

# PERANCANGAN DAN SIMULASI ANTRIAN PAKET DENGAN MODEL ANTRIAN M/M/N DI DALAM SUATU JARINGAN KOMUNIKASI DATA

Idatriska P<sup>1</sup>, R. Rumani M<sup>2</sup>, Asep Mulyana<sup>3</sup>

<sup>1,2,3</sup>Gedung N-203, Program Studi Sistem Komputer,  
Fakultas Teknik Elektro, Universitas Telkom  
Jl. Telekomunikasi No.1, Dayeuhkolot, Bandung 40257 Indonesia  
idatriskap@yahoo.com<sup>1</sup>, rumani@telkomuniversity.ac.id<sup>2</sup>,  
asepmulyana@telkomuniversity.ac.id<sup>3</sup>

## Abstrak

Penelitian ini difokuskan pada model antrian M/M/N yang disimulasikan dengan bahasa C. Dalam simulasi antrian ini, dibuat tiga skenario percobaan untuk melihat performansi algoritma yang telah dirancang, berdasarkan parameter pola kedatangan, *average waiting time*, *utilization*, dan *average number of packet in the queue*. Skenario pertama meneliti tentang pola kedatangan paket. Skenario kedua mengamati pengaruh penambahan *server*. Skenario ketiga meneliti tentang pengaruh penambahan *server* sekaligus dengan penambahan *packet size*.

Dari hasil penelitian yang dilakukan, dibuktikan bahwa pola kedatangan paket bersifat acak (*random*) dan terdistribusi secara eksponensial. Dari hasil simulasi yang dilakukan untuk skenario kedua, ternyata dengan adanya penambahan *server* maka *average waiting time* dan *average number of packet in the queue* semakin kecil, sedangkan *utilization* menjadi semakin besar. Untuk skenario ketiga, hasil yang diperoleh adalah, dengan penambahan *packet size*, diperoleh nilai *average waiting time* dan *utilization* semakin tinggi, sedangkan *average number of packet in the queue* semakin kecil.

**Kata kunci:** *model antrian M/M/N, average waiting time, utilization, average number of packet in the queue, C.*

## 1. Pendahuluan

### 1.1. Latar Belakang

Dalam suatu situasi antrian, proses kedatangan adalah stokastik sehingga perlu diketahui distribusi probabilitas yang menggambarkan waktu kedatangan paket yang berturut-turut dan urutan waktu layanan paket. Suatu sistem antrian terdiri dari *input*, antrian dan *server* sebagai pusat layanan.

Secara umum, *server* terdiri dari satu atau lebih untuk melayani paket yang tiba di sistem pada tingkat tertentu. Paket tiba di pusat layanan untuk mendapatkan layanan, menunggu untuk dilayani jika ada ruang tunggu, dan meninggalkan sistem setelah dilayani. Asumsi penting disini adalah  $average\ arrival\ rate \leq maximum\ service\ rate$ , agar tidak terjadi *dead lock*. Demikian pula, jumlah server yang tersedia (N) dalam model antrian yang digunakan akan menentukan tingkat pelayanan yang dapat diberikan oleh sistem, sesuai dengan tingkat kepadatan lalu-lintas yang terdapat di dalam sistem.

### 1.2 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

Merancang dan mensimulasikan model antrian M/M/N, mengukur dan menganalisis unjuk kerja jaringan menggunakan beberapa parameter unjuk kerja jaringan, yaitu: *average waiting time*, *utilization*, dan *average number of packet in the queue* yang nantinya dapat dijadikan sebagai bahan pertimbangan dalam menggunakan model antrian tersebut secara nyata dilapangan.

## 2. Dasar Teori

### 2.1. Pengenalan Teori Antrian

#### 2.1.1. Definisi Antrian [1]

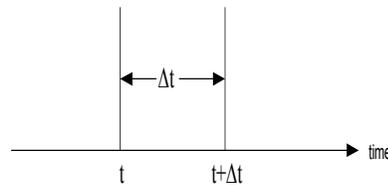
Teori antrian merupakan cabang dari terapan teori probabilitas yang awalnya digunakan untuk mempelajari kemacetan lalu lintas telepon. Teori antrian memegang peran kunci dalam menganalisis unjuk kerja jaringan komunikasi, baik untuk data, suara, maupun gambar.

#### 2.1.2. Proses Markov [4]

Proses Markov adalah proses stokastik, dimana masa lalu tidak mempunyai pengaruh pada masa yang akan datang, bila kondisi pada masa sekarang diketahui (*memoryless property*).

#### 2.1.3. Proses Poisson [5]

Dalam pembahasan tentang teori antrian pada umumnya selalu diasumsikan adanya proses Poisson sebagai proses kedatangan, dimana paket-paket yang masuk ke dalam suatu *node* diasumsikan datang secara acak (*random*).



**Gambar 1.** Interval waktu yang digunakan dalam mendefinisikan proses Poisson

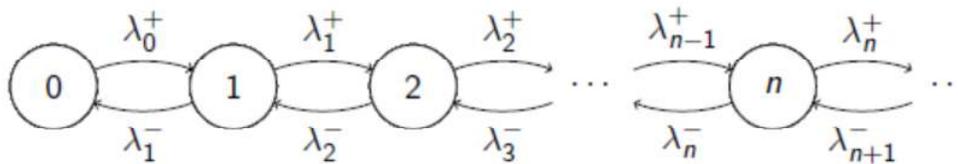
#### 2.1.4. Rumus Little [3]

J.D Little menyatakan jumlah rata-rata pelanggan dalam suatu sistem antrian sama dengan rata-rata datangnya panggilan pada sistem tersebut dikalikan dengan waktu rata-rata pelanggan dalam sistem, dinyatakan dengan rumus:

$$\bar{L} = \lambda \cdot \bar{W} \quad (1)$$

#### 2.1.5. Proses Birth dan Death [6]

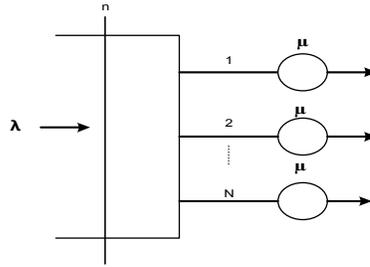
Penggambaran matematis untuk proses trafik yaitu dengan stokastik yang disebut dengan proses *birth* (kelahiran) dan *death* (kematian). Proses ini bergerak dari kondisi  $n$  ke kondisi  $n-1$  jika terjadi kematian, atau bergerak ke  $n+1$  jika terjadi kelahiran. Tetap dikondisi  $n$  jika tidak ada panggilan yang datang maupun keluar dari sistem. Perubahan kondisi akibat proses kelahiran dan kematian ditunjukkan pada gambar 1 berikut:



**Gambar 2.** Diagram state dependent queue (birth-death process)

Proses kelahiran dan kematian sangat penting dalam menganalisis unjuk kerja jaringan.

## 2.2. M/M/N Queue [7]



**Gambar 3. Model Antrian M/M/N [7]**

Model M/M/N *queue* dilayani oleh N *server*, dimana masing-masing *server* mempunyai kebebasan (bersifat independen) dan identik, terdistribusi *exponential service-time distribution*, dengan proses kedatangan diasumsikan dengan Poisson. Jika diasumsikan *exponential service-time statistics*, dengan  $1/\mu$  adalah *average service time* dan  $\lambda$  adalah *arrival rate*, dengan  $\rho = \lambda/N\mu$ , maka probabilitas kesetimbangan dari n paket di dalam sistem dirumuskan dengan: [5]

$$\begin{aligned} p_n &= \frac{(N\rho)^n}{n!} p_0, \quad n \leq N \\ &= N^N \frac{\rho^n}{N!} p_0, \quad n \geq N \end{aligned} \quad (2)$$

$\rho$  didefinisikan sebagai parameter *utilization* per *trunk* atau saluran

$$\rho = \lambda/N\mu < 1 \quad (3)$$

Menggunakan rumus (2) dan  $\sum_{n=0}^{\infty} p_n = 1$ , diperoleh rumus:

$$p_0 = \left[ \sum_{n=0}^{N-1} \frac{(N\rho)^n}{n!} + \frac{1}{1-\rho} \frac{(N\rho)^N}{N!} \right]^{-1} \quad (4)$$

Probabilitas  $P(\text{delay})$  bahwa sebuah paket akan *delay*, dapat dirumuskan sebagai berikut :

$$P_Q(\text{delay}) = \sum_{n=N}^{\infty} p_n = \frac{(\rho n)^n p_0}{(1-\rho)n!} \quad (5)$$

Rumus diatas disebut rumus Erlang-C yang digunakan untuk sistem tipe *delay*.

*Average number queue* (tidak dalam pelayanan)

$$E[N_Q] = \sum_{n=N+1}^{\infty} (n - N) p_n = \sum_{n=N+1}^{\infty} (-N) p_0 \left( \frac{(N\rho)^n}{N! N^{k-N}} \right) \quad (6)$$

$$E[N_Q] = p_0 \frac{(N\rho)^N}{N!} \sum_{n=N+1}^{\infty} (n - N) \rho^{n-N} = p_0 \frac{(N\rho)^N}{m!} \sum_{k=0}^{\infty} k \rho^k \quad (7)$$

Dengan menggunakan rumus (5) dan  $1 - \rho \left( \sum_{k=0}^{\infty} k \rho^k \right) = \frac{\rho}{1-\rho}$ , besarnya *mean number waiting* di dalam antrian adalah sebagai berikut :

$$E[N_Q] = P_Q (1 - \rho) \frac{\rho}{(1-\rho)^2} = P_Q \frac{\rho}{(1-\rho)} \quad (8)$$

Berdasarkan *Little Law*:

$$E[W] = P_Q \frac{\rho}{\lambda(1-\rho)} \quad (9)$$

Menggunakan  $\rho = \frac{\lambda}{m\mu}$ , didapatkan:

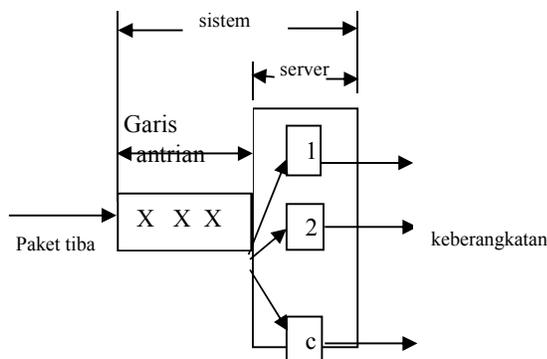
$$E[T] = E[W] + \frac{1}{\mu} = \frac{1}{\mu} + \frac{P_Q}{m\mu - \lambda} \quad (10)$$

Menggunakan *Little Law*, diperoleh :

$$E[N] = \left( \frac{1}{\mu} + \frac{P_Q}{m\mu - \lambda} \right) \lambda = N\rho + \frac{\rho P_Q}{1-\rho} \quad (11)$$

### 3. Perancangan Sistem

#### 3.1. Pemodelan Sistem



**Gambar 4. Model Sistem yang Disimulasikan**

Pada penelitian yang dilakukan, perancangan sistem model antrian M/M/N digunakan untuk menunjukkan performansi algoritma simulasi antrian M/M/N. Pada gambar 3, model antrian M/M/N terdiri dari  $c$  server dengan proses kedatangan acak, distribusi pelayanan eksponensial.

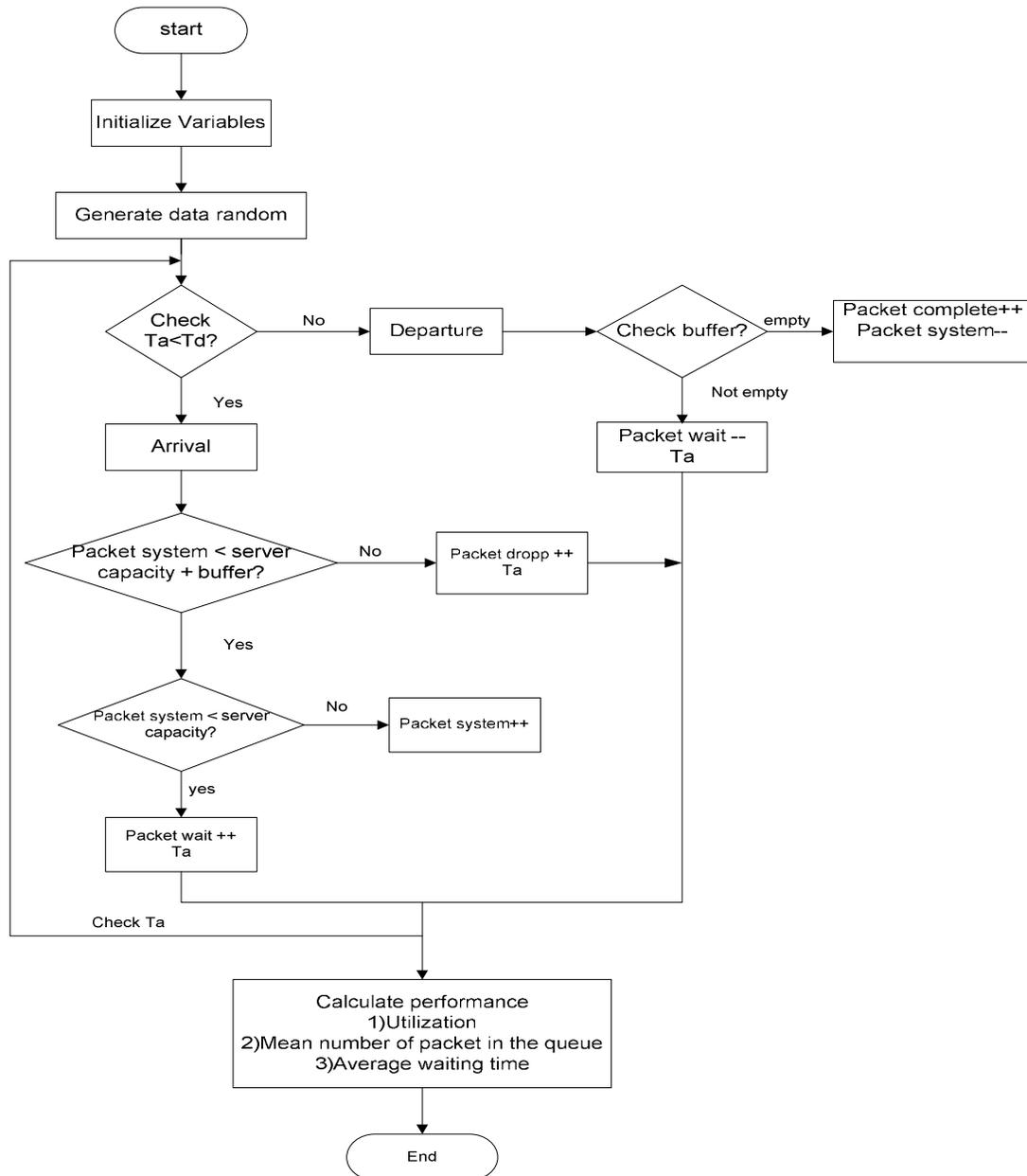
#### 3.2. Asumsi Dasar Simulasi

##### 3.2.1. Antrian M/M/N

Parameter-parameter antrian yang digunakan, adalah sebagai berikut :

- Kedatangan paket diasumsikan bersifat acak dan terdistribusi eksponensial.
- Disiplin antrian yang digunakan adalah *First In First Out* (FIFO).
- Mean service time* adalah sama untuk semua server.
- Diasumsikan bahwa kapasitas *buffer* adalah 1000 paket.
- Rata-rata ukuran sebuah paket di *set* oleh user.
- Bandwidth* yang digunakan, dengan kecepatan 1 Mbps.
- Jumlah server yang digunakan diubah-ubah dari satu sampai lima server.

Adapun diagram alir cara kerja simulator M/M/N queue yang digunakan, terdapat pada gambar 4 berikut.



Gambar 5. Flowchart Simulator M/M/N queue

### 3.2.2. Arrival Event dan Departure Event

Dalam model antrian terdapat dua jenis event yaitu kedatangan (*arrival*) dan kepergian (*departure*). Proses *arrival* ketika paket masuk ke dalam sistem untuk dilayani, dan proses *departure* adalah ketika paket sudah selesai dilayani dan meninggalkan sistem.

### 3.3. Skenario Simulasi [2]

#### Skenario 1 : Analisis pola kedatangan paket

Pada skenario ini akan diuji pengaruh perubahan waktu simulasi untuk mengetahui pola kedatangan paket pada algoritma antrian M/M/N yang telah dirancang. Waktu simulasi yang digunakan adalah 100 detik, 1000 detik dan 10000 detik dengan *mean interarrival time* 1 detik.

### **Skenario 2 : Analisis pengaruh perubahan jumlah server**

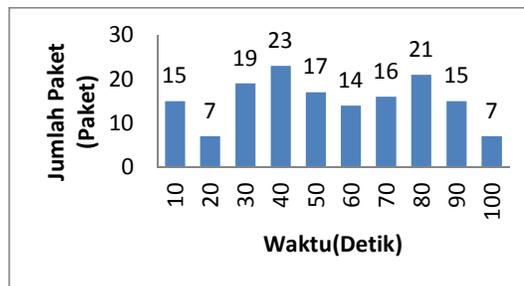
Pada skenario ini akan diuji pengaruh perubahan jumlah *server* terhadap performansi *average waiting time*, *utilization*, dan *average number of packet in the queue*. Jumlah *server* yang akan disimulasikan adalah satu sampai dengan lima.

### **Skenario 3 : Analisis pengaruh perubahan jumlah server dengan penambahan ukuran paket**

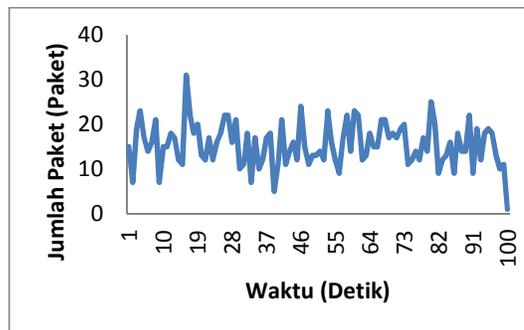
Pada skenario ini akan diuji pengaruh perubahan ukuran paket terhadap performansi *average waiting time*, *utilization*, dan *average number of packet in the queue*. Ukuran paket yang disimulasikan adalah 1000, 2000, dan 3000 byte.

## **4. Simulasi dan Analisis**

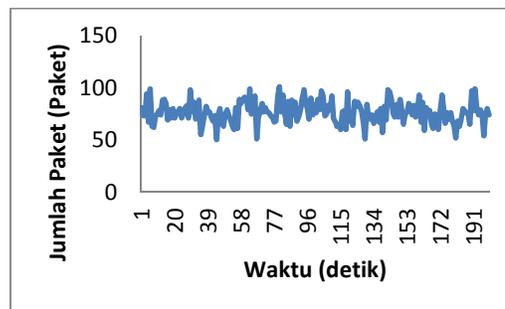
### **4.1 Analisis Pola Kedatangan Paket**



**Gambar 6. Grafik Hasil Skenario dengan Waktu Simulasi 100 Detik**



**Gambar 7. Grafik Hasil Skenario dengan Waktu Simulasi 1000 Detik**



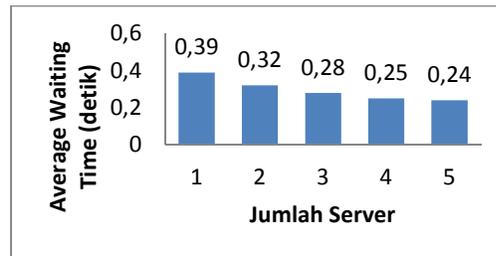
**Gambar 8. Grafik Hasil Skenario dengan Waktu Simulasi 10000 Detik**

Berdasarkan simulasi yang dilakukan selama waktu 100 detik, 1000 detik, dan 10000 detik maka dapat dilihat pada gambar-gambar 6, 7, dan 8 diatas, bahwa pola kedatangan

bersifat acak dan berdistribusi eksponensial. Distribusi suatu kedatangan diperhitungkan melalui *interarrival time*. Bila *interarrival time* setiap paket adalah *random* dan eksponensial maka kedatangan paket-paket tersebut mengikuti suatu distribusi Poisson.

## 4.2 Skenario Perubahan Jumlah Server

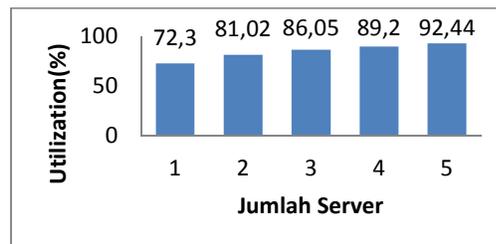
### 4.2.1 Analisis *Average Waiting Time* Penambahan Server



Gambar 9. Grafik *Average Waiting Time* Penambahan Jumlah Server

Dari gambar 9 diatas, dapat diamati bahwa pengaruh penambahan *server* terhadap *average waiting time*. Pada gambar tersebut terlihat bahwa penambahan jumlah *server* memperkecil *average waiting time*. *Average waiting time* terbesar yaitu 0,39 detik pada saat jumlah *server* 1. Namun ketika jumlah *server* dinaikkan sampai 5 menghasilkan *average waiting time* yang paling kecil yaitu 0.24 detik, hal ini dikarenakan jumlah *server* semakin banyak sehingga mempercepat proses pelayanan paket. Penambahan jumlah *server* mengakibatkan penurunan *average waiting time* dalam antrian.

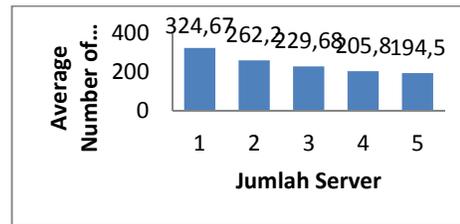
### 4.2.2 Analisis *Utilization* Penambahan Server



Gambar 10. Grafik *Utilization* Penambahan Server

Dari gambar 10 diatas, diperoleh bahwa pengaruh penambahan *server* terhadap utilisasi. Pada gambar tersebut terlihat bahwa penambahan *server* memperbesar utilisasi. Utilisasi terkecil yaitu 72,3% pada saat jumlah *server* 1. Namun ketika jumlah *server* dinaikkan sampai 5 menghasilkan utilisasi yang paling besar yaitu 92,44%, hal ini dikarenakan penambahan *server* akan meningkatkan nilai utilitas dari *resource* atau *server* itu sendiri, sehingga menyebabkan pemanfaatan dari setiap *server* tersebut optimal.

### 4.2.3 Analisis *Average Number Of Packet In The Queue* Penambahan *Server*

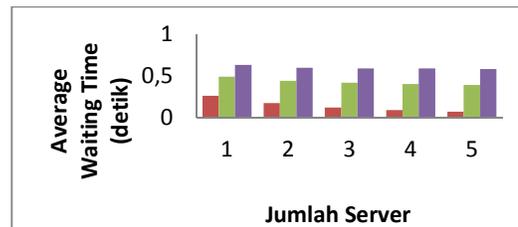


**Gambar 11.** Grafik *Average Number Of Packet In The Queue* Penambahan *Server*

Dari gambar 11 diatas, dapat diamati bahwa pengaruh penambahan *server* terhadap *average number of packet in the queue*. Pada gambar tersebut terlihat bahwa penambahan jumlah *server* memperkecil *average number of packet in the queue*. *Average number of packet in the queue* terbesar yaitu 324.67 paket pada saat jumlah *server* 1. Namun ketika jumlah *server* dinaikkan sampai 5 menghasilkan *average number of packet in the queue* yang paling kecil yaitu 194,5 paket. Semakin banyak jumlah *server* maka *server* akan melayani setiap paket dengan cepat, sehingga jumlah paket yang menunggu di *buffer* akan berkurang.

### 4.3. Skenario Penambahan *Packet Size*

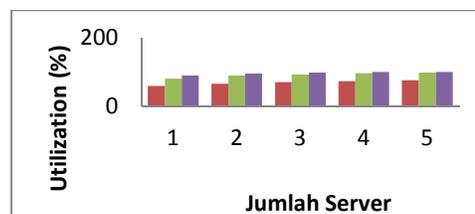
#### 4.3.1. Analisis *Average Waiting Time* Penambahan *Packet Size*



**Gambar 12.** Grafik *Average Waiting Time* Penambahan *Packet Size*

Dari gambar 12 diatas, dapat diamati, bahwa untuk paket berukuran 3000 *byte*, *average waiting time* tertinggi 0,63 detik dan terendah 0,58 detik, paket berukuran 2000 *byte* memiliki *average waiting time* tertinggi 0,49 detik dan terendah 0,39 detik, paket berukuran 1000 *byte* memiliki *average waiting time* tertinggi di 0,26 detik dan terendah 0,07 detik. Dari grafik diatas semakin besar nilai ukuran paket maka nilai *average waiting time* semakin tinggi.

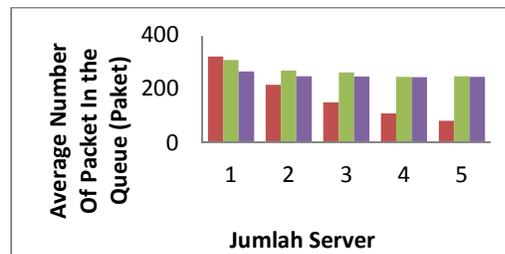
#### 4.3.2. Analisis *Utilization* Penambahan *Packet Size*



**Gambar 13.** Grafik *Utilization* Penambahan *Packet Size*

Dari grafik pada gambar 13 diatas, diperoleh hasil, bahwa untuk paket berukuran 1000 *byte* memiliki utilisasi terendah dengan nilai 58,8% dan tertinggi 74,77%, paket berukuran 2000 *byte* memiliki utilisasi terendah dengan nilai 80,59% dan tertinggi 97,61%, paket berukuran 3000 *byte* memiliki utilisasi terendah dengan nilai 89,61% dan tertinggi 99,71%. Berdasarkan simulasi yang dilakukan, perubahan ukuran paket menyebabkan nilai utilisasi semakin tinggi. Semakin besar ukuran paket maka nilai utilisasi semakin tinggi.

#### 4.3.3. Analisis Average Number Of Packet In The Queue Penambahan Packet Size



**Gambar 14. Grafik Average Number Of Packet In The Queue Penambahan Packet Size**

Dari gambar 14 diatas, diperoleh hasil, bahwa untuk paket berukuran 1000 *byte* memiliki *average number of packet in the queue* tertinggi dengan nilai 320,3 paket dan terendah 82,2 paket, paket berukuran 2000 *byte* memiliki *average number of packet in the queue* tertinggi dengan nilai 308,1 paket dan terendah 247,99 paket, paket berukuran 3000 *byte* memiliki *average number of packet in the queue* tertinggi dengan nilai 264,38 paket dan terendah 244,92 paket. Ketika terjadi penambahan ukuran paket, *average number of packet in the queue* yang dihasilkan semakin kecil, untuk ketiga ukuran paket tersebut

## 5. Kesimpulan dan Saran

### 5.1. Kesimpulan

Dari hasil skenario simulasi serta pengambilan data dan analisa performansi algoritma antrian M/M/N maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Hasil simulasi menunjukkan bahwa pola kedatangan paket pada algoritma ini adalah acak (*random*) dan terdistribusi eksponensial; dengan demikian, maka dapat disimpulkan bahwa pola kedatangan paket yang disimulasikan, mengikuti distribusi Poisson.
2. Penambahan *server* memperkecil *average waiting time*; *average waiting time* terbesar adalah 0,39 detik pada saat menggunakan 1 *server*. Pada saat jumlah *server* diperbesar menjadi 5, diperoleh *average waiting time* terkecil, yaitu 0,24 detik.
3. Semakin banyak jumlah *server* maka nilai *utilization* akan semakin besar. Pada saat digunakan 1 *server*, besarnya *utilization* adalah 72,3%, dan ketika jumlah *server* diubah menjadi 5, maka besarnya *utilization* meningkat menjadi 92,44%.
4. Pada saat digunakan 1 *server*, diperoleh *average number of packet in the queue* terbesar, yaitu 324,67 paket. Namun ketika jumlah *server* ditambah menjadi 5 *server*, diperoleh *average number of packet in the queue* mengecil menjadi 194,5 paket.
5. Dari skenario untuk meneliti pengaruh penambahan ukuran paket terhadap nilai utilisasi, diperoleh hasil, bahwa dengan ukuran paket 1000, 2000, dan 3000 *byte*, nilai utilisasi ternyata semakin tinggi. .
6. Dari skenario penambahan ukuran paket, untuk meneliti pengaruhnya terhadap *average number of packet in the queue*, ternyata nilai *average number of packet in the queue* yang dihasilkan semakin kecil, untuk ketiga ukuran paket tersebut diatas (1000, 2000 dan 3000

byte). Penambahan *server* menyebabkan waktu pelayanan paket semakin cepat, sehingga jumlah rata-rata paket yang menunggu di dalam antrian menjadi berkurang.

## 5.2. Saran

Adapun beberapa saran yang dapat disampaikan, untuk penelitian lebih lanjut, adalah sebagai berikut :

1. Membuat algoritma untuk mengukur unjuk kerja jaringan berdasarkan variabel jumlah *server*,  $N$  (dari model antrian  $M/M/N$ ), menggunakan simulator jaringan seperti misalnya *Network Simulator*, OPNET, dan lain-lain.
2. Melakukan analisis kinerja jaringan dengan model antrian  $M/M/N$  yang belum dilakukan dalam penelitian ini, antara lain *throughput*, *packet drop*, dan *average queue length*.

## Referensi

- [1] Gross,D. & Harris, Carl M. 1998. *Fundamentals of Queueing Theory*. John Wiley & Sons,Inc., USA.
- [2] Ismail, M.N., & Zin, A.M. 2009. *Traffic Engineering: Simulation Model and Real Network Environment Over Single and Multiple Links*. Universitas Kuala Lumpur., Malaysia.
- [3] Iversen, Villy. 2001. *Teletraffic Engineering*. Technical University of Denmark, Denmark.
- [4] Kleinrock, Leonard. 1975. *Queueing Systems Volume1: Theory*. John Wiley & Sons,Inc., USA.
- [5] Kleinrock, Leonard. 1976. *Queueing Systems VolumeII: Computer Applications*. John Wiley & Sons,Inc., USA.
- [6] Korilis, Yanis A. 2003. *Networking Theory and Fundamentals*.John Wiley & sons, Inc., USA.
- [7] Sadiku,Matthew & Tofighi, Mohammad. 1999. *A Tutorial On Simulation Of Queueing Models*. Department of Electrical and Computer Engineering, Temple University, USA.