

## TP2 - Étude de paramètres de performances

### But du TP

Le but de ce TP est de travailler sur les notions suivantes :

- (1) le calcul de la gigue (pas vu en cours) ;
- (2) le taux d'occupation d'un buffer.

### Rappel - Utiliser ns-2 en salle de TP

Pour lancer une simulation en ns-2 sur votre poste :

1. Composer votre script ns avec votre éditeur de texte favori et sauvegarder le dans un fichier portant l'extension .tcl
2. Exécuter le. Pour cela, il suffit de saisir dans un `xterm` la commande suivante :  
`ns nomDuProgramme.tcl` (ou `ns < nomDuProgramme.tcl`)
3. Accéder au fichier contenant la trace de la simulation (fichier `out.tr`)

Assurez vous que la variable d'environnement `LC_NUMERIC` est bien fixée à '.'. Si ce n'est pas le cas, exécutez la commande `export LC_NUMERIC='.'` dans votre terminal courant. Vous pouvez également forcer POSIX comme standard pour votre terminal avec la commande `export LC_ALL=POSIX`.

### Exercice - Calcul de la gigue (1h30)

La gigue est une quantité souvent utilisée dans les réseaux informatiques, notamment lorsqu'on s'intéresse à la qualité de service (QoS). La gigue est une quantité temporelle qui permet d'estimer la variabilité du délai de bout-en-bout. Plus précisément, la gigue du paquet  $n$ ,  $g_n$  peut s'exprimer comme :  $g_n = d_n - d_{n-1}$  où  $d_n$  désigne le délai de bout-en-bout du paquet  $n$ . Ainsi, la gigue d'un paquet mesure le temps que le paquet a mis en plus ou en moins que le paquet précédent du même flux pour traverser le réseau.

On considère la topologie décrite sur la figure 1. **Fixer la taille de chaque buffer à 10.** Attention, il vous faut 6 lignes de commandes pour cela (cf. Annexe).

#### Travail demandé :

1. La gigue d'un paquet est-elle une quantité positive ou négative ?

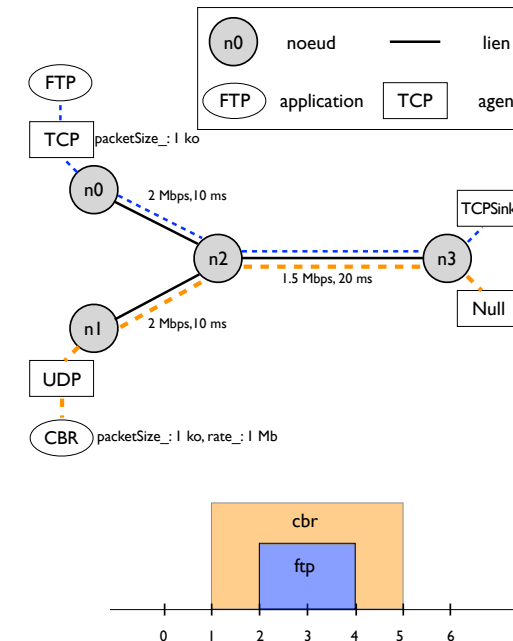


FIGURE 1 – Expérience à simuler pour l'exercice Calcul de la gigue.

2. Pour quels types d'applications la gigue est une quantité importante à connaître ?
3. Composer le script ns permettant de réaliser cette expérience. Vérifier que votre expérience est correcte en vous assurant que l'instant de génération du paquet avec le numéro de séquence 211 d'UDP a lieu à l'instant 2.688s.
4. Commencer par composer un script awk calculant la gigue de chaque paquet UDP (en prenant en compte le temps d'attente sur le premier nœud). Indiquer la gigue pour les paquets UDP de numéro de séquence 134 et 135.
5. Compléter votre script pour calculer la valeur moyenne de la gigue sur les paquets UDP de numéros de séquence compris entre 0 et 120 inclus, puis entre 125 et 300 inclus. **Attention : ne calculer la gigue d'un paquet que si ce paquet ET celui le précédant ont bien été reçus.** Vous pouvez vérifier que votre script retrouve bien les valeurs trouvées à la question 3 pour les paquets 134 et 135. Quelles valeurs trouvez-vous pour ces 2 moyennes ? Conclure quant-à l'utilité de calculer une telle quantité.
6. Exprimer littéralement la formule pour la gigue du paquet 1 en fonction des délais de bout-en-bout des paquets 0 et 1. Puis faire de même pour les paquets 2 et 3. En déduire l'expression de la gigue moyenne pour les 3 premiers paquets, puis pour les  $n$  premiers

paquets. Que remarquez-vous ?

7. Proposer une autre façon plus pertinente de calculer une valeur moyenne de la gigue ?
8. (Question subsidiaire) Quelle valeur trouvez-vous ?

## Exercice - Mesure de l'occupation du buffer (1h30)

Dans cet exercice, on souhaite représenter sur un graphique l'évolution du taux d'occupation (nombre de paquets en attente) dans le buffer. Pour cela, on considère la topologie de la figure 1.

La taille du buffer correspond au nombre maximal de paquets en attente (incluant le paquet en cours d'émission si un paquet est en train d'être émis). **Attention, sous ns-2, lorsqu'un buffer est plein, tout nouveau paquet cherchant à entrer est d'abord accepté puis immédiatement rejeté.**

Travail demandé :

1. Reprendre le script ns de l'exercice précédent en vous assurant que la taille de chaque buffer est bien égale à 10 (cf. Annexe).
2. On s'intéresse au comportement de la file dans le buffer du nœud  $n_2$  vers  $n_3$ . Plus précisément on souhaite obtenir l'ensemble de ses changements d'état ainsi que leur instant. Editer un script awk afin de générer un fichier texte avec une nouvelle ligne à chaque changement d'état de la file indiquant le temps courant et la taille courante de la file. **Astuce : utiliser un simple espace comme délimiteur.**
3. Représenter l'évolution de cette variable dans un graphe en fonction du temps (cf. Annexe sur Gnuplot).
4. Que peut-on observer ? Comment expliquer la présence de plusieurs régimes pendant la période où le flux CBR et le flux TCP cohabitent ?
5. On s'intéresse à présent à la taille moyenne de la file. Calculer la taille moyenne sur l'ensemble de la simulation, c'est-à-dire de 0 à 5s. Puis entre 2s et 4s. **Attention, les intervalles de temps entre deux changements de taille de la file ne sont pas nécessairement de longueur identique.**

## Annexes

### A - Fixer la taille des buffers

nœud Par défaut, dans ns-2, les buffers sont de taille 50 et de type "Drop Tail".

Pour changer la taille d'un buffer, on utilise l'instruction : `$ns queue-limit $nx $ny L` où L où représente la taille du buffer. Attention, il s'agit du buffer de  $n_x$  vers  $n_y$  (et pas celui de  $n_y$  vers  $n_x$ ).

Remarque : pour observer visuellement avec NAM le comportement de la file du buffer entre  $n_x$  et  $n_y$ , utiliser la commande `$ns duplex-link-op $nx $ny queuePos 0.5`

## B - Tracer un graphe sur Gnuplot

Pour représenter sur un graphe l'évolution d'une variable, on peut utiliser `gnuplot`. Il suffit alors de saisir les commandes suivantes sur un terminal `gnuplot` ou d'exécuter directement un script `gnuplot <script.gnu>` :

```
# Exemple de script gnuplot
# commentaire après un #
# le fichier "plot_file.txt" contient les valeurs à tracer
# trace la courbe (colonne 2 en fonction de la colonne 1) avec style 1 et titré "courbe 1"
plot 'plot_file.txt' using 1:2 title 'courbe 1' with lines ls 1

# ajoute une seconde courbe (colonne 3 fonction de colonne 1) avec style 2 et titré "courbe 2"
replot 'plot_file.txt' using 1:3 title 'courbe 3' with lines ls 2

# légendes des axes
set xlabel 'temps'
set ylabel 'fenêtre de congestion'
replot

# pour sauvegarder le graphe dans un fichier postscript
set term png
set output 'mon_graphe.png'
replot
```