence des réseaux de données et les réseaux «téléphoniques»
  - ATM: multiplexage de trafics avec des contraintes différentes
  - Internet + Qualité de Service

  Il se résume par: Garantir des délais.

Temps réel et réseaux

Progression de l'idée dans les réseaux de télécommunication:

Compétition entre deux « philosophies »

réservation de ressources $\leftrightarrow$ multiplexage statistique

garanties de service $\leftrightarrow$ meilleur effort

ATM $\leftrightarrow$ Internet

IntServ $\leftrightarrow$ DiffServ

Temps réel et automatique/1

Satisfaire les contraintes de délai nécessite un contrôle plus rigoureux des flux d'information.

Approche du problème par des outils de l'automatique: Chang 1992 (reprenant des idées de Cruz 1990) puis Le Boudec & Thiran 1998

- flot d'information comme signal
- éléments de réseau comme « boites noires »
- composition d'éléments: séries, parallèles, feedbacks
- contrôle actif sur le flot: mise en forme du trafic

Temps réel et automatique/2

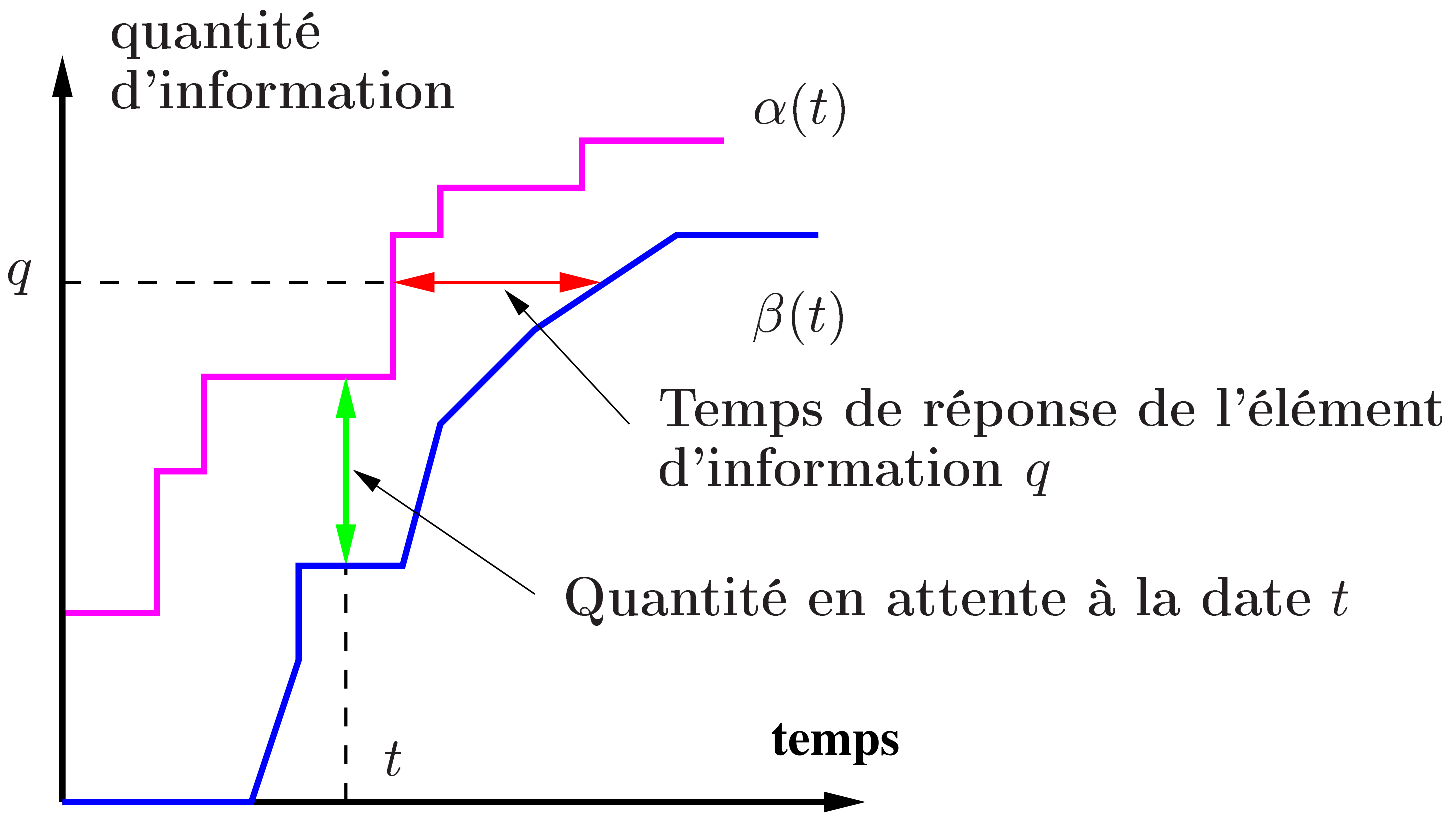
Problème: garantir la faisabilité d'un ensemble de tâches.

- Approche « évaluation de performances »
  - fixer l'algorithme d'ordonnancement
  - calculer le temps de réponse maximum
  - comparer avec les échéances

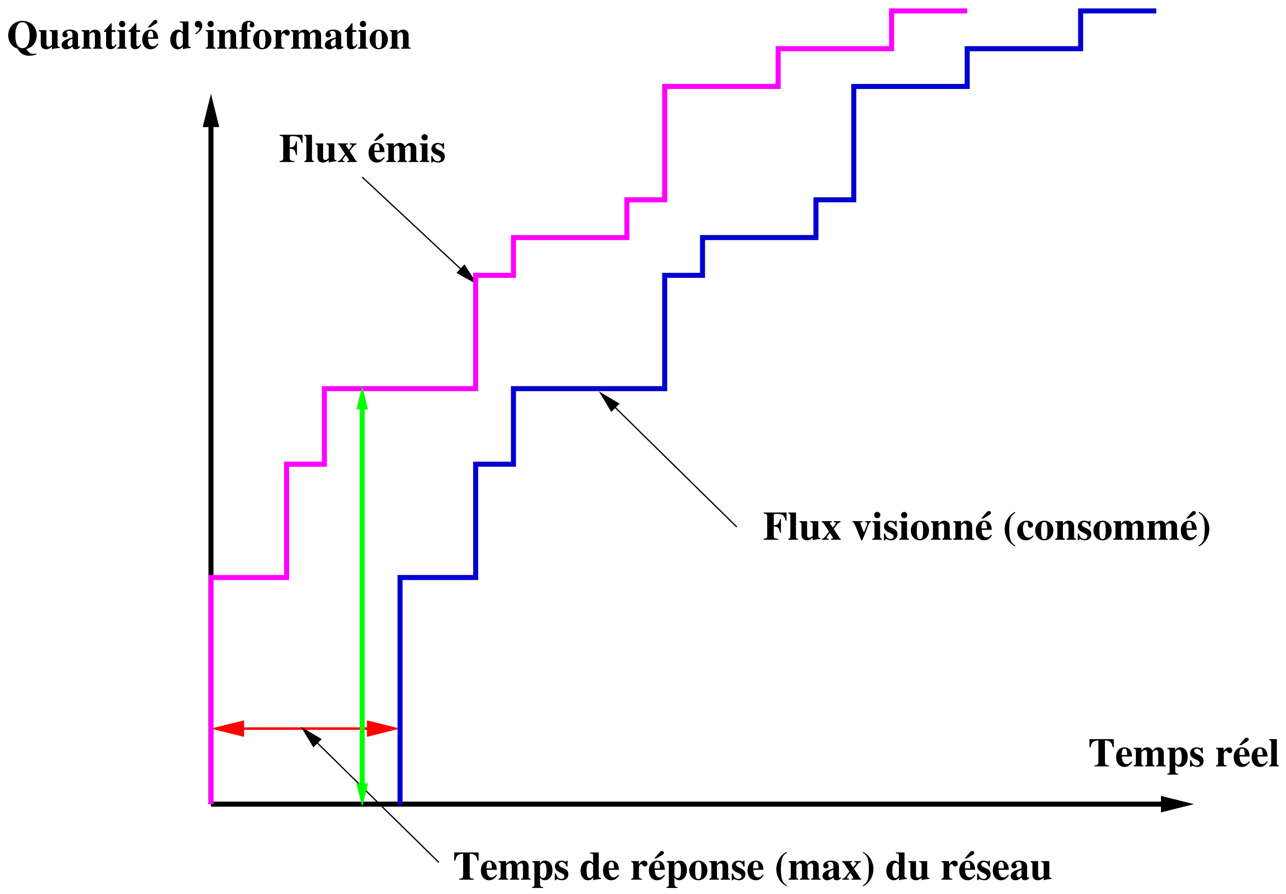
- Approche « contrôle » (et ordonnancement)
  - se donner une information sur le processus de tâches
  - se donner une classe de « politiques »
  - se donner un critère de coût (retard maximum)
  - trouver la politique qui minimise le critère

Calcul des réseaux (version Le Boudec-Thiran)

Représentation du trafic: courbes d'arrivées et courbes de service



Exemple: réseau à délai constant (ex: video)



Éléments de Calcul des Réseaux

Ingrédients:

- Temps discret (Chang) ou temps continu (Le Boudec-Thiran)
- $A = \{A(t)\}$, où $A(t)$ quantité totale de paquets arrivée à la date t
- A est α-contrainte si: pour tout s:

$$\sup_n \{A(n+s) - A(n)\} \ \leq \ \alpha(s) \ .$$

- Un système offre le service β si la quantité servie dans un intervalle de durée s est $\geq \beta(s)$.

Garanties de performance:

Soit un système

- offrant la courbe de service $\beta(\cdot)$
- soumis à des arrivées α-contraintes

Alors,

- la taille de la file d'attente est bornée par:

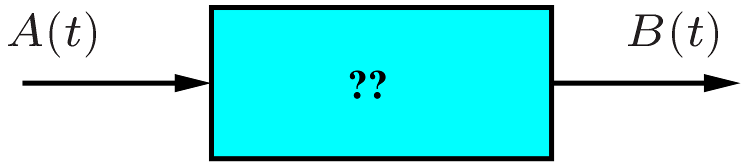
$$W(t) \leq \sup_{0 \leq s \leq t} \{\alpha(s) - \beta(s)\}$$

- le délai des paquets est borné par

$$D(q) \leq \alpha^{-1}(q) - \beta^{-1}(q) .$$

Régulation de trafic

Q: Soit f une fonction (sous-additive). Comment rendre le trafic f contraint?



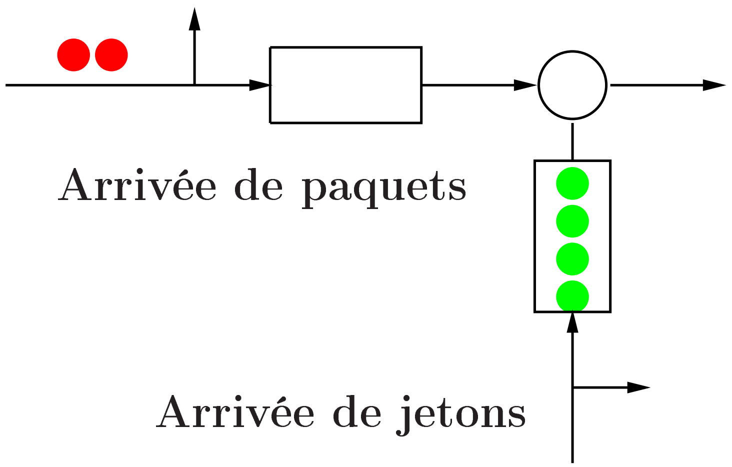
R: En construisant:

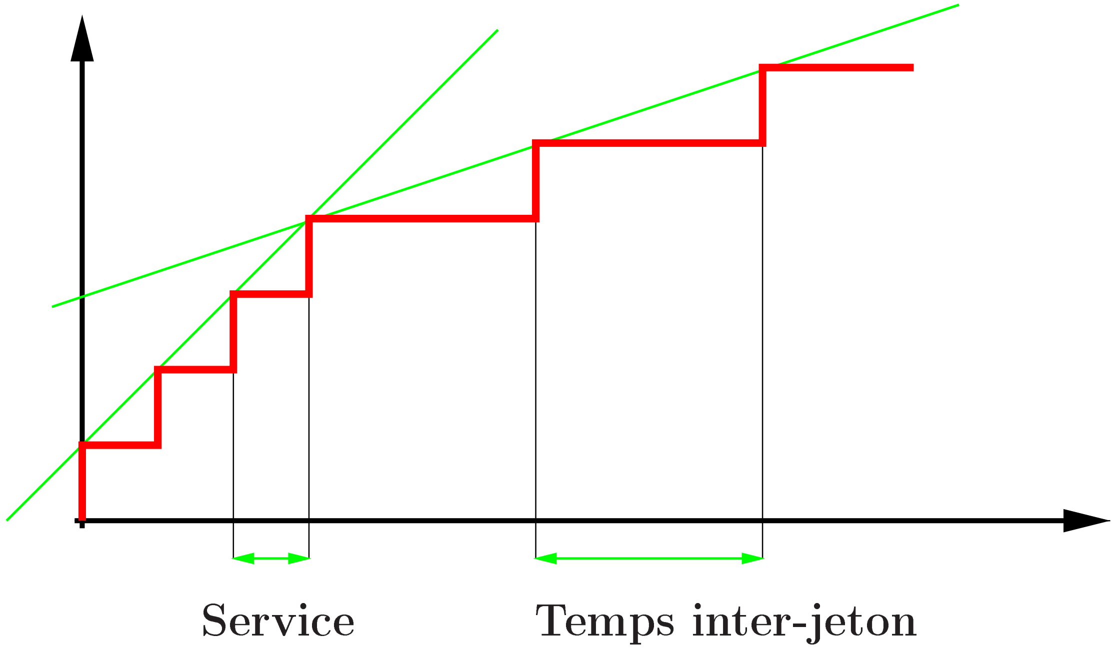
$$B(t) = \min_{0 \leq s \leq t} \{A(s) + f(t-s)\}$$

C'est une min-convolution: $B = A \star f$.

Exemple de régulateur: le seau à jetons

Le seau à jetons (Token bucket):





Le cas multiclasse

Cas des trafics « (σ, ρ) »:

$$\sup_n \{A(n+s) - A(n)\} \ \le \ \sigma \ + \ \rho \ s \ .$$

Multiplexage

Si on superpose des trafics d'arrivées $A^1, \ldots, A^N$, on a une contrainte (σ, ρ) avec:

$$\sigma \ = \ \sum_{i=1}^{N} \sigma_i \ , \qquad \rho \ = \ \sum_{i=1}^{N} \rho_i \ .$$

Routage

Soit une fonction de filtrage (ou routage): $r(n) \in \{0, 1\}$, telle que:

$$\sup_n \{r(n+s) - r(n)\} \ \leq \ \delta \ + \ \pi \ s \ .$$

Si on filtre un trafic (σ, ρ) par la fonction de routage r, on obtient un trafic contraint par:

$$\sigma' \ = \ \pi\sigma \ + \ \delta \qquad \rho' \ = \ \pi\rho \ .$$

Résultat de la combinaison: σ a tendance à augmenter au cours de la traversée d'un réseau.

Le Modèle Trajectoriel pour le temps réel

Synthèse de travaux de S. Lefebre-Barbaroux (92), C. Chaouiya (94) et J. Migge (99).

Mis en œuvre lors de conventions avec Thomson ASM et EDF CCC.

Peu avancé depuis la réunion STS (GdR ARP) du ... 11/12/98.

Contexte:

- ensemble de tâches récurrentes (infinité d'*instances*)
- une unité centrale
- une politique d'ordonnancement
- des dates butoir

Objectif:

- calculer le temps de réponse maximum des instances des tâches
- ou au moins des bornes supérieures
- afin de tester la faisabilité

Les approches traditionnelles du problème

Approche « instant critique » Liu & Leyland 73

- On exhibe la configuration «la pire» des arrivées de tâches
- On calcule le temps de réponse maximal
- Manipulations de fenêtres temporelles, et principe de point fixe
- ... et on recommence si le modèle des tâches et/ou la politique de service change

Approche « model checking »

- On part d'une spécification du système et des contraintes temporelles
  Réseau de Petri, logique temporelle, langages synchrones...

- On génère un automate temporisé

- On vérifie qu'aucune violation de contrainte n'intervient

- ... en gérant l'explosion combinatoire

Utilisation de principes de la théorie des files d'attente

- séparation du processus d'arrivée de travail du mécanisme de service
- utilisation du processus de quantité de travail
- prise en compte de toutes les trajectoires possibles
- notion de période d'activité (par classe)
- bornes trajectorielles, couplages

Résultats

- preuves rigoureuses (!) pratiquement automatisables (?)
- résultats appliquables immédiatement à de nouveaux processus d'arrivée de tâches
- formalisme de représentation des politiques d'ordonnancement

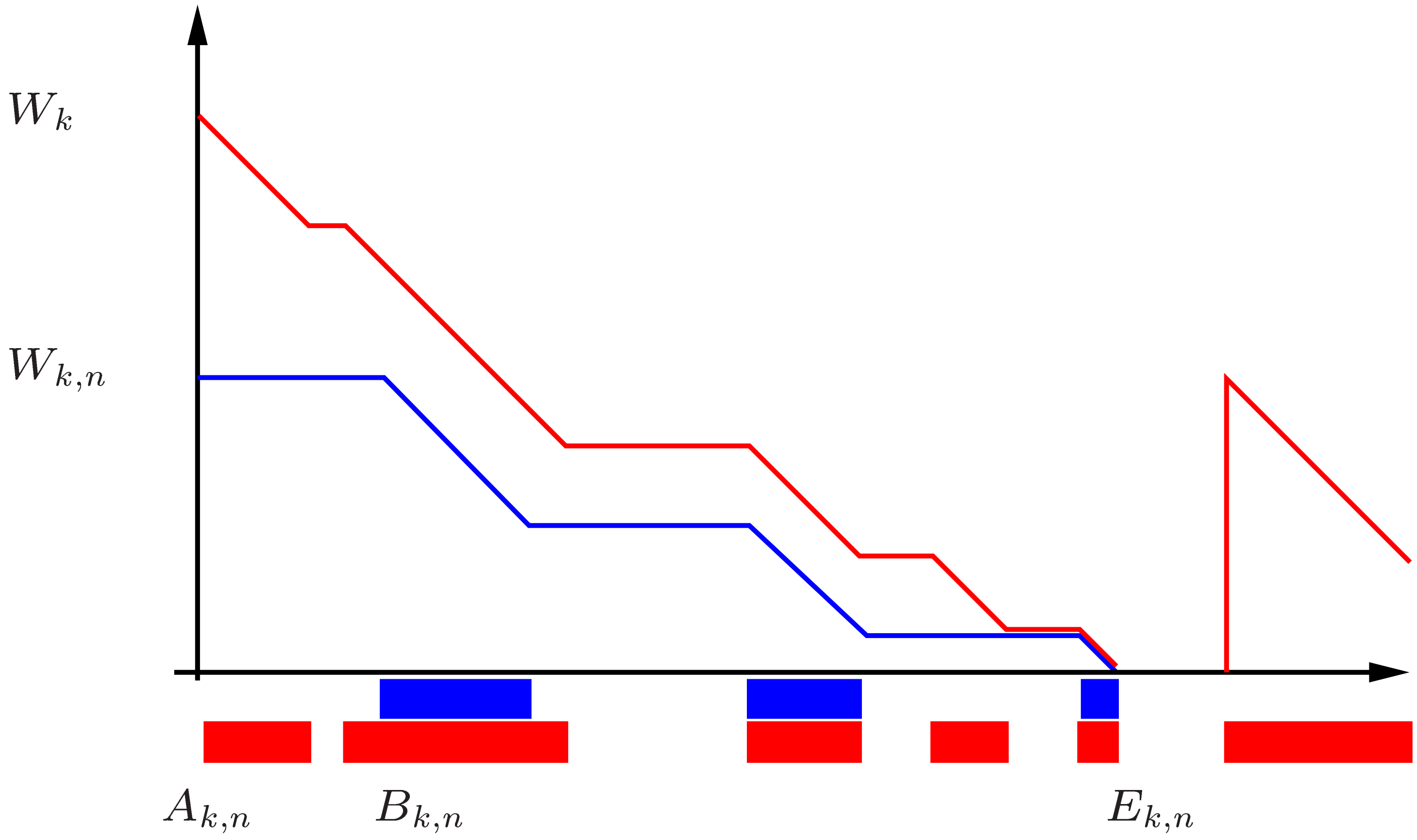
Notations...

- $\tau_{k,n}$: instance n de la tâche k
- $A_{k,n}$: date d'arrivée de $\tau_{k,n}$
- $C_{k,n}$: temps de service requis par $\tau_{k,n}$
- $D_{k,n}$: échéance relative de $\tau_{k,n}$

Toutes ces quantités peuvent varier $\Longrightarrow$ ensemble de trajectoires $\omega \in \Omega$.

- $B_{k,n}$: date de début d'exécution
- $E_{k,n}$: date de fin d'exécution

Processus de charge



Politiques d'ordonnancement

Une description formelle de ce qu'est une politique d'ordonnancement

- À chaque instance $\tau_{k,n}$, un vecteur de priorité: suite finie de réels, dépendant du temps

$$\Gamma_{k,n}(t) \in \bigcup_{n \geq 1} \mathbb{R}^n .$$

- Les vecteurs sont ordonnés dans l'ordre lexicographique $\leq$

- Unique politique de service: plus haute priorité d'abord

Exemples:

- Priorité préemptive fixe (FPP)

$$\Gamma_{k,n} = (k, n)$$

- Première échéance d'abord (EDD/EDF)

$$\Gamma_{k,n} = (D_{k,n}, k, n)$$

$\Longrightarrow$ $D_{k,n}(t)$, Échéance basée sur Courbe de Service (Service Curve Earliest Deadline) $D_{k,n}$ dernier instant de «conformité»

- FIFO

$$\Gamma_{k,n} = (A_{k,n}, k, n)$$

- LIFO préemptif

$$\Gamma_{k,n} = (-A_{k,n}, k, n)$$

- SRPT

$$\Gamma_{k,n}(t) = (W_{k,n}(t), k, n)$$

- Promotion de priorité

$$\Gamma_{k,n}(t) \;=\; \begin{cases} P_{k,n} & \textsf{si } t < B_{k,n} \\ Q_{k,n} & \textsf{si } t \geq B_{k,n} \end{cases}$$

$\Longrightarrow$ toutes les politiques non préemptives

- Priorités en couches

$$\Gamma_{k,n} \;=\; (\ell_1, \Gamma'_{k,n})$$

- et même Round Robin (Migge & Navet 2002), « priority ceiling », critical sections, etc...

Construction des trajectoires

Conditions pour l'existence d'une unique trajectoire solution:

- Pas de point d'accumulation dans les dates d'arrivée
- L'ensemble des vecteurs $\{\Gamma_{k,n}\}$ est décidable:

$$\tau_{k,n} \neq \tau_{k',n'} \quad \text{et} \quad W_{k,n} > 0,\ W_{k',n'} > 0 \qquad \Longrightarrow \Gamma_{k,n} \neq \Gamma_{k',n'}$$

- Les vecteurs préservent l'ordre par intervalle:

  pour tout $s < t$: l'ordre entre $\Gamma_{k,n}(u)$ et $\Gamma_{k',n'}(u)$ s'inverse un nombre fini de fois entre s et t.

Calcul de bornes

Soit $\mathcal{H}_{k,n}$ l'ensemble des instances plus prioritaires que $\tau_{k,n}$:

$$\mathcal{H}_{k,n} \;=\; \{\tau_{i,j} \mid \Gamma_{i,j} \geq \Gamma_{k,n}\} \;\cup\; \{\tau_{k,n}\}$$

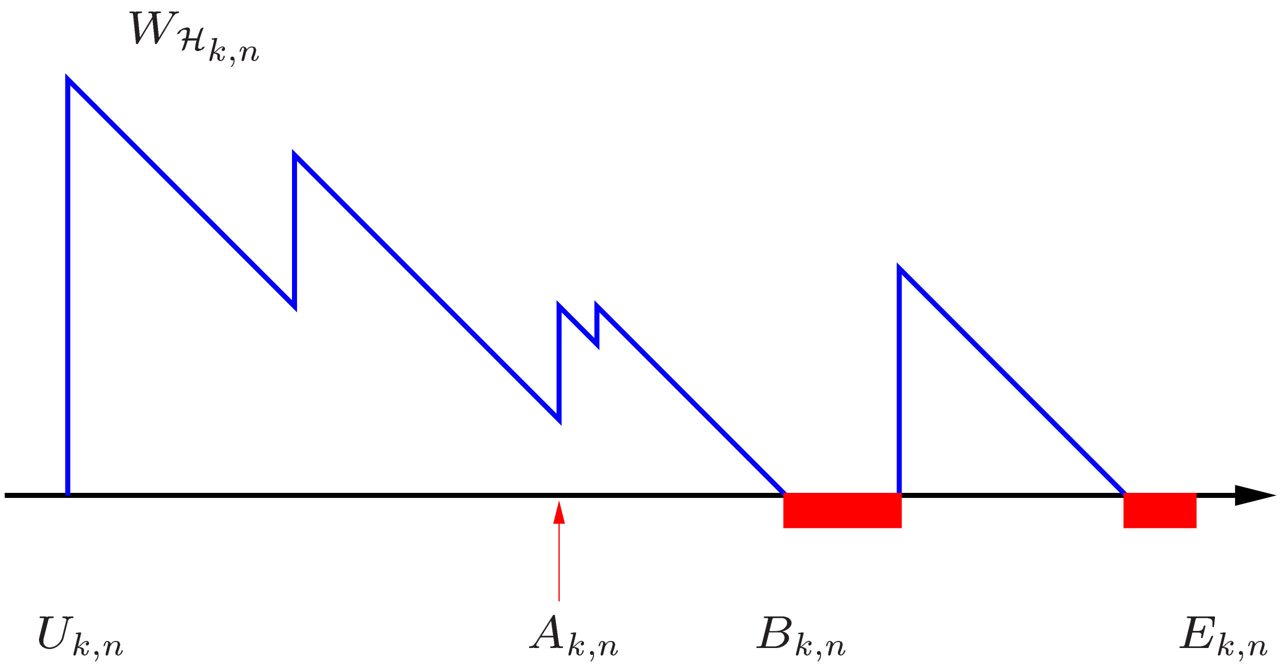
Alors:

$$\begin{aligned} E_{k,n} &= \min\{t > A_{k,n} \mid W_{k,n}(t) \;=\; 0\} \\ &= \min\{t > U_{k,n} \mid S_{\mathcal{H}_{k,n}}(U_{k,n}, t) \;=\; t - U_{k,n}\} \end{aligned}$$

$U_{k,n}$: date de début de la période d'interférence de $\tau_{k,n}$.

$S_{\mathcal{H}_{k,n}}(a, b)$ est la fonction d'arrivée de travail des instances de $\mathcal{H}_{k,n}$.

Période d'interférence et temps de réponse:



Temps de réponse maximal

Si on remplace S par un majorant $\widehat{S}$, on obtient

$$\hat{E}_{k,n} \;\geq\; E_{k,n}\ .$$

D'où la méthode:

1. Étant donnée une description d'un ensemble de tâches, trouver des fonctions $\widehat{S}_k$ telles que:

$$S_k(t, s+t; \omega) \;\leq\; \widehat{S}_k(s)$$

2. Elles correspondent à un processus de tâches $(\widehat{A}, \widehat{C}, \widehat{D})$

3. Appliquer la formule du temps de réponse. Pour FPP, on a:

$$\max_{\omega} \max_{n} R_{k,n}(\omega) \;\leq\; \max_{q} \max_{j} \widehat{R}_{k,j}(\omega)$$

où j est tel que $\widehat{U}_{k,j}(q) = 0$.

4. Donc, simuler la trajectoire du système sur la première période d'interférence et enregistrer le temps de réponse maximal.

Directions de recherche

- Analyse probabiliste (entamée pour FPP)
- Extension au cas de réseaux

  \+ Analyse Holistique de Tindell

  − Réseaux « réentrants » instables (Bramson, Dai, Weiss)

  $\Longrightarrow$ avancer le Network Calculus multiclasse

- Systèmes fermés

$$A_{n+1} = \max\{E_n, \; A_n + T_n\}$$

- Lien avec les méthodes formelles

Bibliographie

Network Calculus

R. Cruz, «A calculus for network delay», part I & II, *IEEE Trans. Information Theory*, vol. 37-1, pp. 114–131 & 132–141, jan. 1991.

C.S. Chang, *Performance Guarantees in Communication Networks*, Springer Verlag, 2000.

J.-Y. Le Boudec et P. Thiran, *Network Calculus – A theory of deterministic queueing systems for the Internet*, LNCS 2050, Springer Verlag, 2001.

Analyse «au pire cas» pour le Temps Réel (quelques références)

C. Chaouiya, S. Lefebvre-Barbaroux et A. Jean-Marie, «Real-Time Scheduling of Periodic Tasks», *in* Scheduling Theory and its Applications, John Wiley, 1995. Rapport INRIA RR-1576 (jan 1992).

J. Migge, A. Jean-Marie et N. Navet, «Timing analysis of compound scheduling policies: application to Posix1003.1», *The Journal of Scheduling*, 2002.

J. Migge, *Ordonnancement sous contraintes temps-réel: un modèle à base de trajectoires*, Thèse de doctorat, Université de Nice-Sophia Antipolis, 1999. Rapports INRIA RR-3561 (nov 1998) et RR-3678 (avr 1999).

K.W. Tindell et J. Clark, «Holistic schedulability analysis for distributed hard real-time systems», *Microprocessors and Microprogramming*, mar. 1994.

M. Spuri, «Holistic Analysis for Deadline Scheduled Real-Time Distributed Systems», INRIA RR-2873, (avr. 1996).