

Algorithmique du Network Calculus

Le Network Calculus

- Formulaire de calcul de garanties déterministes sur les performances des réseaux de communications.
- Principe: avec les spécifications limites d'un système (p.ex. garanties locales sur le service fourni ou sur le trafic), calculer des bornes sur les performances (p.ex. délais ou charge) en se ramenant à des systèmes (min,+) linéaires.
- Calcul de bornes de pire cas → sûreté de systèmes critiques (avions, voitures, satellites), optimisation de QoS dans des réseaux à trafic très contrôlé.

Les objets

- Caractéristiques du système modélisées par des courbes.
- Courbes d'arrivées (pour le trafic) :
 - bits reçus entre 0 et t
 - temps t
 - $A(t)$
 - A est α -contraint si $\forall s \leq t, A(t) - A(s) \leq \alpha(t-s)$
- Courbes de service min. (du système) :
 - bits servis entre 0 et t
 - temps t
 - $S(t)$
 - S est un β -serveur si $\forall t, B(t) \geq \inf_{0 \leq s \leq t} A(s) + \beta(t-s)$
- Lecture des performances :
 - charge max
 - délai max
 - flux α -contraint dans un β -serveur.

Les opérations

- Analyse du système en combinant les courbes avec les opérations suivantes :
- Addition: $(f + g)(t) = f(t) + g(t)$
 - Minimum: $\min(f, g)(t) = \min(f(t), g(t))$
 - Convolution: $(f \otimes g)(t) = \inf_{0 \leq s \leq t} (f(s) + g(t-s))$
 - Déconvolution: $(f \oslash g)(t) = \sup_{s \geq 0} (f(t+s) - g(s))$
 - Clôture: $f^*(t) = \inf_{n \geq 0} f^{(n)}(t), f^{(n)} = \underbrace{f \otimes \dots \otimes f}_n$
- Exemple :
-
- $a_2 = (a_1 \otimes (\beta_1 \otimes \beta_2))^*$

Objectifs

- Identifier des classes de fonctions permettant des calculs effectifs.
- Développer des algorithmes pour les opérations du Network Calculus et les analyser.
- Implanter les algorithmes, en vue d'une intégration à Scilab.
- A plus long terme, développer un outil d'analyse de performances reposant sur le Network Calculus (prenant en compte la topologie du réseau et ses caractéristiques locales, les contraintes imposées au trafic, les politiques de service).

Algèbre (min,+)

Algorithmique du Network Calculus

Systèmes à événements discrets

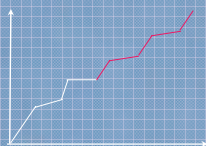
Géométrie algorithmique

Etat de l'art

- Théorie développée depuis les années 90 (Chang, Cruz, Le Boudec, Thiran), comprenant des travaux de formalisation, de modélisation et de mathématiques mais pratiquement pas d'algorithmique.
- Analogies avec la théorie classique (+,x) des systèmes peu exploitables.
- Quelques logiciels existent (Disco), mais font des calculs partiels ou ont des restrictions à des cas très particuliers.
- Librairies (max,+) de Scilab pour le cas discret avec des fonctions à valeurs et arguments entiers (par manipulation de séries formelles).

Premiers pas

- Une classe stable pour toutes les opérations du Network Calculus : les fonctions affines par morceaux, ultimement pseudo-périodiques, c.à.d.
 - $\exists T, c, d > 0, \forall t > T, f(t+d) = f(t) + c.$
- Premiers algorithmes pour cette classe en cours d'analyse et en cours d'implantation.



Pistes

- Exhiber des bases de fonctions permettant de décomposer les courbes et accélérer les calculs.
- Mettre au point des transformations simplifiant les calculs (comme la transformée de Legendre-Fenchel pratiquée pour les fonctions convexes).
- Calculs exacts vs. calculs approchés.
- Liens avec la géométrie algorithmique (problèmes d'arrangement de courbes).
- Questions de complexité, voire de décidabilité.
- Questions de stabilité numérique.

Questions connexes

- Étendre le formalisme : étendre les règles de combinaisons ou introduire de nouvelles opérations pour intégrer de nouvelles politiques de services.
- Optimisation : comment optimiser le dimensionnement et/ou le contrôle du réseau pour garantir une bonne qualité de service ?
- Analyse de trafic réel : comment calculer efficacement la courbe d'arrivée d'un trafic réel à partir d'un échantillon ou à la volée ?

Participants : Laurent Jouhet et Eric Thierry

