

# Projet IN407

## STRATÉGIE DE GESTION DE FLUX

Ce projet a été réalisé par Dimitri Walter et Sam Khong Lucas Rasamizany issu respectivement du groupe de TD 2 et du groupe de TD 3 en L2 Informatique à l'UVSQ. Ce document vise à être un compte rendu du travail réalisé par rapport au projet en expliquant l'architecture de notre application et son fonctionnement.

### Introduction

Dans le cadre de notre projet, nous avons développé une application de gestion de flux pour un réseau de communication à commutation de paquets. Ce compte rendu vise à présenter l'architecture de l'application, ainsi que son fonctionnement détaillé. Pour cela, nous aborderons la structure du code, le rôle de chaque classe et module, et nous expliquerons le fonctionnement de l'interface utilisateur.

### Architecture de l'application

L'application est principalement constituée de plusieurs classes et modules interconnectés, chacun jouant un rôle spécifique dans la simulation du réseau de communication. Voici un aperçu de la structure du code :

- **main.py** : Ce fichier contient la classe principale App, qui représente l'interface graphique de l'application. Il fait usage du module customtkinter pour personnaliser les widgets graphiques.
- **buffer.py** : Ce module définit la classe Buffer, qui modélise les buffers de stockage des paquets dans le réseau. Chaque buffer a une capacité limitée et peut perdre des paquets s'il est plein.
- **paquet.py** : Ce module contient la classe Paquet, qui représente un paquet de données transitant à travers le réseau. Chaque paquet est associé à un temps d'arrivée et contient des informations telles que l'adresse source, l'adresse de destination et la taille.

- **source.py** : Ce module définit la classe Source, qui génère des paquets de manière aléatoire avec des attributs tels que l'adresse IP source et de destination, ainsi que la taille du paquet.

## Fonctionnement de l'interface utilisateur

L'interface utilisateur de l'application est conçue pour permettre à l'utilisateur de configurer et de visualiser la simulation du réseau de communication. Voici une explication détaillée des fonctionnalités de l'interface :

- **Configuration des paramètres** : L'utilisateur peut configurer les paramètres de simulation tels que le taux d'arrivée des paquets à la source (`source_lambda`) et le taux d'arrivée des paquets dans le buffer principal (`buffer_attente_lambda`). Ces paramètres déterminent la fréquence à laquelle de nouveaux paquets sont générés dans le réseau.
- **Choix de la stratégie de gestion de flux** : L'interface propose plusieurs stratégies de gestion de flux, notamment "Max capacités", "Tour de rôle", "Random" et "Simple". Chaque stratégie définit comment les paquets sont traités et transmis dans le réseau.
- **Visualisation des statistiques** : L'interface affiche en temps réel les statistiques globales telles que la capacité, les pertes, les transmissions et les réceptions pour chaque élément du système (source, buffers, destination). Ces statistiques permettent à l'utilisateur de suivre l'état du réseau pendant la simulation.
- **Contrôle de la simulation** : L'utilisateur peut démarrer, redémarrer ou remplir les buffers avant de lancer la simulation. Ces contrôles permettent de préparer le réseau et de démarrer la simulation à tout moment.

## Fonctionnement de la simulation

Une fois que l'utilisateur a configuré les paramètres et choisi une stratégie de gestion de flux, la simulation démarre en fonction de la stratégie sélectionnée. Voici comment chaque stratégie fonctionne :

- **Max capacités** : Les paquets sont envoyés dans le buffer avec la capacité la plus élevée. Si la destination est disponible, les paquets sont transmis. Sinon, ils sont stockés dans le

buffer.

- **Tour de rôle** : Un paquet est pris de chaque buffer à tour de rôle et transmis à la destination.
- **Random** : Les paquets sont envoyés dans un buffer choisi au hasard. Si la destination est disponible, les paquets sont transmis.
- **Simple** : Un paquet est envoyé de la source à un buffer, puis de ce buffer à la destination.

## Conclusion

En conclusion, notre application de gestion de flux offre une interface conviviale pour simuler le fonctionnement d'un réseau de communication à commutation de paquets. Grâce à sa structure modulaire et à ses différentes stratégies de gestion de flux, elle permet à l'utilisateur de mieux comprendre les défis liés à la transmission des données dans un tel environnement.