Algorithmes de contrôle de réseaux à hauts débits

Philippe Jacquet
INRIA Rocquencourt

April 26, 1993

[résumé par Philippe Jacquet]

1. Introduction : les réseaux à hauts débits

Une problématique des réseaux à hauts débits proches de la norme ATM est introduite. Les composants rapides qui sont directement en contact avec le support optique du réseau présentent des capacités d'intégration limitées : à hautes vitesses, faibles capacités de mémoire. Au dessus de ces composants très rapides se trouvent les composants ordinaires de la station connectée : grandes capacités de mémoire pour des vitesses modérées.

Un paquet est constitué d'un nombre variable de cellules de taille fixe qui sont les composants élémentaires des données qui circulent sur le réseau. Les cellules trouvent leur place sur des slots.

Au cours de la réception d'un paquet dans une station, le composant rapide procède au transfert des cellules du réseau vers un buffer rapide (faible capacité). Le composant lent transfère les cellules des buffers rapides vers des buffers lents (mais à grande capacité). En phase d'émission, le processus est inverse : le composant lent dépose les cellules dans les buffers rapides et le composant rapide assure le transfert des buffers au réseau lui même.

La faible capacité des buffers rapides liée à la lenteur des composants à grande capacité constitue le maillon critique du système et nécessite des algorithmes de contrôle de flux originaux. Cet exposé résume un rapport de recherche de Paul Mühlethaler et Philippe Jacquet.

2. L'algorithme de contrôle de flux

La problématique des réseaux à haut débit introduit la nouveauté suivante : les délais de propagation sont trop importants pour permettre le blocage des stations pendant l'aller et retour d'une simple requête d'émission. L'algorithme proposé permet à la station de continuer à transmettre entre deux requêtes. Les délais de propagation d'aller et retour sont supposés identiques entre chaque paire de stations comme c'est le cas sur les architectures en anneau.

Algorithme proposé :

(1) chaque fois qu'une station a un paquet à transmettre elle envoie une cellule de requête à l'adresse de la destination ; la cellule de requête contient la durée prévue de la transmission du paquet entre sa première et sa dernière cellule (nombre de cellules multiplié par l'intervalle inter-cellulaire) ; en attendant le retour de la requête le paquet reste en buffer lent ;

(2) la station source sépare chacune de ses requêtes, sans regard de leurs destinations, d'un intervalle de temps de durée au moins égale au temps présumé de transmission des paquets annoncé ;

(3) à chaque requête reçue, la station destinatrice renvoie vers la source une cellule de réponse contenant la valeur d'un compteur appelé compteur de transmission ; ensuite, sans regard sur la provenance de cette requête, la station destination augmente son compteur de transmission de la quantité contenue dans la cellule de requête ;

(4) la destination décrémente son compteur à chaque unité de temps : le compteur de transmission s'identifie à la longueur d'une file d'attente virtuelle ;

(5) la station source commence la transmission de son paquet (c'est-à-dire le dépile de ses buffers lents) à la date d'échéance c'est-à-dire la date de réception de la cellule de réponse plus la valeur du compteur de transmission qu'elle contient.

Le paquet reste stocké dans la mémoire de masse de la source (c'est-à-dire au niveau du composant lent) jusqu'à la date d'échéance. À la date d'échéance, le composant lent procède au transfert des cellules vers la mémoire du composant rapide. La fréquence de ce transfert en fait détermine la durée de l'intervalle inter-cellulaire, qui est supposé être identique pour toutes les stations du réseau. La variation introduite par les problèmes d'accès multiple au niveau du médium physique est négligée.

Comme la source ne s'abstient pas de transmettre pendant le délai d'acheminement de la requête, d'autres échéances peuvent expirer entre temps, de façon à ce que la station soit déjà en phase de transmission de paquet lorsqu'expire la dernière date d'échéance. Dans ce cas le dernier paquet reste en mémoire de masse et gagne une file d'attente. Ce genre de collision entre deux phases de transmission simultanées n'est susceptible d'avoir lieu que lorsque au moins deux destinations différentes sont concernées. La file d'attente ci-dessus décrite est appelée *file de sortie*.

La file d'attente dont il s'agit d'optimiser la longueur est la mémoire du composant rapide de la destination, appelée *file d'entrée*. Pour chaque paquet les cellules arrivent dans cette mémoire à la même fréquence qu'elles en sont extraites : c'est-à-dire la vitesse de transfert du composant lent. Donc si un paquet est unique en phase de réception ses cellules n'occuperont pas de manière significative la mémoire rapide. Si plus de deux paquets arrivent, leurs cellules cumuleront une fréquence d'arrivée double de la fréquence d'extraction et la file d'attente d'entrée s'allongera d'au moins un paquet. C'est ce genre de collision en file d'arrivées que l'algorithme cherche à éviter.

Exemples : Supposons qu'une source envoie tous ses paquets vers une seule destination. Les différents compteurs de transmission reçus n'entrent jamais en collision et la file de sortie reste toujours vide. Si toutes les sources d'une station visent cette station comme destinataire unique, il n'y a collision ni sur les files de sortie ni sur la file d'entrée.

3. Les files d'attente de La Palice : définition et propriétés

Le modèle mathématique qui convient à l'algorithme de contrôle de flux ci-dessus décrit est le modèle des files d'attente de La Palice. Une file de La Palice est une file classique FIFO, avec un serveur en général unique, dans laquelle les clients arrivent selon une loi particulière au modèle :

(1) chaque client a un temps de service S et une date de rendez-vous, S est une variable aléatoire ;
(2) les dates de rendez-vous de deux clients successifs sont séparées par un laps de temps de durée supérieure ou égale au temps de service du premier des deux clients ;
(3) chaque client arrive dans la queue avec un retard spécifique D, D est une variable aléatoire.

Ces files d'attente de La Palice (introduites dans [1] et [2]) ont des propriétés intéressantes. Le paramètre intéressant est le délai d'attente W en file ou la charge Q. W et Q sont des variables aléatoires (respectivement associées au client et au temps).

Une première propriété (la plus évidente, origine de l'attribution du nom de La Palice) est que si le retard D est identiquement nul ou constant, alors il n'y a pas de file d'attente à proprement parler : W et Q sont identiquement nuls.

Une seconde propriété un peu moins évidente est que si D et S sont des variables aléatoires bornées, par exemple respectivement par Δ et Σ, alors W et Q sont des variables aléatoires bornées par $\Delta + \Sigma$.

La dernière propriété, de loin la moins évidente, suppose l'indépendance de S et D plus quelques conditions de stabilité de la loi d'arrivée des clients, comme par exemple : l'écart entre deux dates de rendez-vous est en moyenne strictement supérieure à la moyenne du temps de service. La propriété nécessite aussi l'introduction d'une nouvelle terminologie. Une variable aléatoire est sous-exponentiellement Ax^α si $-\log \Pr\{X > x\}$ est supérieur à Ax^α quand $x \to \infty$. Dans la file de La Palice stabilisée, si le temps de service S est sous-exponentiellement Ax^α et le retard D est sous-exponentiellement Bx^β, avec A, B, α et β supérieurs à zéro, alors W et Q sont sous-exponentiellement Cx^γ, pour certains C et γ explicites :

(i) si $\alpha > \beta$ et $\alpha > 1$ alors $\gamma = \beta + 1 - \beta/\alpha$ et $C = \frac{(A\alpha)^{1/\alpha}}{\gamma}(B(1-1/\alpha))^{1-1/\alpha}$;
(ii) si $\alpha < \beta$, alors W et Q s'alignent sur le temps de service S avec $\gamma = \alpha$ et $C = A$;
(iii) si la variable aléatoire S est bornée par une constante Σ (équivalent à $\alpha = \infty$), alors $\gamma = \beta + 1$ et $C = \frac{B}{(\beta+1)\Sigma}$.

Le point (iii) sera utilisé pour une modélisation de l'algorithme de contrôle de flux.

4. Algorithme de contrôle de flux et files de La Palice

Les files de sortie et d'entrée peuvent être vues comme des files de La Palice.

La file de sortie : les temps de service S des paquets sont leur durée de transmission. Les paquets ont des dates de rendez-vous qui sont les dates de retour des requêtes : comme les requêtes sont séparées de plus du temps de service S et que les délais de propagation sont tous identiques, les dates de rendez-vous obéissent au modèle de La Palice. Le retard D est égal à la valeur du compteur de transmission reçu dans la cellule de réponse. On appelle W le délai d'attente du paquet dans la file de sortie.

La file d'entrée : Le temps de service est toujours S mais la date de rendez-vous est maintenant la date d'échéance, c'est-à-dire la date de retour de requête plus la valeur du compteur de transmission. Le retard est le délai W subi dans la file de sortie.

Dans le modèle qui suit on suppose un cas le pire : les destinations des sources et les sources des destinations sont complètement indépendantes et les trafics sont uniformément distribués. Il est intuitif qu'une destination ou une source plus favorisée ne peut qu'améliorer les performances de l'algorithme comme l'illustrent déjà les exemples décrits plus haut. Cette indépendance des sources et destinations fait que les paquets subissent dans la file de sortie des retards D indépendants, et dans la file d'entrée, des retards W indépendants.

Nous faisons l'hypothèse simplificatrice de paquet de durée égale à une unité de temps. Soit λ le taux d'arrivée des paquets par unité de temps. Les compteurs de transmission se comportent comme des longueurs de files d'attente M/D/1 et ont donc un comportement exponentiel : $\Pr\{D > n\} \sim \lambda^n$.

En d'autres termes D est sous-exponentiel $(-\log\lambda)x$. En conséquence l'analyse des files d'attente de La Palice dans le cas S borné par l'unité (cas (iii)) nous donne W sous-exponentiel $\frac{-\log\lambda}{2}x^2$. En fait une analyse plus précise peut nous donner $\Pr\{W > n\} \sim \lambda^{n(n+1)/2}$.

Dans la file d'entrée, le retard W étant maintenant sous-exponentiel $\frac{-\log\lambda}{2}x^2$, alors la charge Q de la file (exprimée en paquets stockés) est sous-exponentielle $\frac{-\log\lambda}{6}x^3$. Une analyse plus spécifique conduit à $\Pr\{Q > n\} \sim \lambda^{n(n+1)(n+2)/6}$. Noter le gain appréciable par rapport à λ^n que l'on aurait sans contrôle de flux. Une variante déterministe existe dans laquelle on recommence toute requête dès que le délai W dépasse un certain seuil constant. Dans ce cas on applique la propriété numéro deux des files de La Palice.

Bibliographie

[1] Jacquet (Philippe). – *More than exponential tail distribution in La Palice queues.* – Research Report n° 1465, Institut National de Recherche en Informatique et en Automatique, 1991.

[2] Jacquet (Philippe) et Mühlethaler (Paul). – *A very simple algorithm for flow control on high speed networks via La Palice queueings: description and analysis.* – Research Report n° 1371, Institut National de Recherche en Informatique et en Automatique, 1991.