

QNAP

Boulic Guillaume, Émeric Tosi

19 mars 2016

Sommaire

| | | |
|----------|---|-----------|
| 1 | RTC : Réseau Téléphonique Commuté | 2 |
| 1.1 | Analyse sur un lien | 2 |
| 1.2 | Analyse sur un réseau de trois commutateurs | 5 |
| 2 | Commutation de paquets | 8 |
| 2.1 | Un commutateur de paquets | 8 |
| | Conclusion | 13 |
| A | Annexes | 14 |
| A.1 | Exemple(s) | 14 |
| A.2 | Analyse sur un lien | 15 |
| A.3 | Analyse sur un réseau de trois commutateurs | 18 |
| A.4 | Un commutateur de paquets | 21 |

Chapitre 1

RTC : Réseau Téléphonique Commuté

1.1 Analyse sur un lien

1.1.1 Énoncé

Considérons un lien d'un réseau à commutation de circuits permettant de véhiculer de la voix téléphonique.

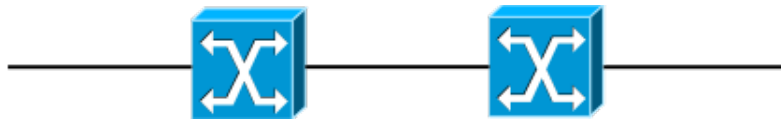


FIGURE 1.1 – Schéma du réseau à commutation de circuit étudié

Chacune des connexions nécessite un débit de 64 Kb.s^{-1} de façon bi-directionnel. On peut multiplexer simultanément C appels téléphoniques sur ce lien.

Le nombre d'utilisateurs est suffisamment grand pour supposer que les arrivées des nouveaux appels suivent une loi de paramètre λ , les durées des appels sont supposées suivre une loi exponentielle de paramètre μ avec ($\frac{1}{\mu} = 3$ minutes).

1.1.2 Probabilité de blocage d'appel en fonction de la charge ρ et de la capacité C

$$P(\text{blocage}) = \frac{\frac{\rho^C}{C!}}{\sum_{i=0}^C \frac{\rho^i}{i!}}$$

avec ρ la charge en Erlang et C la capacité.

Voici les résultats de ce calcul, obtenus grâce aux calculs réalisés par notre script Python inclus en annexe A.2

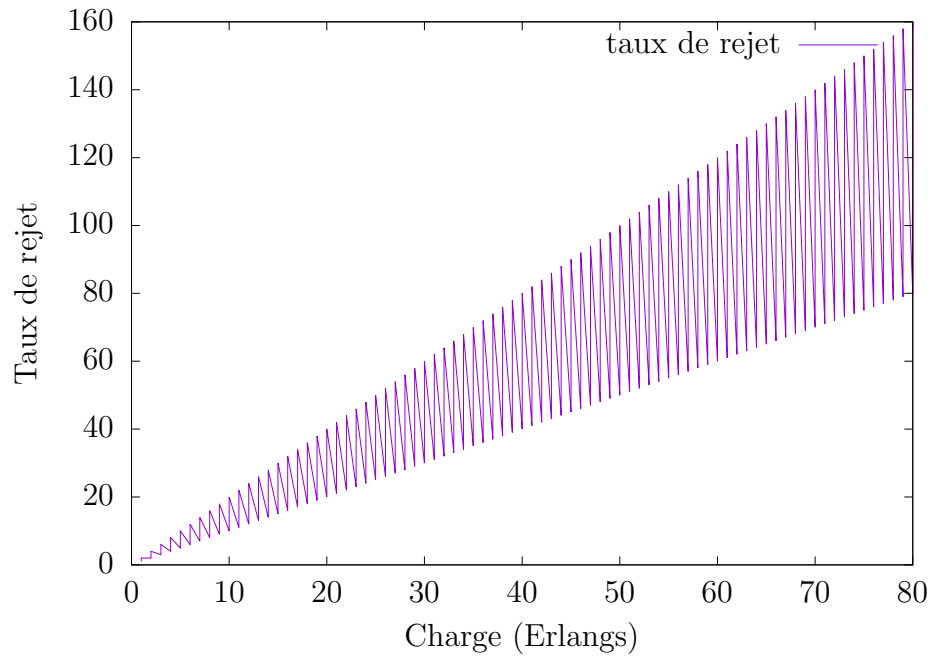


FIGURE 1.2 – Résultats de l'étude théorique.

1.1.3 Simulation de cette probabilité de blocage

Pour une charge $10 < \rho < 70$ et pour une capacité $\rho < C < 2 * \rho$.

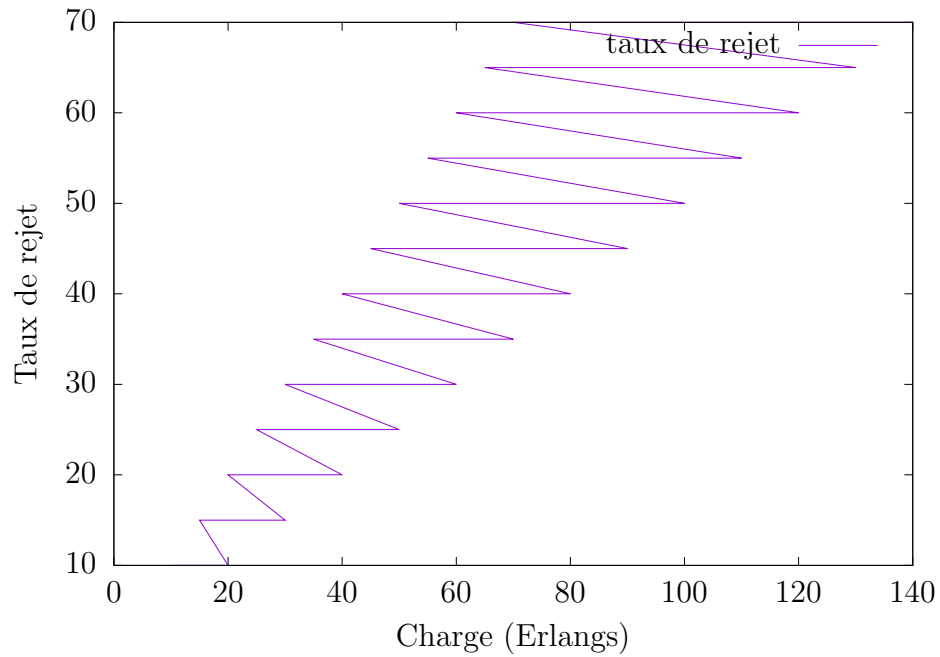


FIGURE 1.3 – Résultats de l'étude théorique.

1.1.4 Comparaison des taux de blocage expérimentaux et théoriques

Blablabla 1

1.2 Analyse sur un réseau de trois commutateurs

1.2.1 Énoncé

Désormais, nous considérons le réseau composé de 3 noeuds.

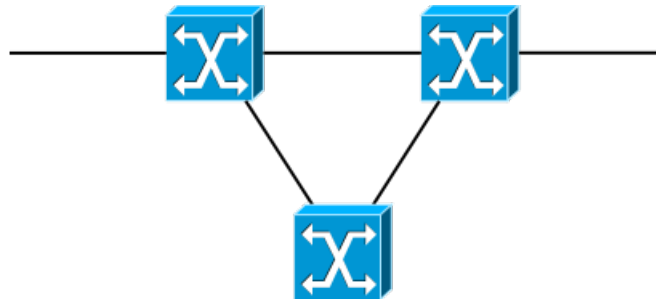


FIGURE 1.4 – Schéma du réseau à 3 commutateurs de circuit étudié

Les arrivées sont supposées Poissonniennes sur chacun des noeuds et le trafic se répartit équiprobablement entre les différents noeuds. Les durées des appels sont supposées exponentielles de même paramètre que dans la première partie (1-a). Nous ne considérons pas les appels locaux ni les appels qui n'aboutissent pas (absence).

1.2.2 Probabilités de blocage avec le chemin de débordement en cas de saturation du chemin direct

Blablabla 2 Avec un canal de débordement on obtient donc une capacité doublée, il n'y plus de blocage ... ?

1.2.3 Comparaison des résultats avec la partie 1.1

Blablabla 3

1.2.4 Problèmes à très forte charge !

Une solution consiste à n'utiliser le chemin de débordement que lorsque celui-ci n'est pas très encombré, c'est à dire en dessous d'un certain seuil d'occupation sur chacun des liens. Cela revient donc à laisser une marge M aux appels directs.

Commentaires

Blablabla 4

Simulation en prenant une marge comprise entre 1 et 3

Blablabla 5

Chapitre 2

Commutation de paquets

2.1 Un commutateur de paquets

2.1.1 Énoncé

Nous cherchons à simuler un lien de sortie d'un commutateur de paquets.



FIGURE 2.1 – Schéma du système à commutateur de paquets étudié

L'arrivée des paquets est supposée suivre une loi exponentielle de paramètre λ . Nous positionnons une file en sortie du commutateur pour stocker les différents paquets. Les paquets ont une longueur exponentiellement distribuée de paramètre $\frac{1}{\nu} = 10$ Kb. Le lien de sortie a un débit de 10 Mb.s^{-1} .

2.1.2 Calcul analytique du temps moyen de service $\frac{1}{\mu}$

$$\begin{aligned}\text{Temps moyen de service} &= \frac{1}{\mu} \\ \iff \frac{1}{\nu} * \frac{1}{D} &= 10 * 10^3 * \frac{1}{10^7} \\ \iff 10^4 * 10^{-7} &= 10^{-3} \text{ seconde}\end{aligned}$$

2.1.3 Déterminer le nombre moyen de paquets dans la file et le temps moyen de réponse en fonction du taux d'arrivée pour différentes durées de simulation

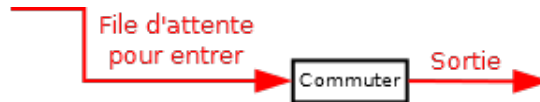


FIGURE 2.2 – Schéma de fonctionnement d'un commutateur de paquets

$$\lambda = \rho * \mu$$

$$\text{Charge de trafic } \rho = \frac{\lambda}{\mu}$$

$$\text{Nombre moyen de client } \bar{N} = \frac{\rho}{(1 - \rho)}$$

$$\text{Temps moyen de reponse } \bar{W} = \frac{1}{(\mu - \lambda)}$$

$$\bar{N} = \lambda \bar{W}$$

| | | | |
|----------------------------|---------|------------|------------|
| ρ | 0.1 | 0.5 | 0.9 |
| λ | 10^2 | $5 * 10^2$ | $9 * 10^2$ |
| Nombre moyen de client | 0.11111 | 1 | 9 |
| Temps moyen de réponse [s] | 0.00111 | 0.002 | 0.01 |

2.1.4 Comparaison du résultat de la simulation avec la théorie

Blablabla 1

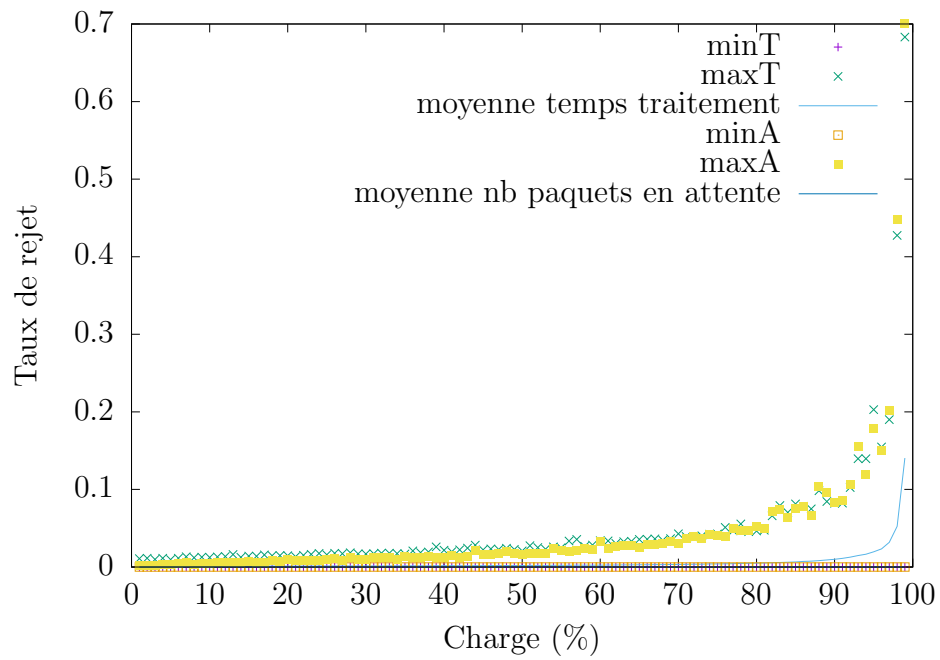


FIGURE 2.3 – Résultats de la simulation.

2.1.5 Cas où les paquets ont une taille fixe de 10 Kb

Calculer analytiquement le temps moyen de service $\frac{1}{\mu}$

On obtient la même chose que précédemment :

$$\text{Temps moyen de service} = \frac{1}{\mu}$$

$$\Longleftrightarrow \frac{1}{\nu} * \frac{1}{D} = 10 * 10^3 * \frac{1}{10^7}$$

$$\Longleftrightarrow 10^4 * 10^{-7} = 10^{-3} \text{ seconde}$$

Résultats en fonction du taux d'arrivée pour différentes durées de simulation

Temps moyen de réponse et nombre moyen de paquets dans la file d'attente Blablabla 2

Analyse et comparaison des résultats

Blablabla 3

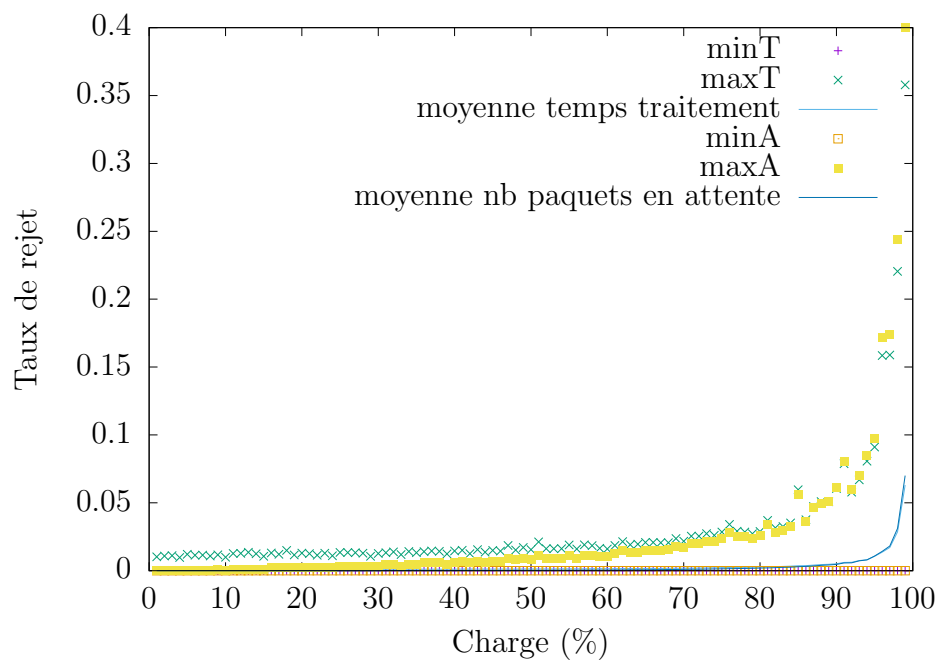


FIGURE 2.4 – Résultats de la simulation pour des paquets à la taille fixe de 10 Kb.

Conclusion

Too much bullshit here :P

Annexe A

Annexes

A.1 Exemple(s)

emphatique **gras** machine à écrire *incliné* PETITES MAJUSCULES

The foundations of the rigorous study of *analysis* were laid in the nineteenth century, notably by the mathematicians Cauchy and Weierstrass. Central to the study of this subject are the formal definitions of *limits* and *continuity*.

Let D be a subset of \mathbf{R} and let $f: D \rightarrow \mathbf{R}$ be a real-valued function on D . The function f is said to be *continuous* on D if, for all $\epsilon > 0$ and for all $x \in D$, there exists some $\delta > 0$ (which may depend on x) such that if $y \in D$ satisfies

$$|y - x| < \delta$$

then

$$|f(y) - f(x)| < \epsilon.$$

One may readily verify that if f and g are continuous functions on D then the functions $f+g$, $f-g$ and $f.g$ are continuous. If in addition g is everywhere non-zero then f/g is continuous.

A.2 Analyse sur un lien

Script Python de calcul de la probabilité de blocage d'appel P en fonction de la charge ρ et de la capacité C .

```
1 import math
2
3 for p in range(1,80+1):
4     for c in range(p,p*2+1):
5         numerateur = pow(p,c) * 1.0 / math.factorial(c)
6         denominateur = 0
7         for i in range(0,c):
8             denominateur += pow(p,i) * 1.0 / math.factorial(i)
9         print( str(c) + " " + str(p) + " " + str(numerateur /
            denominateur))
```


Simulation d'un lien d'un réseau à commutation de circuits.

```
1
2 &
3 /CONTROL/
4     OPTION = NSOURCE;
5     ACCURACY = ALL QUEUE;
6     TMAX = 100000;
7
8 &
9 /DECLARE/
10    QUEUE SRC, LIEN;
11    INTEGER NbRejets, Erlang, I, C, RHO, NbSimu;
12    REAL NbOK, DEBIT, LAMBDA, MU, REJET, X;
13    FILE cbQ;
14
15 &
16 /STATION/
17     NAME = SRC;
18     TYPE = SOURCE;
19     SERVICE = EXP(1./LAMBDA);
20     TRANSIT = LIEN;
21
22 &
23 /STATION/
24     NAME = LIEN;
25     TYPE = MULTIPLE(C);
26     SERVICE = BEGIN
27         EXP(1./(MU));
28         NbOK := NbOK + 1;
29     END;
30     CAPACITY = C;
31     REJECT = BEGIN
32         NbRejets := NbRejets + 1;
33         TRANSIT(OUT);
34     END;
35     TRANSIT = OUT;
36
37 &
38 /EXEC/
39     BEGIN
40         FILASSIGN(cbQ , "p1.q2.data");
41         OPEN(cbQ);
42         &
```

```

43      DEBIT := 64000;
44      MU := 1. / 180;
45      Erlang := 70;
46      NbSimu := 5;
47      &
48      FOR RHO := 10 STEP 5 UNTIL Erlang DO
49      BEGIN
50          &
51          LAMBDA := MU * RHO;
52          &
53          FOR C := RHO STEP 5 UNTIL (RHO * 2) DO
54          BEGIN
55              &
56              NbOK := 0;
57              NbRejets := 0;
58              &
59              FOR I := 1 STEP 1 UNTIL NbSimu DO
60              BEGIN
61                  SIMUL;
62              END;
63              &
64              NbOK := NbOK / NbSimu;
65              NbRejets := NbRejets / NbSimu;
66              &
67              WRITELN(cbQ , " " , RHO , " " , C , " " , NbRejets ,
68                  " " , NbOK , " " , NbRejets / (NbOK + NbRejets));
69              &
70              END;
71      END;
72
73      &
74      /END/
75
76      &
77      & plot "p1.q2.data" u 1:2 w l t "taux de rejet"
78      &

```

A.3 Analyse sur un réseau de trois commutateurs

Simulation d'un lien d'un réseau à 3 commutateurs de circuits.

```
1
2 /CONTROL/
3 OPTION = NSOURCE;
4 TMAX = 100000;
5 ACCURACY = ALL QUEUE;
6
7
8 /DECLARE/
9 QUEUE SRC,LIEN_H, LIEN_B_1, LIEN_B_2;
10 INTEGER nbTotal, nbTransmis, nbRejets,erlang,I,J,capacite,rho,
    NbSimu = 5;
11 REAL somme,DEBIT,lambda,mu,pRejets,X, pourcentage;
12 CUSTOMER REAL duree;
13 FILE fichier;
14
15
16 /STATION/
17 NAME = SRC;
18 TYPE = SOURCE;
19 SERVICE = BEGIN
20     duree := ( 1. / mu );
21     nbTotal := nbTotal + 1;
22     EXP( 1. / lambda );
23 END;
24 TRANSIT = LIEN_H;
25
26 /STATION/
27 NAME = LIEN_H;
28 TYPE = MULTIPLE(capacite);
29 SERVICE = BEGIN
30     CST(duree);
31     TRANSIT(OUT);
32 END;
33 CAPACITY = capacite;
34
35 REJECT = BEGIN
36     nbRejets := nbRejets + 1;
37     TRANSIT(LIEN_B_1);
```

```

38 | END;
39 |
40 |
41 | /STATION/
42 | NAME = LIEN_B_1;
43 | TYPE = MULTIPLE(capacite);
44 | SERVICE = BEGIN
45 |     CST(duree);
46 |     TRANSIT(LIEN_B_2);
47 | END;
48 | CAPACITY = capacite;
49 | REJECT = BEGIN
50 |     nbRejets := nbRejets + 1;
51 |     TRANSIT(OUT);
52 | END;
53 |
54 | /STATION/
55 | NAME = LIEN_B_2;
56 | TYPE = MULTIPLE(capacite);
57 | SERVICE = BEGIN
58 |     CST(duree);
59 |     TRANSIT(OUT);
60 | END;
61 | CAPACITY = capacite;
62 | REJECT = BEGIN
63 |     nbRejets := nbRejets + 1;
64 |     TRANSIT(OUT);
65 | END;
66 |
67 |
68 | /EXEC/
69 |     BEGIN
70 |         FILASSIGN(fichier , "p1.q3.data");
71 |         OPEN(fichier);
72 |
73 |         DEBIT := 64000;
74 |         mu := 1. / 180;
75 |         erlang := 70;
76 |
77 |         FOR rho := 1 STEP 1 UNTIL erlang DO
78 |             BEGIN
79 |                 FOR capacite := rho STEP 1 UNTIL 2 * rho DO
80 |                     BEGIN

```

```

81         lambda := mu * rho;
82         nbTotal := 0;
83         nbRejets := 0;
84         nbTransmis := 0;
85
86         FOR I := 1 STEP 1 UNTIL NbSimu DO
87             BEGIN
88                 SIMUL;
89             END;
90
91         nbTotal := nbTotal / NbSimu;
92         nbRejets := nbRejets / NbSimu;
93         nbTransmis := nbTransmis / NbSimu;
94
95         pourcentage := nbRejets * 1. / nbTotal;
96
97         WRITELN(fichier, rho , " " , capacite , " " ,
98                 nbTotal , " " , nbRejets , " " , pourcentage )
99                 ;
100         END;
101 /END/

```

A.4 Un commutateur de paquets

Simulation d'un lien de sortie d'un commutateur de paquets.

```
1
2 &
3 /DECLARE/
4
5 QUEUE SRC, Q;
6
7 REAL LAMBDA, RO;
8 REAL MU = 1000;
9 INTEGER I;
10
11 QUEUE fakeOUT;
12
13 REAL temps;
14
15 REAL minA;
16 REAL maxA;
17 REAL sumA;
18
19 REAL minT;
20 REAL maxT;
21 REAL sumT;
22
23 CUSTOMER INTEGER nbpsg;
24 CUSTOMER REAL Tdebut;
25
26 FILE cbQ;
27
28
29 &
30 /STATION/
31
32 NAME = SRC;
33 TYPE = SOURCE;
34 SERVICE = BEGIN
35     EXP(1. / LAMBDA);
36     Tdebut := TIME;
37 END;
38 TRANSIT = Q;
39
40
```

```

41 &
42 /STATION/
43
44 NAME = Q;
45 SERVICE = BEGIN
46     EXP(1. / MU);
47 END;
48 TRANSIT = fakeOUT;
49
50
51 &
52 /STATION/
53
54 NAME = fakeOUT;
55 SERVICE = BEGIN
56     temps := (TIME - Tdebut);
57
58     sumT := sumT + temps;
59
60     IF minT > temps THEN
61     BEGIN
62         minT := temps;
63     END;
64
65     IF maxT < temps THEN
66     BEGIN
67         maxT := temps;
68     END;
69
70     sumA := sumA + CUSTNB(Q);
71
72     IF minA > CUSTNB(Q) THEN
73     BEGIN
74         minA := CUSTNB(Q);
75     END;
76
77     IF maxA < CUSTNB(Q) THEN
78     BEGIN
79         maxA := CUSTNB(Q);
80     END;
81 END;
82 TRANSIT = OUT;
83

```

```

84
85 &
86 /CONTROL/
87
88 TMAX = 1000;
89
90
91 &
92 /EXEC/
93
94 BEGIN
95     & On fait varier Rho pour qu'il prenne 99 valeurs definies et qu
          'on realise nos mesures (0.01 , ... , 0.99)
96     FILASSIGN(cbQ,"p2.q3.data");
97     OPEN(cbQ);
98     & boucle des simulations
99     FOR I := 1 STEP 1 UNTIL 99 DO
100     BEGIN
101         & initialisation des variables
102         RO := I * 0.010;
103         LAMBDA := RO * MU;
104         minA := 100000000;
105         maxA := 0;
106         sumA := 0;
107         minT := 100000000;
108         maxT := 0;
109         sumT := 0;
110         & simulation
111         SIMUL;
112         & affichage de la progression
113         WRITELN(I);
114         & ecriture resultats dans le fichier
115         WRITELN(cbQ,I," ",minT," ",maxT," ",sumT/SERVNB(fakeOUT)," "
          ,minA," ",maxA," ",sumA/SERVNB(fakeOUT));
116     END;
117 END;
118
119
120 &
121 /END/
122
123
124 &

```



```
125 & GNUPLOT
126 &
127 & plot "p2.q3.data" u 1:2 t "minT", "p2.q3.data" u 1:3 t "maxT", "
    p2.q3.data" u 1:4 w l t "moyenne temps traitement", "p2.q3.data"
    u 1:5 t "minA" axes x1y2, "p2.q3.data" u 1:6 t "maxA" axes x1y2
    , "p2.q3.data" u 1:7 w l t "mayyenne nb packet att" axes x1y2
128 &
```