

Vous devez écrire un simulateur implémentant la technique décrite dans ce document ainsi qu'un rapport. Le rapport contiendra une brève description du problème (1/4 du rapport max), une brève description de ce que vous avez implémenté (1/4 du rapport max) ainsi qu'une évaluation des performances du protocole étudié. La troisième séance de TP sera consacrée aux projets. Le travail est à effectuer par groupe de 4 maximum. Le travail est à rendre pour le 11 mars.

Consignes :

- Le simulateur est à écrire dans un langage de votre choix parmi C/Java/C++ (si vous préférez le faire dans un autre langage, merci de me le demander).
- Vous devez rendre une archive contenant un **Makefile** ou des **instructions simples** permettant de compiler votre code. Le code doit compiler du premier coup. Un code qui ne compile pas vaut 0.
- Le rapport devra faire **4 pages maximum** en police 10pts (-2pt par page supplémentaire).
- Le rapport est à rédiger dans le format de votre choix (latex, openoffice,...). Ne rendre qu'une version pdf.

Quelques remarques :

- Les différentes questions posées (sur les graphiques ou l'évaluation de performance) ne sont qu'une indication pour vous aider à rédiger votre rapport. Vous êtes encouragés à vous poser d'autres questions et à critiquer vos résultats.
- La limite de pages doit vous pousser à être clair et concis et à faire des choix sur les résultats que vous souhaitez présenter.
- Penser à justifier les choix que vous avez faits et à discuter les résultats (est-ce le bon indicateur ? les intervalles de confiance sont-ils bons ? ...)
- Penser à inclure des graphiques clairs et informatifs (voir la *checklist for good graphics*).
- Environ 2/3 des points sera consacré au rapport, 1/3 au code lui-même (fonctions implémentés, qualité et clarté du code,...)

Projet : TCP

Au protocole TCP est souvent associé un mécanisme de contrôle de congestion : chaque lien essaie de gérer son débit afin d'utiliser au mieux les ressources. Dans ce projet, on va plus particulièrement s'intéresser à la modélisation d'un protocole de type AIMD (additive increase, multiplicative decrease).

Dans la suite, on s'intéresse au modèle suivante :

- Un ensemble de N connexions TCP essaient d'envoyer des paquets au même serveur.
- Chaque connexion possède un état d qui représente le débit auquel elle envoie ses paquets (on considère qu'elle a un nombre infini de paquets) : une connexion dans l'état d envoie un paquet tout les $1/d$ secondes. Les paquets mettent un temps aléatoire pour arriver au serveur (distribué selon une loi exponentielle de moyenne 100ms).
- Quand le serveur reçoit un paquet, il le met dans un buffer de taille fini Q . Si le buffer est plein le paquet est perdu. Le serveur sert les paquets dans l'ordre d'arrivée dans le buffer. Le temps de service d'un paquet est distribué selon une loi exponentielle de moyenne 1ms.
- Quand le serveur traite un paquet, il envoie un message ACK à la connexion dont est originaire le paquet. Cet ACK mets un temps aléatoire pour parvenir à la connexion originale (distribué selon une loi exponentielle de paramètre 100ms).

Le protocole AIMD que l'on veut étudier est le suivant :

- Quand une connexion envoie un paquet, elle considère que si elle n'a pas reçu de ACK au bout de 1 seconde (time-out), il est perdu.
- Tous les 200ms, elle augmente son débit de $a > 0$ (additive increase) sauf si elle a détecté une perte de paquet dans les 200ms dernières, dans ce cas elle multiplie son débit par $b < 1$ (multiplicative decrease).

Exercice 1. Simulation

Écrire un simulateur du protocole décrit ci dessus. Pour cela, on pourra écrire un simulateur à événement discret où l'état du système est :

- L'état $d_1 \dots d_N$ de toutes les connexions.

- Une trace des paquets en circulation pour chaque connexion (afin de détecter les pertes).
- L'état de la file d'attente du serveur.

Les différents événements sont :

- Une connexion envoie un paquet.
- Une connexion augmente ou diminue son débit.
- Le serveur reçoit un paquet.
- Le serveur traite un paquet (et envoie un ACK).
- Un ACK est réceptionné.
- Un time-out doit être déclenché.

Le simulateur devra posséder une fonction permettant de sortir l'état global du système (par exemple débit moyen de toutes les connexions et nombre de paquets dans la file) ainsi qu'une fonction permettant de sortir l'état d'une connexion particulière.

Pour commencer, on pourra prendre les valeurs suivantes. N'hésitez pas à les modifier.

- Taille de la file : $Q = 1000$.
- Additive increase : $a = 1$.
- Multiplicative decrease : $b = 2$.
- Time-out : $t_o = 1s$.

Exercice 2. Quelques Courbes

Afin de vous orienter dans votre étude du système, vous pourrez commencer par tracer les courbes suivantes.

a. L'évolution du nombre de paquets en attente dans le système en fonction du temps ainsi que du débit total envoyés par les connexions en fonction du temps. Quel est le débit efficace du système ?

b. Évolution de l'état d'une connexion donnée au court du temps : voyez vous des motifs apparaître ? Est-ce au contraire assez aléatoire ?

c. Si le temps moyen de traitement d'un paquet par le serveur passe de 1ms à .1 ms (le serveur va 10 fois plus vite), le système est-il réactif ? Même question si le temps de traitement passe de 1ms à 10ms ?

d. Pour une connexion particulière, quel est la quantité de perte d'un paquet en fonction de différents paramètres du système (a , b ou Q selon vos choix).

e. On s'intéresse à "l'équité" dans le système (toutes les connexion arrivent à envoyer le même nombre de paquets). Pour cela, on peut par exemple se fixer un intervalle de temps et tracer la distribution du nombre de paquets effectivement traités par le serveur pour chaque connexion.

Exercice 3. Évaluation de performances

Vous êtes libres de vous inspirer des courbes tracées précédemment pour appuyer votre étude du système. Voici quelques idées dont vous pouvez vous inspirer (mais n'hésitez pas à vous poser d'autres questions).

- Le protocole vous semble-t-il efficace (quantité de pertes raisonnables, délai par paquet)
- Le système est-il réactif aux changements ?
- Il y a-t-il des choix de a , b qui vous semblent clairement mieux ou moins bien que d'autres ?