

Rapport Tp 2 : Analyse de donn  

I. utilisation de l'application :

Pour lanc   l'application de fa  on la plus simple possible:

```
make
```

```
./analyse -f « fichier de donn   »
```

sa aura pour effet d'afficher tout les infos sur le fichier. Les donn  e seront   crites dans le dossier donn  e et les graphes dans le dossier graph.

On peut lanc   l'application avec 3 option suppl  mentaire :

- -tab « fichier du tableau de d  bit » permet le calcul de la taille du paquet sinon la taille sera toujours nulle.
- -pid « pid » « pid » « pid » ... permet l'affichage des donn  es calcul   pour les paquets
- -fid « fid » « fid » « fid » ... permet l'affichage des donn  es calcul   pour les flux

toute les options et -f peuvent   tre mis dans n'importe quelle ordre.

II. R  sultat de l'application :

On r  cup  re plusieurs information apr  s traitement.

Pour les donn  es global on a :nbr routeur, nbr flux ,nbr paquets, nbr perdu, taux de perte, moy_bb(moyen temps bout    bout), moy_timewait (moyenne du temps d'attente dans une fils), moy_timetrans(moyen du temps de transmission).

Toutes les moyennes des donn  es globales sont donn  es avec leur intervalle de confiance(ic) et leur   cart type(ec).

On a aussi des information sur chaque n  ud :

n  noeud = num  ro du n  ud

moy temps d'att(intervale de confiance) = moyennes du temps d'attente dans les filles d'attente dans le noeud.

taux de perte = taux de perte dans le noeud

nbr de perte = nombre de paquet perdu dans le n  ud

nbr d  part = nombre de paquet qui a comme d  part se n  ud

nbr arriv   = nombre de paquet qui a pour arriv   se n  ud

nbr   mis = nombre de paquet envoy   par le n  ud

nbr re  us = nombre de paquet re  ue par le n  ud

Ainsi que de l'information pour chaque flux qu'on a demand   en argument

fid = identifiant du flux

nbr   mis = nombre de paquets envoy   dans le flux

nbr re  us = nombre de paquet re  ue    l'arriv  

nbr lost = nombre de paquets perdu

temps debut = a quelle moment le flux s'ouvre
 temps fin = a quelle moment le flux se fini
 temps de vie = temps de vie du flux
 moy temps d'acheminement(ic) = moyennes du temps que prend un paquet a arrivé a destination.

Et pour finir pour les paquets :

pid = identifiant
 source = nœud de départ
 dest = nœud de destination
 t envoie = le moments ou le paquets a été envoyé sur le réseau
 t fin = le moment ou il arrivé a destination ou a été supprimé
 taille = taille du paquet (0 si pas de tableau de débit)
 moy temps d'att = moyenne que le paquet passe dans une fils d'attente
 moy temps de transmtion = moyenne du temps que prend le paquet pour être transmit d'un nœud a l'autre
 noeud parcourue = liste des nœuds parcourues

On a aussi des graphe qui sont tracé dans le fichier graph :

nbr_de_paquet = nombre de paquet créer dans le réseau en fonction du temps
 nbr_de_paquet_dans_les_files_d'att = nombre de paquets qui sont dans toutes le file d'attente
 nbr_paquet_dans_le_réseau = nombre de paquets dans le réseau aux moment t
 nbr_perdu = nombre de perte dans le réseau.
 node_wait = nombre de paquet qui attend par noeud
 temps_bout_a_bout = temps bout a bout des paquets.

III.exemple et interprétation

on lance :

```
>>time ./analyse -f trace2650.txt -tab res26.txt -pid 5900 3018 5797 552737 -fid 3000 1202 266
```

on obtient :

```

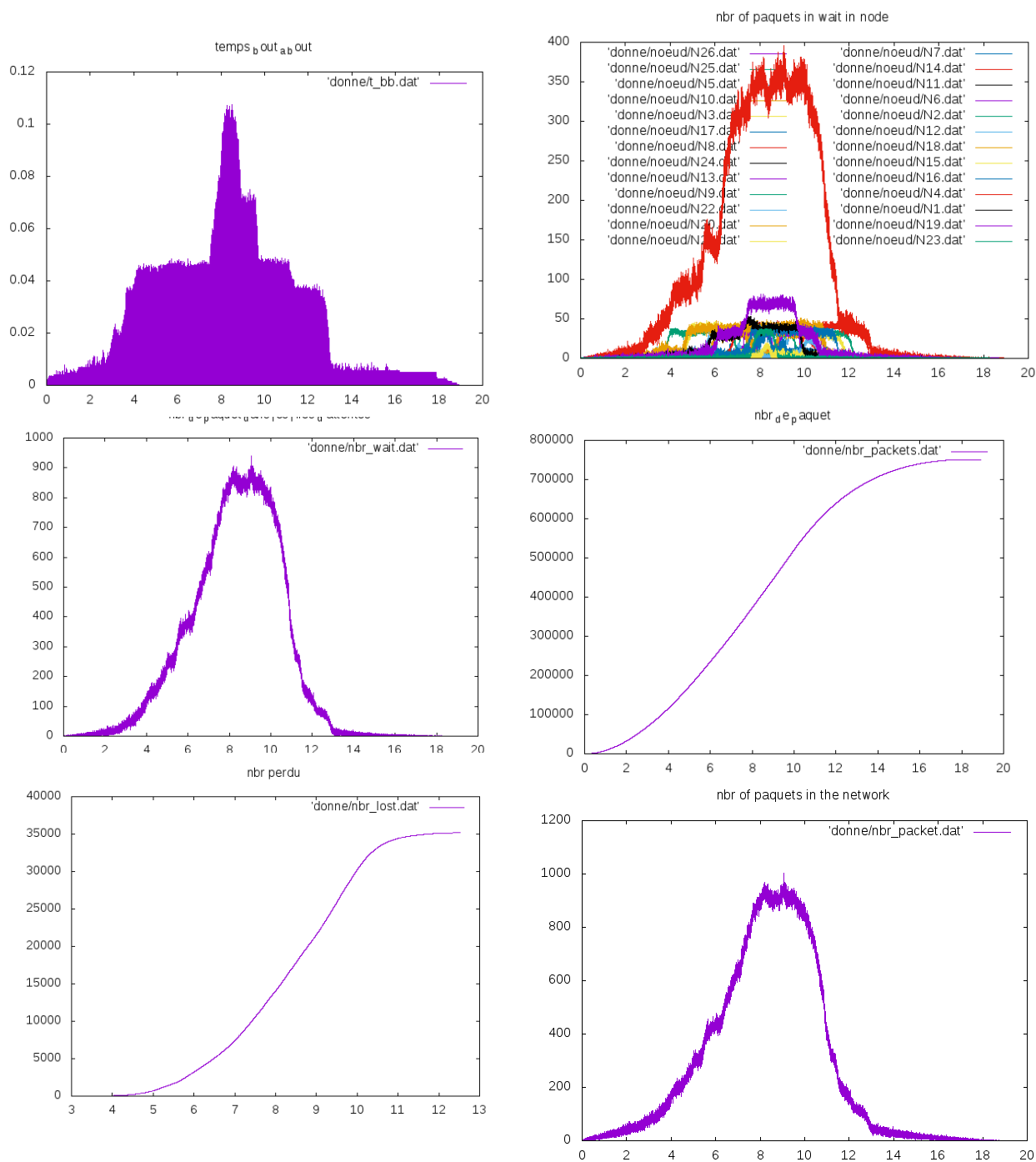
---donnée global-----
nbr routeur:26
nbr flux:3019
nbr paquets:750979
nbr perdu:35164
taux de perte:0.046824
moy_bb: 0.006920 ic=(0.006896,0.006944) ec=0.010204
moy_timewait: 0.004367 ic=(0.004350,0.004383) ec=0.008135
moy_tmetrans: 0.000659 ic=(0.000659,0.000660) ec=0.000580

---routeurs-----
n°noeud moy temps d'att(intervale de confiance) taux de perte nbr de perte nbr départ nbr arrivé nbr émis nbr reçus
26 0.000282(0.000273,0.000291) 0.000000 0 11238 10919 11238 10919
25 0.004371(0.004264,0.004478) 0.008102 376 22068 24343 21692 24343
5 0.004112(0.004020,0.004205) 0.003873 132 20877 13206 20745 13206
10 0.011402(0.011118,0.011686) 0.012758 287 10013 12482 9726 12482
3 0.000800(0.000783,0.000818) 0.000000 0 13386 13298 13386 13298
17 0.000313(0.000300,0.000326) 0.000000 0 10903 10606 10903 10606
8 0.001293(0.001247,0.001340) 0.000000 0 7242 8764 21276 22798
24 0.001244(0.001213,0.001275) 0.000000 0 10316 12350 10316 12350
13 0.007077(0.006863,0.007291) 0.001631 33 10017 9984 10215 10215
9 0.017913(0.017650,0.018176) 0.062459 1813 15991 13036 14178 13036
22 0.000332(0.000327,0.000338) 0.000000 0 27998 25619 27998 25619
20 0.003497(0.003452,0.003541) 0.013634 989 40392 32145 39403 32145
21 0.017314(0.017057,0.017571) 0.056711 1691 16834 12984 15143 12984
7 0.000960(0.000947,0.000972) 0.000131 15 55732 58520 55717 58520
14 0.013967(0.013732,0.014201) 0.047629 1669 19460 15582 17791 15582
11 0.001982(0.001872,0.002091) 0.000000 0 4175 4642 4175 4642
6 0.004137(0.004043,0.004231) 0.015574 589 6121 7937 29293 31698
2 0.000595(0.000580,0.000610) 0.000000 0 13107 12979 15689 15561
12 0.002647(0.002546,0.002748) 0.000164 4 10018 9103 15251 14340
18 0.004581(0.004543,0.004619) 0.052928 5883 54130 49185 56084 57022
15 0.001373(0.001347,0.001400) 0.000000 0 18362 10617 18362 10617
16 0.003175(0.003132,0.003217) 0.022020 2162 48489 49695 46327 49695
4 0.004691(0.004666,0.004717) 0.026268 16878 223002 215442 410206 419524
1 0.005370(0.005260,0.005480) 0.006812 364 16707 22476 30596 36729
19 0.003683(0.003632,0.003734) 0.018444 2250 56997 52447 67293 64993
23 0.005125(0.004890,0.005359) 0.001983 29 7404 7223 7375 7223

```

---Fluxs---									
fid	nbr émis	nbr reçus	nbr lost	temps debut	temps fin	temps de vie	moy temps d'acheminement(ic)		
266	433	426	7	0.829204	7.315106	6.485902	0.000782(0.000782,0.000782)		
1202	402	402	0	3.937430	11.745163	7.807733	0.000678(0.000678,0.000678)		
3000	354	347	7	9.921910	17.452683	7.530773	0.000318(0.000318,0.000318)		
---Paquet---									
pid	source	dest	t envoi	t fin	taille	moy temps d'att	moy temps de transmission		noeud parcourue
552737	4	14	0.458868	0.493093	0	0.033340(0.033340,0.033340)	0.000885(0.000885,0.000885)		14-4
5900	18	14	0.604958	0.605959	0	0.000000(0.000000,0.000000)	0.000593(0.000290,0.000895)		14-4-18
5797	7	13	0.603599	0.605182	0	0.000000(0.000000,0.000000)	0.000950(0.000433,0.001467)		13-4-7
3018	23	16	0.000000	0.001894	0	0.000000(0.000000,0.000000)	0.000631(0.000631,0.000631)		16-1-19-23
real	0m11.435s								
user	0m10.616s								
sys	0m0.184s								

on obtient aussi ces graphiques.



Interprétation :

on peut voir que le réseau a des pertes et donc il est mal proportionné. On peut voir qu'il y a un nœud plus utilisé que les autres. Le nœud 4 est bien plus utilisé que les autres mais aussi il est le plus sous dimensionné car c'est lui qui a le plus grand nombre de perte celui qui a le plus de paquet en attente. On peut aussi en conclure que le nœud 4 a beaucoup de lien avec d'autres nœuds. On peut voir que le nombre de paquet augmente dans le réseau monte de plus en plus rapidement dans le réseau au début jusqu'à 8 secondes donc on a ici pendant toute cette période des ouvertures de flux et ensuite on la courbe qui stagne jusqu'à 10 secondes on peut donc penser que durant cette période on commence à avoir des flux qui se ferment. Après les 10 secondes on la courbe du nombre de paquets dans le réseau qui diminue, on peut en déduire qu'il n'y a plus de flux qui s'ouvre et qu'il sont tous train de se fermer chacun son tour.

Optimisation :

L'analyseur prend 11 secondes pour analyser le fichier trace. Pour arriver à ce temps là, j'ai fait des optimisations. Je lis le fichier trace par paquet de 5000 octets pour éviter des lectures dans le disque à chaque ligne du document qui ralentirait le programme. Pour lister les paquets et les flux j'ai utilisé un tableau de hachage de liste de flux ou paquet de plus je supprime le paquet de sa liste quand il est détruit ou arrivé à destination. Ces tableaux de hachage me permettent réduire fortement les temps de parcours de liste car les éléments sont répartis dans des listes dans tout le tableau de hachage ce qui permet ensuite de parcourir des listes très courtes. Je n'ai pas fait de tableau de hachage pour les routeurs car il y en a souvent très peu donc je n'ai pas trouvé une utilité à en faire.