

## Perbandingan *Mean Opinion Score* (MOS) dari VoIP menggunakan *Controlled Delay* (CoDel) & *DropTail*

Syafwan Almadani Azra<sup>1</sup>, Aji Gautama Putrada Satwiko<sup>2</sup>, Siti Amatullah Karimah<sup>3</sup>

<sup>1,2,3</sup>Fakultas Informatika, Universitas Telkom, Bandung

---

### Abstrak

*Voice over Internet Protocol* (VoIP) merupakan sebuah teknologi yang memungkinkan terjadi komunikasi jarak jauh dengan memanfaatkan jaringan internet sebagai penghubung. Perkembangan VoIP saat ini sangat lah pesat karena *trend* komunikasi saat ini dikuasai oleh *smartphone*. Hal tersebut mengakibatkan terjadinya *congestion* pada jaringan seiring dengan meningkatnya penggunaan layanan VoIP pada *smartphone*. Permasalahan ini dapat diatasi dengan menerapkan mekanisme antrian pada layanan VoIP dalam mengatasi antrian paket data. Mekanisme antrian ini disebut sebagai *Active Queue Management* (AQM). *Active Queue Management* (AQM) menyediakan berbagai macam mekanisme antrian seperti *Controlling Delay* (CoDel) dan *DropTail* yang bertujuan untuk mengurangi terjadinya *congestion*. Dalam penelitian ini diimplementasikan dan dianalisis kualitas layanan VoIP dengan menerapkan *Controlled Delay* (CoDel) dan *DropTail* berdasarkan perhitungan *delay*, *throughput*, *packet loss*, dan *Mean Opinion Score* (MOS) yang didapatkan. Hasil pengujian menunjukkan performansi algoritma CoDel lebih baik jika dilihat dari nilai *delay* dan *throughput* yang didapat, sedangkan algoritma Droptail secara meyakinkan lebih baik dalam penanganan *packet loss*. Jika dilihat dari perbandingan nilai MOS, algoritma DropTail lebih baik dari algoritma CoDel secara subjektif ataupun secara objektif. Ini mengindikasikan implementasi algoritma CoDel terhadap layanan VoIP masih lebih buruk daripada DropTail

**Kata kunci :** VoIP, AQM, CoDel, DropTail, MOS, *congestion*

---

### Abstract

*Voice over Internet Protocol* (VoIP) is a technology that allows remote communication occurs by utilizing the internet as a conduit. Development of VoIP is currently very rapidly because the communication trend is currently controlled by a smartphone. This led to congestion on the network along with the increasing use of VoIP services on Smartphones. This problem can be overcome by implementing mechanisms of queue on VoIP services in addressing data packet queue. This queue mechanism referred to as the *Active Queue Management* (AQM). *Active Queue Management* (AQM) provides a variety of mechanisms such as *Controlling queue Delay* (CoDel) and *DropTail* aimed at reducing the occurrence of congestion. In this study are implemented and analyzed the quality of VoIP services by applying *Controlled Delay* (CoDel) and *DropTail* calculation based on *delay*, *throughput*, *packet loss*, and *Mean Opinion Score* (MOS) obtained. The test results show the performance of the algorithm CoDel better if viewed from the value of the *delay* and *throughput* obtained, while the algorithm Droptail conclusively better in handling *packet loss*. If seen from a comparison of the value of the MOS, *DropTail* algorithm better than algorithms CoDel subjectively or objectively. This indicates the algorithm implementation CoDel against VoIP service is still worse than a *DropTail*

**Keywords:** VoIP, AQM, CoDel, DropTail, MOS, *congestion*

---

### 1. Pendahuluan

#### Latar Belakang

Banyak hal yang bisa diperbuat dengan menggunakan layanan internet. Bisa dilihat dari perkembangan teknologi komunikasi saat ini dimana komunikasi jarak jauh seperti telepon dapat dilakukan dimana saja dengan bantuan internet tanpa terbatas fisik telepon yang dulunya harus tersambung dengan kabel telepon. *Voice over Internet Protocol* (VoIP) merupakan sebuah teknologi yang digunakan dalam mentransmisikan percakapan berbasis suara melalui jaringan internet menggunakan *internet protocol* (IP)[1]. Teknologi ini sangat berkembang dan diminati saat ini dikarenakan kepopuleran *smartphone* yang saat ini dipakai hampir semua kalangan masyarakat. Tidak hanya itu, aplikasi bertukar pesan pada *smartphone* saat ini memungkinkan pengguna untuk saling berkomunikasi melalui suara layaknya telepon konvensional.

Meningkatnya pengguna layanan VoIP menyebabkan terjadinya *congestion* pada jaringan dikarenakan besarnya data yang di transmisikan dan diterima dari satu *node* ke *node* lainnya dalam waktu bersamaan sulit untuk dikendalikan. *Congestion* merupakan sebuah situasi pada jaringan ketika terlalu banyak *source* mengirimkan terlalu banyak data dimana hal ini terlalu cepat bagi jaringan untuk ditangani [7]. *Congestion* sangat mungkin terjadi pada layanan VoIP yang berakibat pada terjadinya *bottleneck* pada jaringan. *Bottleneck*

yang terjadi berakibat pada data-data yang dikirimkan berada terlalu lama dalam *buffer*. Terjadinya penumpukan data-data didalam *buffer* biasanya disebut dengan fenomena *bufferbloat*. Hal ini berdampak buruk pada kualitas layanan VoIP karena meningkatnya *delay* dan *packet loss* [6].

*Active Queue Management* (AQM) merupakan salah satu pendekatan yang dapat mengatasi masalah *bufferbloat* yang terjadi pada layanan VoIP. *Controlling Delay* (CoDel) merupakan salah satu algoritma dari AQM yang diusulkan untuk mengatasi permasalahan *bufferbloat* yang berfokus pada *queuing delay* didalam jaringan internet [9]. CoDel dianggap dapat mengatasi permasalahan *bufferbloat* yang terjadi pada layanan VoIP dikarenakan kemampuan CoDel dalam menjaga *delay* agar selalu tetap kecil. Dengan mekanisme *timestamp* yang diterapkan CoDel memungkinkan setiap paket yang di-*drop* oleh antrian dapat kembali lagi pada antrian agar dapat meminimalisir terjadinya *packet loss* [5][10].

Guna mengetahui seberapa baik kualitas VoIP dengan mengimplementasikan algoritma CoDel dibutuhkan *Quality of Service* (QoS) sebagai tolak ukur. Parameter QoS yang dianalisa adalah *delay*, *throughput*, *packet loss* dan *Mean Opinion Score* (MOS). MOS sendiri merupakan sebuah standar penilaian kualitas suara yang mengacu kepada *International Telecommunication Union* (ITU). Nilai MOS sendiri biasanya didapatkan berdasarkan pendapat koresponden yang telah menilai suatu kualitas suara berdasarkan lima poin parameter yang telah ditetapkan sebelumnya [2][6].

### Topik dan Batasannya

Berdasarkan latar belakang diatas, terdapat permasalahan yang akan diangkat dalam penelitian ini yaitu penanganan *bufferbloat* yang berakibat pada kualitas layanan VoIP dengan menerapkan algoritma *controlled delay* (CoDel) dan *droptail* dalam sebuah *router* yang berperan dalam mengatur data paket antara *client1* dan *client2*. Pada penelitian ini node yang digunakan berupa 4 buah node yang terdiri dari 3 buah node (*router*, *client1*, *server asterisk*) yang berada pada satu *personal computer* yang di virtualisasi menggunakan *VMWare*. Node lainnya yaitu *client2* yang di virtualisasi pada perangkat *laptop* menggunakan *Vmware*. Monitoring data paket yang melalui *router* akan dilakukan menggunakan *wireshark* yang dijalankan pada *host personal computer*. Data paket yang di amati berupa *Real-Time Transport Protocol* (RTP) yang terkirim dari *client1* menuju *client2* melalui *router*. Protokol VoIP yang digunakan setiap *client1* dan *client2* merupakan protokol SIP dan menggunakan *codec* G.711 u-Law. Penilaian kualitas layanan berdasarkan nilai *delay*, *throughput*, *packet loss* dan *Mean Opinion Score* (MOS).

### Tujuan

Tujuan dari penelitian Tugas Akhir ini adalah untuk menganalisis kualitas layanan VoIP berdasarkan *Mean Opinion Score* (MOS), *delay*, *throughput*, dan *packet loss* yang menerapkan algoritma *controlled delay* (CoDel) dan *droptail* pada *router* sebagai perbandingan.

### Organisasi Tulisan

Penulisan tugas akhir ini disusun dalam beberapa bagian yaitu Bagian 1 – pendahuluan berisi pendahuluan untuk menjelaskan hal yang dilakukan pada penelitian ini, Bagian 2 – Studi Terkait berisi tentang jurnal atau penelitian yang berkaitan dengan penelitian yang sedang dilakukan kali ini sehingga bisa dijadikan referensi untuk penelitian ini, Bagian 3 – Sistem yang dibangun berisi penjelasan tentang spesifikasi dan alur kerja pada sistem yang dibangun, Bagian 4 – Evaluasi berisi tentang analisis dan evaluasi dari hasil pengujian yang telah dilakukan, Bagian 5 – Kesimpulan yang didapatkan dari hasil penelitian dan saran membangun untuk penelitian kedepannya.

## 2. Studi Terkait

### 2.1. Voice over Internet Protocol (VoIP)

*Voice over Internet Protocol* (VoIP) merupakan sebuah bentuk komunikasi yang memungkinkan melakukan telepon melalui koneksi jaringan internet. Secara dasar VoIP juga memungkinkan membuat dan menerima panggilan telepon melalui jaringan internet[1].

Aplikasi VoIP biasanya berkerja sebagai berikut. Pengiriman suara dilakukan dengan mengirimkan sebuah data yang berbentuk sinyal analog. Sinyal analog tersebut akan dikonversikan terlebih dahulu dengan Analog to Digital Converter (ADC) yang merubah bentuk sinyal analog ke bentuk digital. Setelah proses konversi dilakukan data digital akan dikirimkan ke tujuan/penerima. Setelah sinyal digital diterima, data tersebut dikonversikan kembali menjadi data sinyal analog dengan Digital to Analog Converter (DAC) agar dapat diterima oleh penerima sesuai dengan data sinyal yang ditransmisikan[2].

### 2.2. Active Queue Management (AQM)

*Active Queue Management* merupakan sebuah metode yang memungkinkan perangkat jaringan untuk mengontrol panjang antrian atau lama waktu paket yang diperbolehkan untuk tetap berada di *buffer*

pada perangkat [12]. Penggunaan dari *Active Queue Management* adalah untuk menghindari terjadinya penumpukan data-data didalam *buffer* yang biasanya disebut *bufferbloat*. Terjadinya *bufferbloat* akan berdampak buruk pada layanan yang bersifat delay sensitive seperti VoIP karena meningkatnya *delay*, *jitter*, dan *packet loss* [8].

### 2.3. *Controlled Delay (CoDel)*

*Controlling Delay (CoDel)* merupakan skema baru dari *Active Queue Management (AQM)* yang diusulkan oleh Nichols dan Jacobson untuk mengatasi masalah *bufferbloat* yang terjadi di internet. CoDel dirancang untuk mengatasi *bufferbloat* pada *network links* seperti *router* dengan mengatur *limit* pada *delay network packet* yang terjadi karena melewati *buffer* [5].

CoDel bergantung pada *packet sojourn time* yaitu merupakan sebuah *actual queue delay* yang dirasakan pada *packet* sebagai patokan untuk memprediksi *congestion* pada jaringan. Jika nilai *packet sojourn time* lebih besar dari nilai target untuk interval waktu tertentu, CoDel akan memulai secara proaktif melakukan *drop* atau *marking* paket-paket untuk mengatur *queue length* [8].

CoDel dapat membedakan dua tipe jenis *queue*, yaitu *good queue* dan *bad queue*. *Good queue* sendiri didefinisikan sebagai antrian yang menunjukkan tidak ada *bufferbloat*. Hal ini dapat dicontohkan seperti LAN dengan kecepatan tinggi dan koneksi internet yang lambat, atau bisa juga seperti koneksi kabel dengan koneksi nirkabel. *Delay* pada *good queue* cenderung rendah karna tidak terjadi *congestion*. *Bad queue* sendiri didefinisikan sebagai antrian yang selalu terisi penuh bersamaan saat transmisi paket dilakukan, jadi *queue* atau antrian tidak pernah kosong yang dapat menyebabkan terjadinya *bufferbloat* [5].

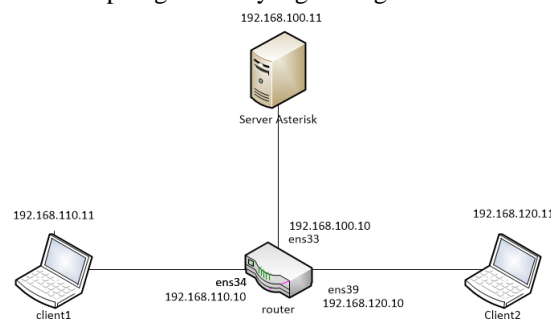
### 2.4. *DropTail*

*Droptail* merupakan algoritma dari *Passive Queue Management (PQM)* yang paling banyak diimplementasikan pada *router* internet saat ini. Algoritma ini identik dengan algoritma *First In First Out (FIFO)*. Algoritma ini menyebabkan *lock-out* dimana beberapa aliran data tidak mendapatkan porsi pembagian rata pada link *bandwidth* yang mengakibatkan aliran data yang lain tidak dapat masuk. Selain itu, ketika *queue* menjadi penuh *packet drop* dengan mudah dilakukan terhadap paket-paket yang datang menuju antrian. Tanpa mempedulikan seberapa penting paket yang dibuang dapat menyebabkan *bufferbloat* pada jaringan [13].

## 3. Sistem yang Dibangun

### 3.1. Topologi Jaringan

Penelitian ini membangun sebuah sistem yang dapat menilai kualitas layanan VoIP dengan menghitung nilai *Mean Opinion Score (MOS)*, *delay*, *throughput*, dan *packet loss*. Sistem ini dibangun secara virtualisasi menggunakan *Vmware Workstation* pada dua *host* yang berbeda. *Client1*, *router*, dan *server asterisk* di virtualisasi pada *host* pertama yaitu *personal computer (PC)*. *Client2* di virtualisasi pada *host* kedua yaitu pada *laptop*. Berikut topologi sistem yang dibangun :



Gambar 1- Topologi

*Bandwidth router* pada setiap *interface* yang terhubung ke *client1* dan *client2* akan disesuaikan agar dapat perbedaan yang jelas antara algoritma *Controlled Delay (CoDel)* dan *Droptail*. Algoritma *Controlled Delay* akan di implementasikan pada *router* yang bertindak sebagai *active queue management (AQM)*.

*DropTail* tidak perlu di implementasikan secara manual pada router karena sudah di anggap sama dengan first in first out sebagai scheduling.

### 3.2. Skenario Perancangan Uji

Skenario yang digunakan dalam penelitian yaitu *router* terhubung dengan *client* dan *server asterisk*. Pengujian dilakukan 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

Pada skenario ini dilakukan 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 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 CoDel pada sisi *router*.

#### b) Kondisi *bottleneck* 85kbps dan 85kbps

Pada skenario ini dilakukan 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 85kbps dan *interface* yang menghubungkan *router* dengan *client2* diatur dengan kecepatan 85kbps. Percobaan panggilan VoIP dilakukan dari *client1* ke *client2* dengan mengimplementasikan algoritma DropTail dan CoDel pada sisi *router*.

#### c) Kondisi *bottleneck* 80kbps dan 80kbps

Pada skenario ini dilakukan 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 CoDel pada sisi *router*.

#### d) Kondisi *bottleneck* 75kbps dan 75kbps

Pada skenario ini dilakukan 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 CoDel pada sisi *router*.

### 3.3. Parameter Uji

Terdapat tiga parameter uji yang digunakan dalam menganalisa mekanisme Controlling Delay (CoDel) dan DropTail

#### a. Delay

*Delay* merupakan waktu yang dibutuhkan oleh sebuah paket data dikirimkan dari pengirim sampai diterima oleh penerima. Kualitas delay dikatakan baik apabila waktu tunda yang terjadi hanya sekitar 0 – 150 ms [2][3]. Rata – rata delay dapat dihitung menggunakan persamaan (1).

$$\text{rata - rata delay} = \frac{\text{total delay}}{\text{total paket diterima}} \dots\dots(\text{persamaan 1})$$

#### b. Throughput

*Throughput* yaitu kecepatan (rate) transfer data efektif, yang diukur dalam bps (bit per second). *Throughput* adalah jumlah total kedatangan paket yang sukses diamati pada tujuan selama *interval* waktu tertentu dibagi oleh durasi *interval* waktu tersebut [3]. *Throughput* dapat dihitung menggunakan persamaan (4).

$$\text{throughput} = \frac{\text{paket data diterima}}{\text{lama pengamatan}} \dots\dots(\text{persamaan 2})$$

#### c. Packet Loss

*Packet Loss* merupakan jumlah paket yang hilang dalam suatu pengiriman paket data pada suatu jaringan. Beberapa penyebab terjadinya *packet loss* adalah adanya *noise*, *collision* dan *congestion*

yang disebabkan oleh terjadinya antrian yang berlebihan pada jaringan. *Packet Loss* pada VoIP dikatakan baik apabila jumlah tingkatan paket yang hilang berkisar antara 0 s/d 0.5% dari pengiriman data [2][19]. Packet loss dapat dihitung menggunakan persamaan (5).

$$packet\ loss = \frac{(paket\ data\ dikirim - paket\ data\ diterima) \times 100\%}{paket\ data\ yang\ dikirim} \dots\dots(\text{persamaan 3})$$

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 (6).

$$R = 94.2 - I_d - I_{ef} \dots\dots(\text{persamaan 4})$$

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

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

Tabel 1- Nilai Kualitas MOS

4. Evaluasi

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).

4.1 Performansi Droptail

Skenario	delay (ms)	throughput kbit/s	packet loss (%)		
			packet dikirim	paket diterima	packet loss
90 kbps	19,992935	169	15037	15037	0%
85 kbps	24,014133	153	10846	10049	7,35%
80 kbps	27,837536	129	9787	7797	20,33%
75 kbps	68,416345	101	5188	3941	24,04%

Tabel 2- performansi DropTail

Pada tabel 2 menjelaskan performansi DropTail pada setiap skenario *bottleneck* yang dilakukan yaitu 90 kbps, 85 kbps, 80 kbps, dan 75 kbps. Dari tabel juga dapat dijelaskan bahwa semakin kecil nilai skenario *bottleneck* maka besar nilai *delay* dan *packet loss* akan semakin besar, sedangkan nilai *throughput* akan semakin kecil.

#### 4.2 Performansi Controlled Delay (CoDel)

Skenario	delay (ms)	throughput (kbit/s)	packet loss (%)		
			packet dikirim	paket diterima	packet loss
90 kbps	10,001142	170	12651	12651	0%
85 kbps	11,519471	142	11110	10077	9,30%
80 kbps	29,721003	137	10082	7958	21,07%
75 kbps	55,979691	107	5997	4369	27,15%

Tabel 3- performansi CoDel

Pada tabel 3 menjelaskan performansi CoDel pada setiap skenario *bottleneck* yang dilakukan yaitu 90 kbps, 85 kbps, 80 kbps, dan 75 kbps. Dari tabel juga dapat dijelaskan bahwa semakin kecil nilai skenario *bottleneck* maka besar nilai *delay* dan *packet loss* akan semakin besar, sedangkan nilai *throughput* akan semakin kecil.

#### 4.3 Nilai Mean Opinion Score

No	DropTail				CoDel			
	90 kbps	85 kbps	80 kbps	75 kbps	90 kbps	85 kbps	80 kbps	75 kbps
1	4	4	4	3	4	4	3	3
2	4	3	4	4	3	3	4	4
3	4	3	3	2	5	4	3	1
4	4	3	3	3	5	4	3	2
5	4	3	3	2	5	4	3	3
6	5	3	3	3	5	4	3	3
7	3	3	2,5	2	4	3	2	3
8	4	3	2,5	2	4	3	2	2
9	4	3	3	2	5	3	3	2
10	5	3,5	2,5	1	5	4	3	2
11	5	4	3	2	4	4	3	2
12	5	4	3	2	4	3,5	3	1
13	4,5	4	3,5	2,5	4	3	3	2
14	4,5	3,5	3	2	4	3	2,5	1
15	4	3	3	1	3,5	3	2	1
16	4	4	2	1	4,5	4	2	1
17	3,5	3	2,5	2	4	3	2	1
18	5	4	3	2	4	3	2	2
19	4	3	3	2	4	3	2	2
20	4	3	3	2	4,5	3	3	2
21	4	3	2	2	5	3	3	2
22	4	3	3	2	4	3	2	2
23	3,5	3	2,5	1	4	3	2	1
24	4	3,5	3	2	4,5	3,5	3,5	2
25	3,5	2,5	2	1	3	2	2	1
26	5	4	3	2	4	3	2	2
27	4	3	3	2	5	3	3	2
28	5	4	3	2	4	3	3	3
29	4	3	2	2	3	3	2	2
30	4	3	2	2	3	3	2	1
Mean	4,18	3,30	2,84	2,02	4,17	3,27	2,60	1,93

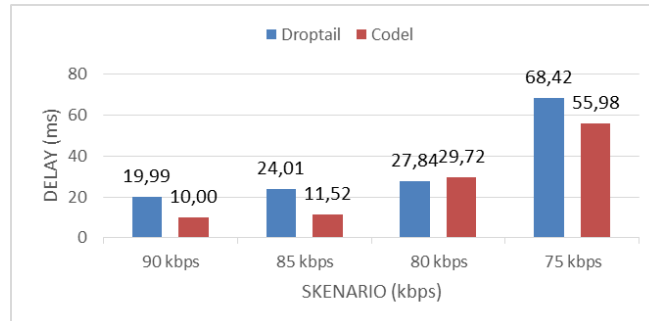
Tabel 5- Nilai Mean Opinion Score

Pada tabel 5 menjelaskan tentang kualitas suara, dinilai dengan *Mean Opinion Score* (MOS) secara subjektif. pada penelitian ini terdapat 30 orang koresponden yang terlibat dalam mendengarkan hasil suara rekaman semua skenario yang di ujikan, lalu koresponden dipersilahkan mengisi nilai 1-5 sesuai dengan kriteria nilai MOS yang di ditetapkan. Dapat dilihat nilai MOS pada percobaan ini berbeda cukup signifikan. Dimana perbedaan terjadi pada skenario yang semakin lama semakin kecil nilai *bandwidth*

yang diterapkan. Dapat dilihat pada skenario *bandwidth* 80 kbps dan 75 kbps bahwa nilai kualitas MOS rata-rata yang didapat lebih kecil sama dengan 3.

#### 4.4 Perbandingan Droptail dan CoDel

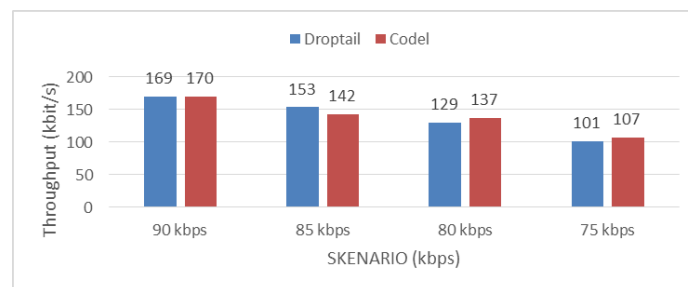
##### a. Delay



Gambar 2- grafik perbandingan delay pada Droptail dan CoDel

Gambar 2 menjelaskan pada skenario 90 kbps, 85 kbps, dan 75 kbps bahwa performansi CoDel dalam menjaga *delay* agar tetap rendah lebih baik dibandingkan dengan DropTail. Namun pada skenario 80 kbps delay yang terjadi pada Droptail dan CoDel tidaklah terlalu berbeda. Walaupun demikian algoritma CoDel tetap dianggap baik dalam menjaga delay agar tetap rendah dilihat dari 3 skenario lainnya.

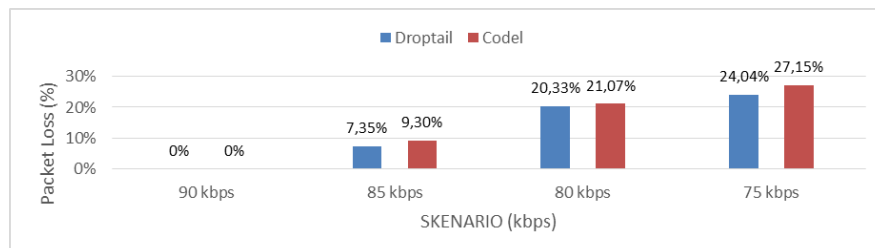
##### b. Throughput



Gambar 3- grafik perbandingan throughput pada Droptail dan CoDel

Gambar 3 menjelaskan bahwa performansi CoDel dalam menangani *throughput* tidak lah terlalu berbeda dengan DropTail. Hal ini disebabkan karena algoritma CoDel dalam menangani paket menggunakan *timestamp*, ketika terjadi antrian yang cenderung penuh akan semakin banyak paket yang di buang oleh CoDel sehingga menyebabkan paket yang diterima semakin sedikit. Dengan demikian CoDel belum bisa menangani *throughput* dengan baik walaupun jika dilihat dari 3 skenario yang lain CoDel lebih unggul daripada DropTail.

##### c. Packet loss



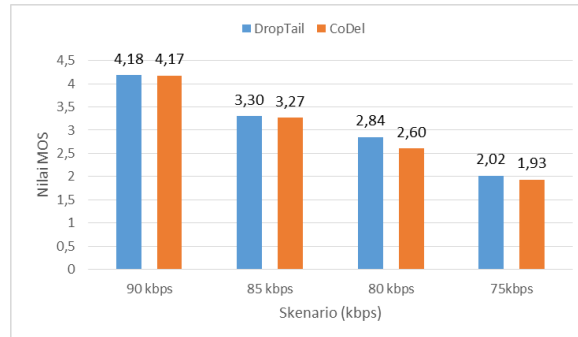
Gambar 4- grafik perbandingan packrt loss pada DropTail dan CoDel

Gambar 4 menyajikan perbandingan packet loss pada algoritma DropTail dan CoDel. Dapat dijelaskan bahwa DropTail lebih baik dalam mengatasi masalah packet loss walaupun sebenarnya persentase

perbedaannya tidaklah begitu signifikan. Hal ini disebabkan karena CoDel menggunakan mekanisme timestamp pada setiap paket yang masuk dalam antrian. Dengan demikian saat terjadinya bufferbloat pada antrian CoDel akan membuang paket yang melebihi waktu saat di antrian. Dengan begitu banyaknya paket yang hilang akan semakin tinggi.

#### d. Mean Opinion Score

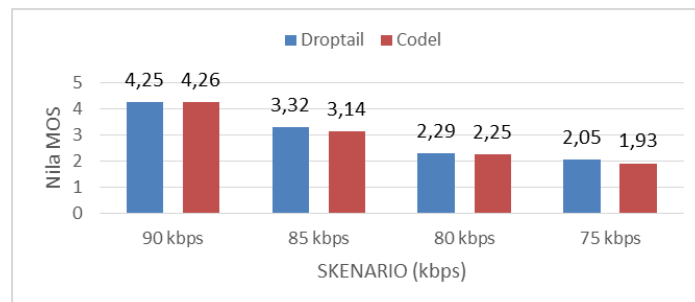
- **Subjektif**



Gambar 5- grafik perbandingan nilai MOS subjektif pada droptail dan CoDel

Gambar 5 merupakan grafik hasil pengujian Nilai MOS yang dilakukan oleh koresponden yang terlibat sebanyak 30 orang. Jika mengacu pada Tabel 1, nilai MOS yang didapatkan setelah menerapkan algoritma DropTail lebih baik daripada CoDel. Namun, pada skenario 80 kbps dan 75 kbps nilai MOS yang didapat belum bisa dikatakan cukup karena nilainya yang masih dibawah 3.

- **Objektif**



Gambar 6- grafik perbandingan nilai MOS objektif pada droptail dan CoDel

Gambar 6 merupakan grafik hasil pengujian Nilai MOS yang dihitung menggunakan nilai *R-factor* yang dapat dilihat pada tabel 4. Jika mengacu pada Tabel 1, nilai MOS yang didapatkan setelah menerapkan algoritma CoDel tidak lebih baik daripada DropTail. pada skenario 80 kbps dan 75 kbps nilai MOS yang didapat belum bisa dikatakan cukup karena nilainya yang masih dibawah 3

## 5. Kesimpulan

Tujuan dari penelitian Tugas Akhir ini adalah untuk menganalisis kualitas layanan VoIP berdasarkan *Mean Opinion Score* (MOS), *delay*, *throughput*, dan *packet loss* yang menerapkan algoritma *controlled delay* (CoDel) dan *droptail* pada *router*. Hasil perbandingan terhadap kedua algoritma tersebut disajikan dalam sub bab 4.4. Dari hasil tersebut dijelaskan bahwa performansi algoritma CoDel lebih baik jika dilihat dari nilai *delay* dan *throughput* yang didapat, sedangkan algoritma Droptail secara meyakinkan lebih baik dalam penanganan *packet loss*. Jika dilihat dari perbandingan nilai MOS, algoritma DropTail lebih baik dari algoritma CoDel secara subjektif ataupun secara objektif. Ini mengindikasikan implementasi algoritma CoDel terhadap layanan VoIP masih lebih buruk daripada DropTail. Fluktuatifnya grafik hasil perbandingan algoritma CoDel terhadap Droptail terjadi dikarenakan algoritma CoDel merupakan sebuah algoritma yang bersifat fleksibel dimana CoDel dapat dikonfigurasi sesuai kebutuhan aplikasi network yang digunakan. Agar mendapatkan nilai performansi terhadap algoritma CoDel lebih baik, parameter yang terlibat di dalam algoritma CoDel seperti



*target delay* dan *interval target* disesuaikan dengan pemakaian *bandwidth* pada aplikasi network yang diuji dan panjang antrian yang digunakan pada skenario.

## 6. Saran

Saran yang bisa diterapkan untuk penelitian ini lebih lanjut agar implementasi algoritma Active Queue Management (AQM) dilakukan pada jaringan 3G atau 4G yang memiliki bandwidth lebih besar dan diharapkan analisis yang dilakukan pada saat jaringan padat yang terisi oleh berbagai jenis paket data untuk menghasilkan *congestion*. Selanjutnya mungkin implementasi bisa dilakukan tidak hanya pada dua host yang di virtualisasi saja, namun diharapkan environment layanan VoIP yang dibangun layaknya nyata tanpa virtualisasi.

## Daftar Pustaka

- [1] Singh, R., & Chauhan, R. (2014, January). A Review Paper: Voice over Internet Protocol. *International Journal of Enhanced Research in Management & Computer Applications*, 3(1), 15-23.
- [2] Budi, E. S. (2012, Oktober). Analisa Quality of Service (QoS) Voice over Internet Protocol (VoIP) dengan Protokol H.323 dan Session Initial Protocol (SIP). *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 1
- [3] Ali, A., Ahmad, N., Akhtar, M. S., & Srivastava, A. (2013, January). Session Initial Protocol. *International Journal of Scientific and Engineering Research*, 4(1).
- [4] Sudiharto, D. W., Yulianto, F. A., & Arista, A. N. (2015). Comparative Analysis of Voice Over Internet Protocol (VoIP) Quality on Priority Queue (PQ) and Class-Based Queue (CBQ) Management System Using Link-Sharing Mechanism Setting. *International Conference on Information and Communication Technology (ICoICT)*.
- [5] Sharma, T. (2014, June). Controlling Queue Delay (CoDel) to counter the Bufferbloat Problem in Internet. *International Journal of Current Engineering and Technology*, 4.
- [6] Suliman, E. A., & Babiker A, D. (2015). UMTS VoIP Codec QoS Evaluation. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, 10(2).
- [7] Mohammed, S. E., & Nerma, M. H. (2015, August). Performance Analysis Of Active Queue Management (AQM) In VOIP Using Different Voice Encoder Scheme. *International Journal of Scientific & Technology Research*, 4(08).
- [8] Raghuvanshi, D. M., B., A., & Tahiliani, M. P. (2013). On the Effectiveness of CoDel for Active Queue Management. *Third International Conference on Advanced Computing & Communication Technologies*. doi:10.1109/ACCT.2013.27
- [9] White, G., & Padden, J. (2012, November). *PRELIMINARY STUDY OF CODEL AQM IN A DOCSIS NETWORK*. Cable Televisions Laboratories, Inc.
- [10] Martin, J., Hong, G., Westall, J. (2014, August). Managing Fairness and Application Performance with Active Queue Management in DOCSIS-based Cable Networks. Clemson, South Carolina.
- [11] Mandala, Satria. Noer Iskandar, Muhammad. Arief Nugroho, Muhammad. (2017) "Active Queue Management (AQM) Performance Analysis Based On Cotrolled Delay (CoDel) Algorithm On User Datagram Protocol (UDP)". Bandung
- [12] Fairhurst, F. B. (2015, July). IETF Recommendations Regarding Active Queue Management. Internet Engineering Task Force (IETF).
- [13] Adam, R. (2013). Active Queue Management: A Survey. *IEEE Communications Surveys & Tutorial*, 15.