

# Performansi Algoritma *Proportional Integral Controller Enhanced* (PIE) dan *DropTail* Pada Layanan VoIP

Siti Amatullah Karimah<sup>#1</sup>, Aji Gautama Putrada<sup>\*2</sup>, Fadli Rianda<sup>#3</sup>

# *Fakultas Informatika, Universitas Telkom*

*Jl. Telekomunikasi No. 01, Terusan Buah Batu, Bandung 40257*

<sup>1</sup> [karimahsiti@telkomuniversity.ac.id](mailto:karimahsiti@telkomuniversity.ac.id), <sup>2</sup> [ajigps@telkomuniversity.ac.id](mailto:ajigps@telkomuniversity.ac.id)

<sup>3</sup> [sirvaith@student.telkomuniversity.ac.id](mailto:sirvaith@student.telkomuniversity.ac.id)

## Abstract

Voice over Internet Protocol (VoIP) is a communication technology that uses the Internet to conduct long-distance communication. VoIP develops along with the development of mobile devices and the Internet. This massive use of the Internet will cause the possibility of bufferbloat. Bufferbloat is an event where a large buffer will always be fully loaded and result in the length of the queue in the buffer. If this is allowed, the transmission delay will be high. One method that can be used to overcome this problem is Active Queue Management (AQM). This study presents the performance of 2 AQM pieces namely Proportional Integral controller Enhanced (PIE) and DropTail for VoIP services. Quality measurement was assessed using the calculation of Mean Opinion Score (MOS). Assessment is divided into two, namely subjective and objective. Subjectively, the value of MOS is obtained by listening to the sound quality directly. Objectively, the MOS value is obtained by R-Factor calculation. The results of the comparison of the value of Mean Opinion Score (MOS) both subjectively and objectively indicate that the quality of VoIP with the PIE algorithm is better than Droptail which is 30% and 20% better on subjective and objective judgments. Both MOS are taken from the smallest traffic shaping, which is 70 Kbps, which is the worst case of the tested scenario.

**Keywords:** MOS, VoIP, AQM, *Proportional Integral Controller Enhanced*, DropTail

## Abstrak

Voice over Internet Protocol (VoIP) merupakan teknologi komunikasi yang memanfaatkan Internet untuk melakukan komunikasi jarak jauh. VoIP berkembang seiring dengan berkembangnya *mobile device* dan Internet. Penggunaan jaringan Internet secara masif ini akan menimbulkan kemungkinan terjadinya bufferbloat. Bufferbloat merupakan kejadian dimana buffer yang berukuran besar akan selalu terisi penuh dan berakibat pada panjangnya antrian dalam buffer. Jika hal ini dibiarkan, maka delay transmisi menjadi tinggi. Salah satu metode yang dapat digunakan untuk mengatasi masalah ini adalah Active Queue Management (AQM). Penelitian ini menyajikan performansi 2 buah AQM yakni *Proportional Integral controller Enhanced* (PIE) dan *DropTail* untuk layanan VoIP. Pengukuran kualitas dinilai menggunakan perhitungan *Mean Opinion Score* (MOS). Penilaian terbagi menjadi dua yaitu subjektif dan objektif. Secara subjektif, nilai MOS didapatkan dengan mendengarkan langsung kualitas suara. Secara Objektif, nilai MOS didapatkan dengan perhitungan *R-Factor*. Hasil perbandingan nilai *Mean Opinion Score* (MOS) baik secara subjektif maupun objektif menunjukkan bahwa kualitas VoIP dengan algoritma PIE lebih baik daripada *Droptail* yakni 30% dan 20% lebih baik pada penilaian subjektif dan objektif. Kedua MOS ini diambil dari *traffic shaping* terkecil yakni 70 Kbps yang merupakan *worst case* dari skenario yang diujikan.

**Kata Kunci:** MOS, VoIP, AQM, *Proportional Integral Controller Enhanced*, DropTail

## I. PENDAHULUAN

**T**EKNOLOGI saat ini menjadi salah satu hal yang melekat dan tak terpisahkan dari kehidupan masyarakat. Salah satunya adalah Internet. Berbagai macam jenis data membanjiri Internet, bukan hanya data berupa teks, namun juga data multimedia yang bersifat *real time* dan membutuhkan penggaransian performansi, agar layanan dapat diterima dengan baik oleh pengguna. Salah satu layanan multimedia yang saat ini berkembang adalah layanan *Voice over Internet Protocol* atau yang biasa disebut VoIP. Pembanjiran data di Internet secara langsung akan mempengaruhi kualitas layanan VoIP. Teknologi ini memungkinkan pengguna untuk melakukan panggilan telepon melalui jaringan IP [2]. Aplikasi VoIP ini bersifat *realtime* sehingga membutuhkan *jitter* dan *delay* yang rendah. Pada saat ini sebagian besar penyedia layanan VoIP di internet lebih memilih protokol UDP dibanding TCP sebagai protokol *transport*. Tidak adanya *congestion control* pada UDP mengakibatkan *delay* dan *jitter* pada *receiver* menjadi besar. Hal ini menyebabkan masalah pada jaringan dan akan berdampak langsung pada pengguna layanan VoIP[3].

Paket data VoIP yang dirancang untuk mengalir secepat mungkin membuat antrian pada *buffer* cepat penuh. *Buffer* diperlukan untuk menampung sementara arus paket yang cepat. *Buffer* yang terus dialiri arus paket, lama-lama akan menjadi penuh dan mengakibatkan waktu penanganan paket menjadi sangat lama. Jika hal ini dibiarkan terus-menerus, akan menyebabkan *latency* yang tinggi, dimana fenomena ini biasa disebut *bufferbloat*. *Bufferbloat* merupakan salah satu kondisi *buffer* dengan ukuran besar dan cenderung selalu penuh dan menyebabkan antrian panjang di dalam *buffer*, jika hal ini dibiarkan maka akan menyebabkan jeda transmisi yang tinggi [4].

Solusi dari fenomena *bufferbloat* adalah dengan menggunakan *Active Queue Management* (AQM) yang berguna untuk memantau dan mengelola *queue*. Sebagian besar *router* saat ini menggunakan metode *first in-first out* (FIFO) yang biasanya disebut Droptail sebagai disiplin antriannya. Ada banyak macam *Active Queue Management* (AQM), seperti CoDel, PIE, RED, HRED, dsb. *Proportional Integral Controller Enhanced* (PIE) merupakan salah satu jenis AQM yang memiliki kualitas baik dalam menangani koneksi *real-time*.

*Proportional Integral Controller Enhanced* (PIE) merupakan salah satu algoritma *Active Queue Management* (AQM) dengan tujuan utama desainnya adalah melakukan kontrol terhadap *latency* agar tetap rendah, utilitas *link* yang tinggi, implementasi yang mudah, kestabilan terjamin dan respon yang cepat [5]. Pada penelitian ini akan meninjau kinerja PIE dalam mengatasi layanan *real-time* dengan mengimplementasikannya pada jaringan VoIP.

Pada penelitian sebelumnya dilakukan perbandingan kualitas layanan menggunakan algoritma CoDel, dimana pada penelitian sebelumnya, CoDel memiliki beberapa kekurangan diantaranya *drop behavior* yang selalu berubah setiap waktu yang akan berimbas pada *underutilization*. Algoritma PIE yang digunakan akan menyelesaikan permasalahan *drop behavior* ini dimana PIE mengadopsi kekuatan algoritma RED, dimana proses *dropping* dilakukan sebelum terjadinya *congestion*, sehingga akan meningkatkan performansi dan kualitas layanan.

## II. DASAR TEORI

### A. Voice Over Internet Protocol (VoIP)

Voice over Internet Protocol (VoIP) adalah integrasi dari layanan telepon konvensional dengan semakin bertambahnya jumlah aplikasi berbasis IP (*IP-based applications*). Teknologi ini merupakan salah satu teknologi yang sangat berkembang dan banyak didukung oleh penyedia layanan telekomunikasi[12]. Layanan ini memungkinkan pengguna membuat panggilan telepon melalui koneksi Internet.

VoIP bekerja dengan cara mengirimkan sinyal digital. Sebelum proses transmisi dilakukan, pertama-tama dilakukan konversi data berupa sinyal analog menjadi sinyal digital menggunakan ADC (*Analog to Digital*

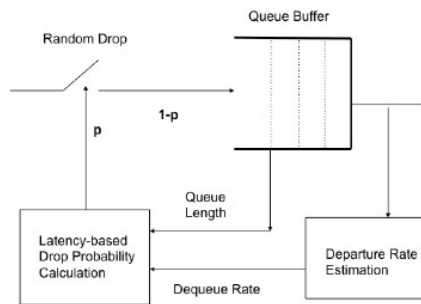
Converter). Setelah itu data berupa sinyal digital ini ditransmisikan ke sumber tujuan, untuk kemudian di konversi kembali menjadi sinyal analog di tujuan menggunakan DAC (*Digital to Analog Converter*)[13].

**B. Active Queue Management (AQM)**

*Active Queue Management* adalah suatu metode dimana perangkat jaringan dapat mengontrol panjang antrian atau lama waktu paket yang diperbolehkan untuk tetap berada di *buffer* pada perangkat. *Active Queue Management (AQM)* merupakan salah satu cara untuk menangani kondisi *buffer* dengan ukuran besar yang cenderung selalu penuh dan menyebabkan antrian panjang di dalam *buffer* yang dapat menyebabkan jeda tranmisi yang tinggi jika terjadi terus-menerus atau biasa disebut dengan *bufferbloat*[2].

**C. Proportional Integral Controller Enhanced (PIE)**

*Proportional Integral Controller Enhanced* atau yang lebih dikenal dengan PIE merupakan gabungan dari kelebihan algoritma RED dan CoDel yakni mudah diimplementasikan seperti RED dan tetap mengontrol *latency* seperti CoDel[10].



GAMBAR 1. SKEMA PIE

**a) Random Dropping**

Seperti pada kebanyakan AQM, PIE akan melakukan *drop* paket secara *random*, bergantung pada *drop probability* ( $p$ ), dimana  $p$  didapatkan dari komponen *drop probability calculation*.

**b) Drop Probability Calculation**

Berikut adalah *drop probability* pada algoritma PIE :

- Estimasi *delay* pada *queue*

$$cur_{del} = \frac{qlen}{avg_{drate}} \tag{1}$$

- Menghitung nilai  $p$

$$p = p + a * (cur_{del} - ref_{del}) + \beta * (cure_{del} - old_{del}) \tag{2}$$

- Update *delay*

$$old_{del} = cur_{del} \tag{3}$$

**c) Departure Rate Estimation**

Pada saat paket dikirim :

- Memutuskan untuk berada dalam siklus pengukuran, jika :  
 $qlen > dq_{threshold}$ ;
- Apabila nilai di atas *True*, perbarui hitungan keberangkatan  $dq_{count}$  :
- Perbarui *departure rate*  $dq_{count} > dq_{threshold}$  dan reset penghitungan ;

$$dq_{count} = dq_{count} + dq_{pktzise}; \quad (4)$$

$$\begin{aligned} dq_{int} &= now - start; \\ dq_{rate} &= \frac{dq_{count}}{dq_{int}}; \\ avg_{drate} &= (1 - \epsilon) * avg_{drate} + \epsilon * dq_{rate}; \\ start &= now; \\ dq_{count} &= 0 \end{aligned} \quad (5)$$

**d) Handling Bursts**

Tiga komponen di atas adalah komponen dasar dari algoritma PIE. Berikut ini adalah perhitungan *burst allowance* dari algoritma PIE,

Pada saat paket diterima :

- If  $burst\_allow > 0$   
*Enque packet bypassing rabdin drop;*

Pada saat  $dq_{rate}$  diperbarui

- *Update burst allowance:*
- If  $p = 0$ ; and both  $cur_{del}$  and  $old_{del}$  less than  $ref_{del}/2$ , reset  $burst_{allow}$ ,  
 $burst_{allow} = max\_burst$ ;

Jadi semua komponen di atas adalah komponen penting pada skema algoritma PIE untuk mengontrol *latency* tetap rendah dan untuk menjaga tidak terjadinya *bufferbloat*. Pada Linux (dengan kernel 3.5 atau di atasnya), modul kernel PIE dapat diterapkan menggunakan *tc* atau *traffic control*.

**D. Droptail**

*Droptail* adalah algoritma AQM yang sangat sederhana dan merupakan konfigurasi *default* pada kebanyakan *router*. AQM ini tidak membedakan *traffic* dari sumber yang berbeda. Selama antrian terisi, paket berikutnya akan di *drop*. Dengan kata lain, AQM ini akan melakukan *drop* ekor (*tail*) dari urutan paket.

Ide dari AQM *Droptail* ini adalah paket pertama yang tiba di *router* adalah paket pertama yang akan dikirimkan. Mengingat bahwa jumlah ruang *buffer* yang terbatas pada setiap *router*, jika sebuah paket tiba dan antrian penuh, maka *router* akan membuang paket tersebut. Hal ini dilakukan tanpa mempedulikan jenis aliran atau seberapa pentingnya paket tersebut. Kinerja sistem antrian dengan karakteristik M/M/1 tergantung pada parameter berikut:

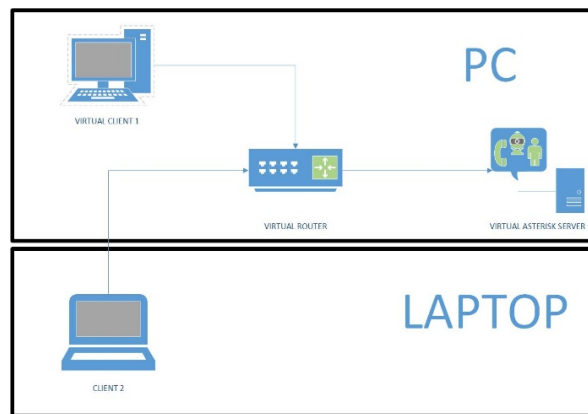
- a. Tingkat kedatangan paket.
- b. Ukuran paket.
- c. Kapasitas layanan.

Antrian M/M/1 umumnya digambarkan oleh proses Poisson yang mengatur kedatangan paket ke *buffer* terbatas. Ketika sebuah paket mencapai kepala *buffer*, diproses oleh server dan dikirim ke tujuannya [14].

### III. PERANCANGAN SISTEM

#### A. Topologi Jaringan

Pada penelitian ini dibangun sistem yang dapat melakukan perhitungan nilai *Mean Opinion Score* (MOS) dan *R-Factor* pada jaringan VoIP. Sistem ini dibangun menggunakan VMware Workstation dengan beberapa node yaitu *client 1*, *client 2*, *router*, dan *server Asterisk* dimana *client 1*, *router*, *server Asterisk* berada pada 1 *Personal Computer* (PC) dan *client 2* pada Laptop.



GAMBAR 2. TOPOLOGI

Pada *interface router* untuk *client 1* dan *client 2*, *bandwidth* akan disesuaikan sampai mendapatkan angka yang menunjukkan perbedaan yang jelas antara menggunakan AQM PIE dan Droptail. *Active Queue Management* sendiri diatur pada *router*. Pada linux, modul kernel PIE dapat diterapkan menggunakan *tc* atau *traffic control*.

- a) **Client**  
*Client* berupa komputer virtual yang menggunakan *operating system* Linux yang di *install software softphone* Zoiper.
- b) **Server Asterisk**  
Asterisk diimplementasikan pada secara *virtual* menggunakan *operating system* Linux.
- c) **Router**  
*Router* menggunakan Linux pada sebuah komputer *virtual*.

#### B. Skenario Pengujian

Skenario yang digunakan dalam penelitian ini yakni *router* terhubung dengan *client* dan *server asterisk*. Pengujian dilakukan menggunakan *traffic shaping* dengan mengatur *bandwidth* pada setiap *interface router* yang terhubung ke *client1* dan *client2*. Pengaturan *bandwidth* dilakukan sedemikian rupa agar terjadi *bottleneck*.

##### a) Kondisi *bottleneck* 90kbps dan 90kbps

Skenario ini diimplementasikan pada *router* yang terhubung pada *client1* dan *client2*. Fungsi *bottleneck* diatur pada setiap *interface* yang terlibat antara *client* dan *router*. *Interface* yang menghubungkan *router* dengan *client1* diatur dengan kecepatan 90kbps dan *interface* yang menghubungkan *router* dengan

*client2* diatur dengan kecepatan 90kbps. Percobaan panggilan VoIP dilakukan dari *client1* ke *client2* dengan mengimplementasikan algoritma DropTail dan PIE pada sisi *router*.

**b) Kondisi bottleneck 80kbps dan 80kbps**

Skenario ini di implementasikan pada *router* yang terhubung pada *client1* dan *client2*. Fungsi *bottleneck* di atur pada setiap *interface* yang terlibat antara *client* dan *router*. *Interface* yang menghubungkan *router* dengan *client1* diatur dengan kecepatan 80kbps dan *interface* yang menghubungkan *router* dengan *client2* diatur dengan kecepatan 80kbps. Percobaan panggilan VoIP dilakukan dari *client1* ke *client2* dengan mengimplementasikan algoritma DropTail dan PIE pada sisi *router*.

**c) Kondisi bottleneck 75kbps dan 75kbps**

Skenario ini di implementasikan pada *router* yang terhubung pada *client1* dan *client2*. Fungsi *bottleneck* di atur pada setiap *interface* yang terlibat antara *client* dan *router*. *Interface* yang menghubungkan *router* dengan *client1* diatur dengan kecepatan 75kbps dan *interface* yang menghubungkan *router* dengan *client2* diatur dengan kecepatan 75kbps. Percobaan panggilan VoIP dilakukan dari *client1* ke *client2* dengan mengimplementasikan algoritma DropTail dan PIE pada sisi *router*.

**C. Parameter Pengujian**

Terdapat tiga parameter uji yang digunakan dalam menganalisa mekanisme *Proportional Integral Controller Enhanced* (PIE) dan DropTail

**a. Delay**

*Delay* adalah waktu yang dibutuhkan paket data untuk ditransmisikan dari pengirim ke penerima. Kualitas delay dikatakan baik apabila waktu tunda yang terjadi hanya sekitar 0 – 150 ms [2][3]. Rata – rata delay dapat dihitung menggunakan persamaan (6).

$$\text{rata – rata delay} = \frac{\text{total delay}}{\text{total paket diterima}} \quad (6)$$

**b. Throughput**

*Throughput* merupakan kecepatan transfer data efektif, diukur dalam satuan bps (bit per second). *Throughput* adalah jumlah total kedatangan paket yang sukses, diambil dari sisi tujuan selama *interval* waktu tertentu dibagi oleh durasi *interval* waktu transmisi [3]. *Throughput* dapat dihitung menggunakan persamaan (7).

$$\text{throughput} = \frac{\text{paket data diterima}}{\text{lama pengamatan}} \quad (7)$$

**c. Packet Loss**

*Packet Loss* merupakan jumlah paket yang hilang dalam suatu pengiriman paket data ketika di transmisikan di jaringan. *Packet loss* terjadi oleh beberapa sebab diantaranya *noise*, *collision* dan *congestion* yang disebabkan oleh ketidakmampuan *buffer* menampung antrian paket pada jaringan. *Packet Loss* pada VoIP dikatakan baik apabila berkisar antara 0 s/d 0.5% dari pengiriman data [2][19]. *Packet loss* dapat dihitung menggunakan persamaan (8).

$$\text{packet loss} = \frac{(\text{paket data dikirim} - \text{paket data diterima}) \times 100\%}{\text{paket data yang dikirim}} \quad (8)$$

d. *R-factor*

*R-Factor* merupakan hasil pengukuran dari *E-Model* yang dipergunakan untuk memprediksi nilai dari *Mean Opinion Score* (MOS) secara objektif. *E-Model* itu sendiri merupakan pengukuran secara objektif pada jaringan telekomunikasi yang diperkenalkan oleh ETSI dan distandarkan oleh ITU-T G.107 [4]. Berikut rumus untuk memperoleh MOS secara objektif menggunakan persamaan (9).

$$R = 94.2 - I_d - I_{ef} \tag{9}$$

Yang mana :

$I_d$  = faktor dari penurunan kualitas suara oleh suatu *delay*.

$I_{ef}$  = faktor dari penurunan kualitas karena teknik kompresi dan *packet loss*.

$$I_d = 0,024d + 0,11(d - 177,3) \times H_{(d-177,3)}$$

$$I_{ef} = 7 + 30 \ln(1 + 15e)$$

Yang mana :

$R$  = faktor kualitas pengiriman

$d$  = *delay*

$H$  = fungsi *heavyside*, yang mana:

$$H_{(x)} = 0, \text{ if } x < 0, \text{ dan}$$

$$H_{(x)} = 1, \text{ if } x \geq 0$$

$e$  = persentase terjadi *packet loss*

Konversi nilai  $R$  ke MOS, berikut aturan yang harus digunakan:

- $R < 0$ ;  $MOS = 1$
- $R > 100$ ;  $MOS = 4,5$
- $0 < R < 100$ ;  $MOS = 1 + 0,035R + (7 \times 10^{-6})R(R - 60)(100 - R)$

e. Nilai *Mean Opinion Score*

*Mean Opinion Score* (MOS) merupakan sebuah penilaian secara subjektif dalam mengukur kualitas suara yang direkomendasikan oleh ITU-T pada tahun 1996. Nilai MOS diperoleh dengan menilai langsung kualitas suara yang didengar [6]. Berikut nilai MOS beserta kriteria kualitas dapat dilihat pada tabel berikut

TABEL I  
NILAI KUALITAS MOS

MOS	Kualitas
5	Sangat Baik
4	Baik
3	Cukup
2	Buruk
1	Sangat Buruk

IV. EVALUASI SISTEM

Pada evaluasi ini menjelaskan hasil yang didapat dari percobaan sesuai dengan skenario yang telah dipaparkan sebelumnya. Berikut hasil percobaan terhadap *delay*, *throughput*, *packet loss*, *R-factor*, dan Nilai *Mean Opinion Score* (MOS).

A. Performansi Drop Tail

TABEL II  
 PERFORMANSI DROP TAIL

Traffic Shaping (Kbps)	Rata-Rata Delay (ms)	Throughput (Mbit/s)	Dropped Packet		
			Receive	Expected	Loss
90	19.986686	0,168	11020	11020	0%
80	41.797767	0,112	8157	10996	25.82%
70	99.270147	0.08	4582	9297	50,72%

Pada tabel 2 menjelaskan peformansi Droptail pada setiap skenario *bottleneck* yaitu 90 Kbps, 80 Kbps, 70 Kbps. Pada tabel 2 terlihat bahwa semakin kecil nilai *traffic shaping* maka semakin besar nilai rata-rata *delay* dan *packet loss*, sedangkan nilai *throughput* semakin kecil. Hal ini menjadi sangat wajar mengingat semakin kecil *bandwidth* yang tersedia, semakin tinggi pula *latency* yang akan dialami sebab jaringan menjadi lebih padat.

B. Performansi Proportional Integral Controller Enhanced (PIE)

TABEL III  
 PERFORMANSI PIE

Traffic shaping (Kbps)	Rata-Rata Delay (ms)	Throughput (Mbit/s)	Dropped Packet		
			Receive	Expected	Loss
90	19.994444	0,168	11017	11017	0%
80	25.927081	0,128	9529	10761	11.45%
70	62.84687	0,096	4093	6786	39.68%

Pada tabel 3 menjelaskan peformansi Proportional Integral Controller Enhanced (PIE) pada setiap skenario *bottleneck* yang diujikan. Pada table tersebut juga terlihat semakin kecil *traffic shaping*, semakin besar nila rata-rata *delay* dan *packet loss*, sedangkan nilai *throughput* semakin kecil.

C. Nilai Mean Opinion Score (MOS)

Pada tabel 4 menjelaskan tentang kualitas suara secara subjektif. Survei dilakukan terhadap 30 orang. Pemilihan 30 orang ini didasarkan pada referensi [14] yang menyebutkan bahwa 30 merupakan jumlah *sample* yang dapat mewakili dengan rentang usia responden berkisar antara 18 – 20 tahun dan memiliki pendengaran normal. 30 orang ini mendengarkan rekaman dari semua skenario (skenario dengan *traffic shaping* sebesar 90kbps, 80kbps dan 70kbps) yang diujikan, lalu mengisi nilai antara 1 – 5. Pada skenario 1 nilai MOS tidak terlalu signifikan, tetapi pada skenario 2 dan 3 mulai terlihat nilai yang cukup signifikan.



TABEL IV  
NILAI MOS

No	PIE			Droptail		
	90 Kbps	80 Kbps	70 Kbps	90 Kbps	80 Kbps	70 Kbps
1	4	3.5	1.5	4	3	1
2	4	3	3	4	2	2
3	5	3	2	5	2	1.5
4	5	3	2	4	2	1
5	4	3	2	4	2	1
6	4	3	2	4	2	2
7	4	3	3	4	3	3
8	5	3.5	2	4	3	1
9	5	3	3	5	2	3
10	5	4	2	4	2	2
11	4	3	2	4	2	1
12	5	3	1	5	3	1
13	5	3	2	5	2	1.5
14	4	3	3	4	2	2
15	4	3	2	3	2	2
16	5	3	2	4	3	2
17	4	3	1	3	2	1
18	3	2	1	3	2	1
19	3.5	2.5	1	3	2	1
20	4	3	2	3.5	2.5	1.5
21	3.5	3	1.5	3	2.5	1
22	3	2	1	3	2	1
23	5	3	2	5	3	2

24	5	3	2	4	2	1
25	5	3	1	5	3	1
26	4	3	2	4	2	2
27	5	4	2	4	2	2
28	4.5	3	1.5	4	3	1
29	5	3	2	5	2	1
30	3	1.5	0.5	3	1	0.5
Mean	4.317	2.967	1.833	3.983	2.267	1.467

D. Nilai R-Factor

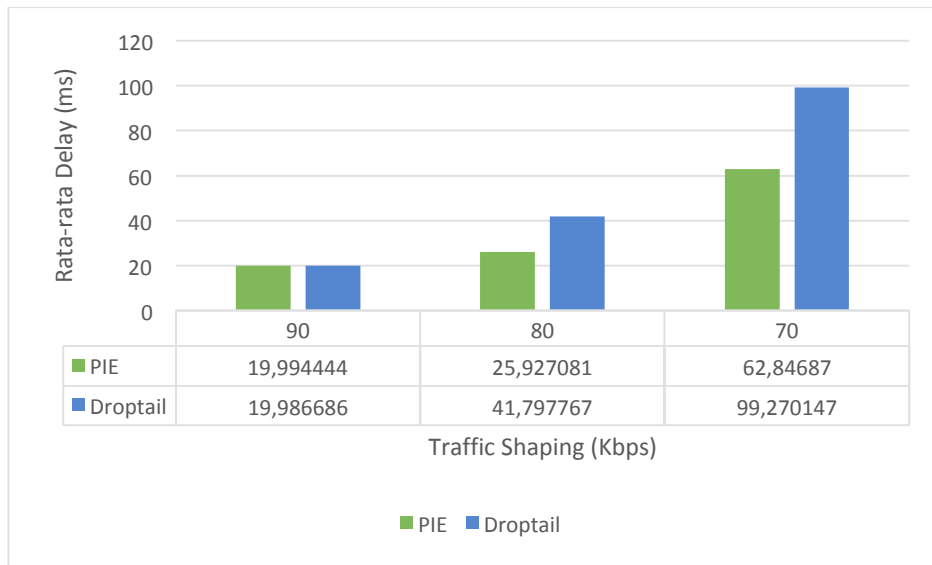
TABEL V  
 NILAI R-FACTOR

Traffic shaping (Kbps)	R-Factor	
	PIE	Droptail
90	86.720	86.720
80	56.586	38.686
70	27.521	20.237

Pada tabel 5 menjelaskan tentang nilai dari R-Factor dari semua skenario yang diujikan. Sama seperti nilai *Mean Opinion Score* (MOS), perbedaan yang signifikan terlihat pada skenario 2 dan 3.

E. Perbandingan Drop Tail dan PIE

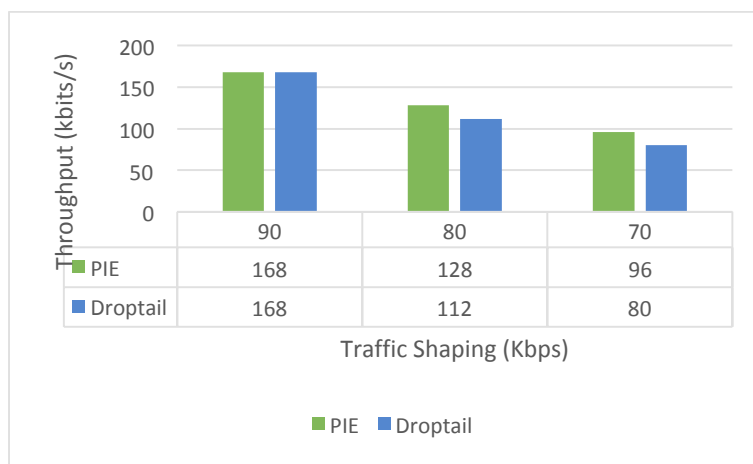
a) Delay



GAMBAR 3. RATA-RATA DELAY PIE DAN,DROPTAIL

Gambar 3 menjelaskan bahwa pada skenario 1 (90 Kbps) perbedaan *delay* tidak terlalu signifikan, akan tetapi pada skenario 2 (80 Kbps) dan skenario 3 (70 Kbps) perbedaan *delay* mulai terasa. Hal ini dikarenakan paket RTP sangat kecil yaitu 85 Kbps sehingga pada skenario 1 tidak terasa perbedaannya. Pada skenario 2 dan skenario 3 algoritma PIE menjaga *delay* agar tetap rendah. Hal ini disebabkan karena algoritma PIE menjaga *latency* tetap rendah dengan cara menggunakan *timestamp* untuk menandai setiap paket dalam antrian. Ketikan paket melebihi *timestamp*, maka paket tersebut akan di-*drop* untuk menjaga antrian tidak penuh dan menjaga *delay* tetap rendah.

**b) Throughput**

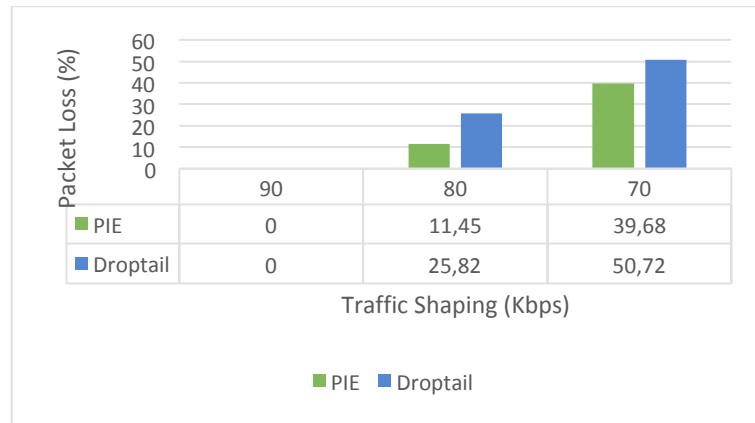


GAMBAR 4. THROUGHPUT PIE DAN DROPTAIL

Gambar 4 menjelaskan peformansi kedua algoritma dalam menangani throughput, algoritma PIE lebih besar daripada droptail. Pada setiap skenario memiliki nilai *throughput* yang berbeda, hal ini dikarenakan

*bandwidth* yang diberikan. Jadi semakin besar *bandwidth* maka *throughput* juga akan semakin besar. Perbedaan *throughput* antara kedua antara PIE dan Droptail tidak terlalu signifikan karena RTP yang sangat kecil. Nilai *throughput* pada PIE lebih bagus karena PIE menjaga *latency* tetap rendah pada *router*.

**c) Packet Loss**



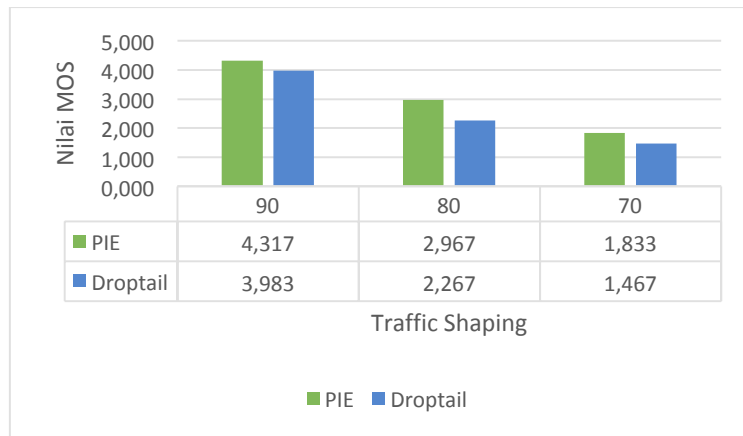
GAMBAR 5. PACKET LOSS PIE DAN DROPTAIL

Gambar 5 menjelaskan tentang perbandingan *packet loss* pada setiap scenario yang diujikan. Pada skenario 1 tidak terjadi *packet loss* karena kecilnya paket RTP itu sendiri, sehingga paket langsung diantarkan dari *client 1* ke *client 2*. *Packet loss* mulai terjadi pada skenario 2 (80 Kbps) dan skenario 3 (70 Kbps) karena data mulai terjadi *congestion* pada *router* disebabkan *bandwidth* yang disediakan sangat kecil. Algoritma PIE menangani *packetloss* lebih baik karena PIE memiliki hitungan *Drop Probability* dengan menjaga *latency* tetap rendah dan menjaga antrian dalam *router* tetap stabil. sedangkan algoritma Droptail akan membuang (*drop*) semua paket apabila antrian dalam *router* penuh.

**d) Nilai Mean Opinion Score (MOS)**

Terdapat dua macam nilai *Mean Opinion Score* (MOS) pada penelitian ini,yaitu secara subjektif dan objektif. Secara subjektif nilai langsung diberikan setelah mendengar semua skenario dimana pada penelitian ini nilai diambil dari rata-rata nilai yang diberikan oleh 30 orang. Secara Objektif nilai MOS diambil dari nilai R-Faktor.

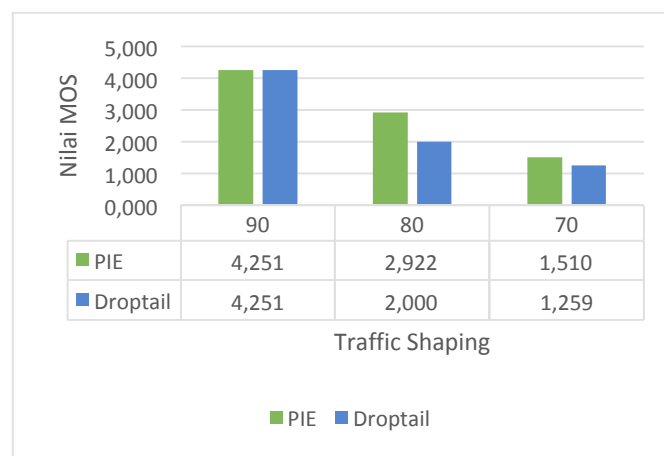
- Subjektif



GAMBAR 6. MOS SUBJEKTIF PIE DAN DROPTAIL

Pada gambar 6 terlihat jelas perbedaan nilai MOS antara PIE dan Droptail. Hasil ini diambil dari rata-rata total nilai dari penilaian dari 30 orang yang mendengarkan langsung rekaman *call* VoIP pada setiap skenario. Algoritma PIE mendapatkan nilai MOS lebih baik daripada Droptail, hal ini juga mengacu dari hasil parameter *delay*, *throughput*, dan juga *packetloss* dimana PIE lebih baik daripada Droptail. Jadi semakin bagus penanganan *throughput*, *delay*, dan *packetloss* maka semakin bagus juga kualitas yang dihasilkan.

• **Objektif**



GAMBAR 7. MOS OBJEKTIF PIE DAN DROPTAIL

Pada gambar 7 merupakan nilai *Mean Opinion Score* (MOS) yang didapatkan dari nilai *R-factor* yang dapat dilihat pada tabel 5. Dapat dilihat secara objektif nilai PIE juga lebih baik daripada Droptail sesuai dengan nilai subjektif yang didapatkan seperti gambar 5. Nilai yang didapatkan juga tidak jauh berbeda. Hal ini membuktikan bahwa dalam memaksimalkan kualitas layanan VoIP, menggunakan algoritma PIE lebih baik daripada Droptail.

## V. KESIMPULAN

Tujuan penelitian ini adalah untuk menguji kualitas MOS pada algoritma *Proportional Integral Controller Enhanced* (PIE) dan Droptail pada jaringan *Voice over Internet Protocol* (VoIP). Hasil perbandingan antara dua algoritma tersebut telah di sajikan pada sub bab 4.5. Dari hasil pengujian terlihat bahwa semakin kecil *traffic shaping* yang diimplementasikan, semakin turun performansinya, baik dilihat dari MOS secara subjektif dan objektif, *throughput*, *packet loss* maupun *delay*. Pada skenario *traffic shaping* 90 Kbps, perbedaan kualitas layanan VoIP baik menggunakan algoritma PIE maupun Drop Tail memiliki kualitas yang setara. Namun ketika dilakukan *traffic shaping* 80 Kbps dan 70 Kbps terlihat bahwa kualitas layanan PIE mengungguli Drop Tail. Pada kasus terburuk, yakni ketika dilakukan *traffic shaping* sebesar 70Kbps, terlihat jelas bahwa perbandingan PIE memiliki nilai MOS yang lebih besar 30% secara subjektif dan lebih besar 20% secara objektif. Baiknya nilai MOS yang dihasilkan algoritma PIE disebabkan oleh sistem *random dropping* yang membuang paket secara acak sehingga kualitas suara lebih jelas, sedangkan Droptail akan membuang semua paket apabila antrian sudah penuh.

## DAFTAR PUSTAKA

- [1] Z.Nasution. (2014). <http://repository.ut.ac.id>. Retrieved from <http://repository.ut.ac.id>: <http://repository.ut.ac.id/4482/2/SKOM4322-TM.pdf>
- [2] Satria Mandala, M. N. (2017, Maret 1). Actove Queue Management (AQM) Performance Analysis based on Controlled Delay (CoDel) Algorithm on UDP. *Indonesian Journal of Computing*, 2, 1. doi:10
- [3] A Nur Lina, 2014, Congestion (pada jaringan data), (online) <http://nurlinaamik.blogspot.co.id/2014/06/congestion-pada-jaringan-data.html>. Diakses : 10 November 2017.
- [4] La Surimi, R. P. (2015, July). Analisis Kualitas VoIP pada SCTP Menggunakan ECN dan AQM. *Indonesian Journal of Computing and Cybernetics Systems* , 9, 123.
- [5] Alilied Telesis "QoS White Paper". <https://www.alliedtelesis.com/documents/qos-white-paper>. Diakses : 12 November 2017.
- [6] Ginanjar Rahmansyah, T. B. (2015). Analisis Mekanisme Active Queue Management (AQM) Berbasis Controlled Delay., II, p. 1145. Bandung. Retrieved 8 14, 2018
- [7] ITU-T Series G. 2003. "Transmission system and media, Digital systems and networks".
- [8] Andrew, 2011, "Difference Between R-Factor and MOS Score in VoIP Quality". <http://www.differencebetween.com/difference-between-r-factor-and-mos-score-in-voip-quality/>. Diakses : 12 November 2017.
- [9] *International Conference on High Performance Switching and Routing* (pp. 148-155). Taipei: IEEE. doi:10.1109/HPSR.2013.6602305
- [10] ISKANDAR, M. N. (2017). ANALISIS PERFORMANSI ACTIVE QUEUE MANAGEMENT (AQM) BERBASIS ALGORITMA CONTROL DELAY (CoDel) PADA STREAMING USER DATAGRAM PROTOCOL (UDP). (S. M. MUH. ARIF NUGROHO, Ed.) Bandung, Indonesia: Telkom.
- [11] Nobuhiko Kitawaki, A. T. (2004, July 28). Perceptual QoS Assessment. *erceptual QoS assessment technologies for VoIP*, 28 - 34. doi:10.1109/MCOM.2004.1316526
- [12] Setiawan, E. B. (2012, October). ANALISA QUALITY OF SERVICES (QoS) VOICE OVER INTERNET PROTOCOL (VoIP) DENGAN PROTOKOL H.323 DAN SESSION INITIAL PROTOCOL (SIP). *ANALISA QUALITY OF SERVICES (QoS) VOICE OVER INTERNET PROTOCOL (VoIP) DENGAN PROTOKOL H.323 DAN SESSION INITIAL PROTOCOL (SIP)*, 7 - 8. Retrieved from <https://ojs.unikom.ac.id/>
- [13] Misbahul, F. (2016, November). Simulasi Antrian Paket Data Jaringan dengan Mekanisme Drop Tail. *Jurnal Ilmiah FIFO, VIII*, 151 - 160. Retrieved 11 10, 2017
- [14] Hogg and Tanis.2013.Probability and Statistical Inference (9e). USA :Pearson.