

Projet de PRS

La Paire de Sockets

Rémy HIDRA, Claire LAVERNE

ANALYSE DES CLIENTS

Scénario 1

Drop beaucoup de paquets.

Attend un délai faible aléatoire avant de renvoyer un ACK.

Scénario 2

Drop peu de paquets.
Délai plus long.

Scénario 3

Plusieurs clients 1.
Les stratégies du scénario 1 seront sans doute adaptées.



SCÉNARIO 1 – Stratégie et ordre d'implémentation

- ▣ Fiabilité : attente basique de l'ACK du message envoyé
- ▣ Taille des données maximale
- ▣ Fenêtre statique : envoi de rafales de paquets, puis attente de tous les ACKs
- ▣ Fenêtre dynamique : rafales de tailles variables, en fonction de la qualité du réseau
- ▣ Envois agressifs : écoute ponctuelle des ACKs reçus, afin de détecter des duplicates ACKs éventuels

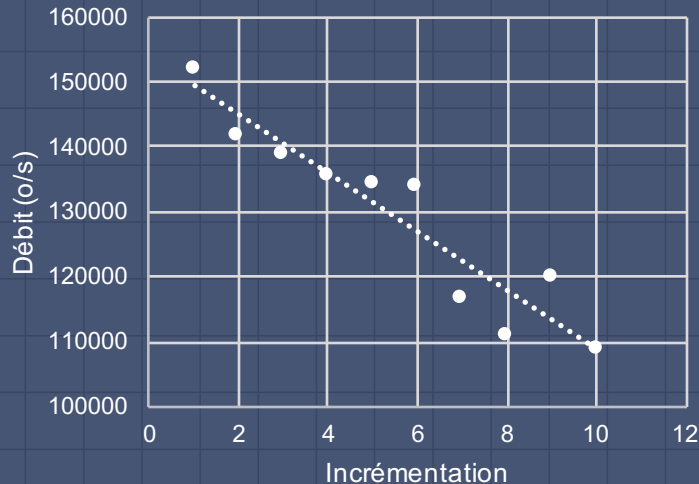
SCÉNARIO 1 – Algorithme

- Envoi d'une rafale de taille *cwnd*
- Attente d'un ACK
- Adaptation de *cwnd* en fonction de l'ACK
- Recul dans le fichier et dans le numéro de séquence, afin de reprendre la transmission au dernier paquet reçu

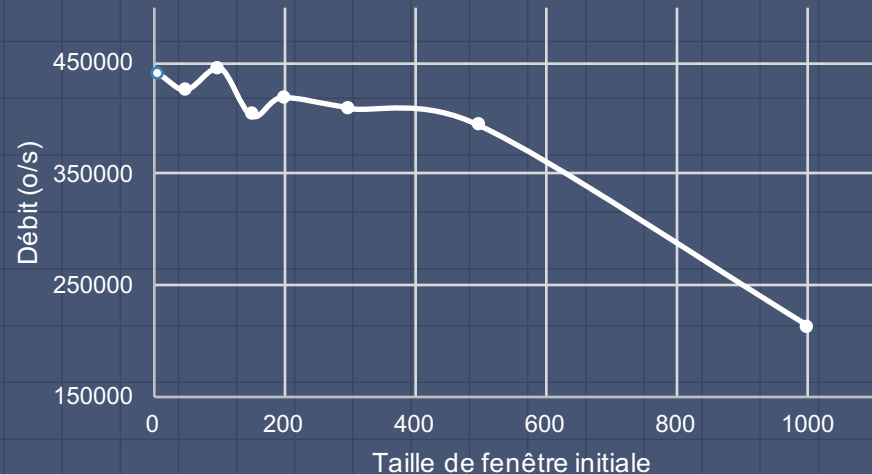
SCÉNARIO 1 – Mesures des constantes optimales

- Détermination de la meilleure incrémentation et taille initiale de la fenêtre

Débit en fonction de l'incrément



Débit en fonction de la taille de fenêtre initiale



SCÉNARIO 2 – Stratégie & Ordre d'implémentation

- ▣ Reprise du serveur 1
- ▣ Réception de la totalité des ACKs, afin de détecter le plus vite possible un paquet perdu
- ▣ Renvoi de toute une fenêtre qui n'a pas été reçue

SCÉNARIO 2 – Algorithme

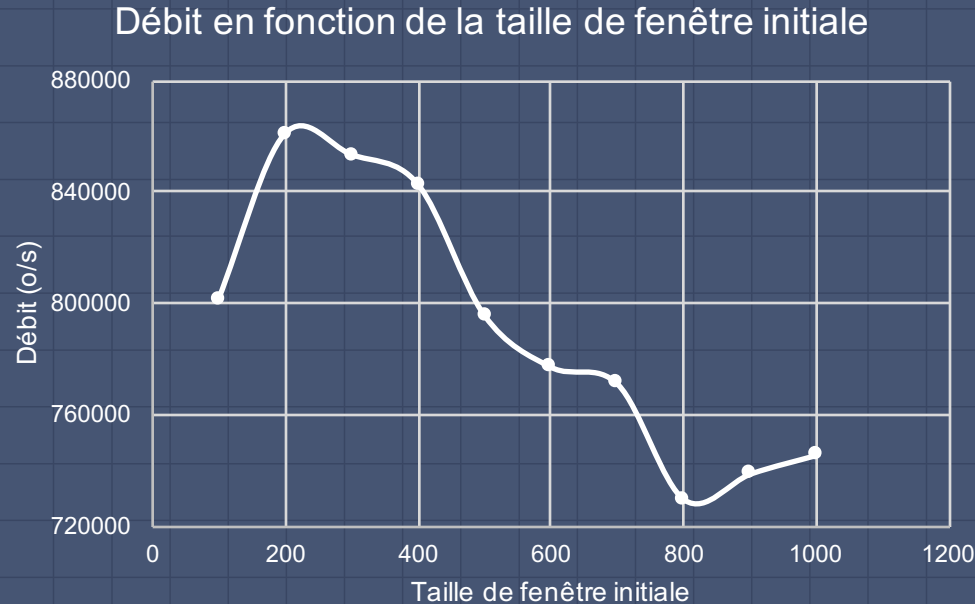
- Envoi d'une rafale de taille *cwnd*
- Attente de la totalité des ACKs de la fenêtre ou timeout
- Adaptation de *cwnd* en fonction de l'ACK



SCÉNARIO 2 – Mesures des constantes optimales

8

- Détermination de la meilleure taille de fenêtre initiale



SCÉNARIO 2 – Retour sur notre travail & Améliorations

- ▣ Envoi de rafales d'un même paquet, pour éviter que le client le drop (échec)
- ▣ Adaptation du timeout de réception des ACKs, en fonction du RTT
- ▣ Fast retransmit pour ne retransmettre que les paquets nécessaire

SCÉNARIO 3 – Stratégie

- ▣ Reprise du serveur 1
- ▣ Implémentation d'une gestion multi-threadée pour supporter les connexions multiples simultanément



Résultats

- ▣ Serveur 1: 1,03 Mo/s
- ▣ Serveur 2: 0,87 Mo/s
- ▣ Serveur 3: 4,18 Mo/s

