

# Projet 1 - Rapport

Antoine Gennart

October 20, 2018

## 1 Introduction

Dans le cadre du cours LINGI1341 : *Computer networks : information transfer*, il nous a été demandé de réaliser un protocole de transfert fiable basé sur des segments UDP.

Ce rapport décrit l'architecture globale du projet, et réponds aux questions imposées dans le cadre du projet.

## 2 Architecture

- **sender.c** Implémente la fonction principale du sender.
- **receiver.c** Implémente la fonction principale du receiver.
- **pkt.c** Implémente toute la structure des packets, ainsi que toutes les fonctions pour manipuler ces packets.
- **network.c** Implémente la connexion UDP entre deux machines.
- **utils.c** Contient un certain nombre de fonction utiles et qui n'avaient pas leur place dans les autres fichiers.
- **fifo.c** Implémente la mémoire FIFO (*first in first out*) qui va être utilisé par le receiver pour gérer les *acknowledgment*

## 3 Questions

### 3.1 Que mettez-vous dans le champs Timestamp, et quelle utilisation en faites-vous?

Le champs timestamp contient un nombre de millisecondes. Il est calculé comme suit :

$$\text{timestamp} = t_{epoch} \% (2^{32} - 1)$$

ou  $t_{epoch}$  correspond au temps écoulé depuis l'époque en millisecondes et  $\%$  représente l'opération modulo.

### 3.2 Comment réagissez-vous à la réception de paquets PTYPE\_NACK?

Je ne réagis pas, j'ignore le packet.

### 3.3 Comment avez-vous choisi la valeur du retransmission timeout?

La valeur du *retransmission timeout* est calculé via l'algorithme de *Van Jacobson*. Comme décrit dans le cours (p. 159 du syllabus), il est initialisé de la manière suivante :

$$\begin{aligned} srtt &= rtt \\ rttvar &= \frac{rtt}{2} \\ rto &= srtt + 4 \cdot rttvar \end{aligned}$$

Lorsque d'autres mesures du rtt arrivent, nous pouvons mettre à jours les valeurs de rtt, srtt et rto comme suit:

$$\begin{aligned} rttvar &= (1 - \beta) \cdot rttvar + \beta \cdot (srtt - rtt) \\ srtt &= (1 - \alpha) \cdot srtt + \alpha \cdot rtt \\ rto &= srtt + 4 \cdot rttvar \end{aligned}$$

Dans le cadre du projet, j'utilise les valeurs conseillées dans le syllabus, à savoir  $\alpha = \frac{1}{8}$  et  $\beta = \frac{1}{4}$ . Ces valeurs particulières permettent de remplacer des *floating point operations* par des *bits shifts*, ce qui est beaucoup moins coûteux pour le processeur.

### **3.4 Quelle est la partie critique de votre implémentation, affectant la vitesse de transfert?**

Il s'agit de l'attente de l'acknowledgement.

### **3.5 Quelles sont les performances de votre protocole?**

### **3.6 Quelles sont les stratégies de test que vous avez utilisées?**

Pour le moment la stratégie de test consiste à envoyer et recevoir un fichier, pour ensuite comparer le fichier d'entrée avec le fichier de sortie. Si ces deux fichiers sont différents, le test échoue.