

## ANALISIS PENINGKATAN QUALITY OF SERVICE ALIRAN TCP MENGGUNAKAN DYNAMIC RANDOM EARLY DETECTION (DRED)

Evalina Lia Yulianti<sup>1</sup>, Adiwijawa<sup>2</sup>, Setyorini<sup>3</sup>

<sup>1</sup>Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

---

### Abstrak

Active Queue Management (AQM) merupakan salah satu algoritma penanganan kongesti yang menggunakan sistem umpan balik yang melihat rata-rata antrian secara dinamis serta menentukan kapan dan paket yang mana yang akan ditandai. Referensi pada [1] menyatakan bahwa Random Early Detection (RED) merupakan salah satu metode AQM yang menandai paket secara random dengan probabilitas tertentu. Dari referensi tersebut dapat disimpulkan bahwa performansi RED bergantung pada level kongesti dan setting parameter-parameter yang dibutuhkannya.

Dynamic Random Early Detection (DRED) merupakan salah satu varian metode RED yang melakukan perubahan parameter RED secara dinamis. Parameter utama yang berubah secara dinamis adalah parameter  $wq$  (bobot antrian), yang sebelumnya diset secara konstan oleh RED. Secara teori, DRED memiliki performansi yang lebih baik dibandingkan dengan RED, karena DRED mampu menstabilkan nilai panjang antrian rata-rata setelah paket keluar dari antrian. Sehingga buffer overflow yang biasanya terjadi pada RED dapat dihindari.

Tugas akhir ini bertujuan membandingkan performansi RED dengan DRED yang meliputi panjang antrian rata-rata, packet loss, throughput, dan utilisasi buffer. Analisa performansi dilakukan pada simulasi jaringan sederhana dengan single bottleneck router dengan bantuan Network Simulator.

Kata Kunci : AQM, RED, DRED.

---

### Abstract

Active Queue Management (AQM) is one of congestion avoidance mechanism that uses loop back system to controls average queue size dynamically and decide when and which packet to drop. Based on reference [1], Random Early Management (RED) is one of AQM method that mark packets randomly with a certain probability. Also based on reference, can be conclude that RED's performance is depends on its congestion level and parameter settings.

The Dynamic Random Early Detection (DRED) is one of RED's variant which changes RED's parameter dynamically. The main parameter that changes dynamically is  $wq$  (queue weight) which sets as a constant value. Theoretically, DRED has a better performance than RED, because DRED capable to stabilize the average queue size after dequeuing packets. So, DRED can avoid buffer overflow.

The final project's aim is to compare RED and DRED's performance, that contains comparing average queue size, packet loss, and throughput. This performance analysis is simulated by Network Simulator in the simple network simulation with single bottleneck router.

Keywords : AQM, RED, DRED.

---

# 1. Pendahuluan

## 1.1 Latar Belakang

*Active Queue Management (AQM)* merupakan salah satu algoritma yang digunakan untuk menangani kongesti. AQM menggunakan sistem umpan balik yang memantau rata-rata panjang antrian secara dinamis serta menentukan kapan dan paket mana yang akan dibuang. Ketika terjadi kongesti, *gateway* akan memberitahukan kepada pengirim sehingga pengirim akan mengurangi laju pengiriman pakatnya. [3]

Salah satu algoritma AQM yang sering diimplementasikan dalam penelitian-penelitian yang berkaitan dengan penghindaran kongesti (*congestion avoidance*) adalah *Random Early Detection (RED)*. Dengan mekanisme RED, antrian akan membuang/menandai secara acak paket-paket yang datang jika ukuran antrian penuh (atau hampir penuh). RED digunakan untuk mempertahankan *throughput* agar tetap tinggi dan *delay* tetap rendah dengan cara mengontrol rata-rata ukuran antrian.

Estimasi rata-rata ukuran antrian dan perhitungan probabilitas *dropping* merupakan dua komponen utama dalam algoritma RED. Kesuksesan RED bergantung pada bagaimana perhitungan rata-rata ukuran antrian dan bagaimana menentukan probabilitas *dropping*. Perhitungan rata-rata ukuran antrian dirumuskan seperti pada persamaan 1.1 :

$$avg(t) \leftarrow (1 - wq) \cdot avg(t - 1) + wq \cdot q \tag{1.1}$$

di mana :

- avg(t) : nilai panjang rata-rata antrian pada waktu ke-t
- avg(t-1) : nilai panjang rata-rata antrian pada waktu ke-(t-1)
- wq : bobot antrian
- q : panjang antrian pada waktu ke-t

Sementara itu, perhitungan probabilitas *dropping* dirumuskan seperti persamaan 1.2 di bawah ini :

$$pb \leftarrow \max p \cdot \frac{(avg - \min th)}{(\max th - \min th)} \tag{1.2}$$

di mana :

- $p_b$  : probabilitas *dropping packet*
- $\max_p$  : nilai maksimum dari  $p_b$
- avg : nilai rata-rata antrian
- $\min_{th}$  : nilai *threshold* minimum
- $\max_{th}$  : nilai *threshold* maksimum

Pada *original RED*, *wq* yang digunakan untuk mengontrol sensitivitas *gateway* terhadap terjadinya kongesti diset pada nilai yang konstan. Sehingga, *original RED* tidak dapat bereaksi dengan cepat apabila terjadi lonjakan trafik yang tinggi, dan akibatnya *buffer overflow* tidak dapat dihindari.

Untuk mengatasi hal tersebut, maka dikembangkan suatu pendekatan yang disebut *Dynamic Random Early Detection* (DRED), di mana nilai  $wq$  dapat disesuaikan dengan kondisi jaringan. Dalam pemodelan DRED, terdapat metode baru untuk memperbaiki kinerja algoritma RED. Yaitu dengan cara mengatur nilai  $wq$  secara dinamis dengan perubahan surplus dari ukuran antrian terhadap rata-rata ukuran antrian. Lonjakan yang masih dapat ditampung oleh *gateway* ditentukan oleh ukuran buffer. Jika ukuran buffernya besar, maka lonjakan trafik yang lebih besar pun masih dapat ditampung. Nilai surplus digunakan untuk menentukan nilai rasio  $R$  (persamaan (1.3)) yang berfungsi untuk menentukan apakah ada lonjakan dan keagresifan pada trafik yang datang. Berdasarkan variasi nilai rasio tersebut, maka nilai  $wq$  dapat ditentukan secara dinamis.

$$R = \frac{\text{surplus}}{u\_buf} \quad (1.3)$$

di mana :

$R$  : rasio / perbandingan antara surplus terhadap ukuran buffer

surplus : nilai perbandingan ukuran antrian terhadap rata-ratanya

$u\_buf$  : ukuran buffer

Dengan DRED, diharapkan *gateway* dapat mendeteksi terjadinya kongesti lebih awal daripada yang dilakukan oleh *original* RED, sehingga probabilitas *dropping* paket pun semakin berkurang.

Pada tugas akhir ini akan dibahas mengenai algoritma *Dynamic* RED dan dilakukan analisis perbandingan *Quality of Service* antara algoritma RED dan *Dynamic* RED.

## 1.2 Perumusan Masalah

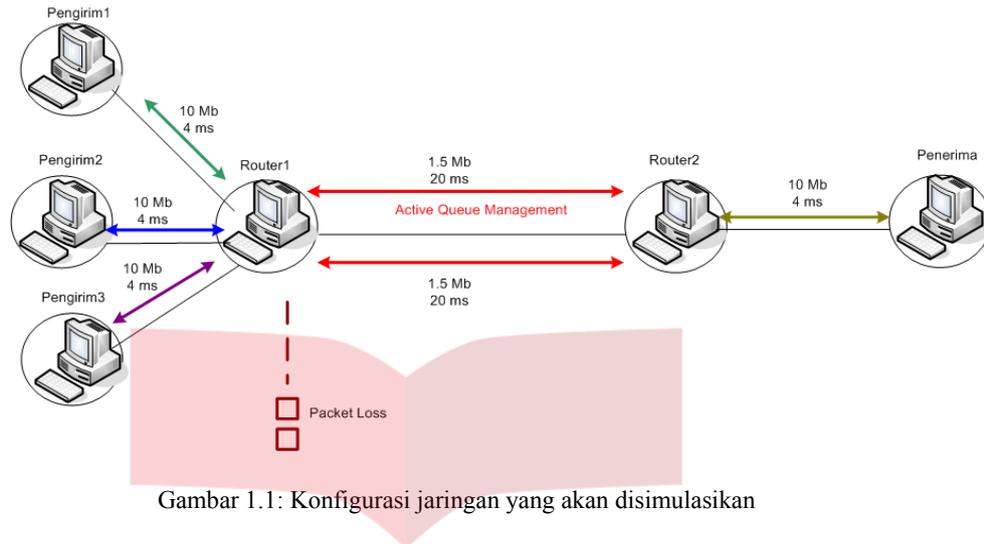
Permasalahan yang dijadikan objek penelitian tugas akhir ini menitikberatkan pada :

- 1) Bagaimana mengimplementasikan algoritma *Dynamic* RED dan RED pada simulasi aliran TCP.
- 2) Bagaimana perbandingan performansi antara algoritma *Dynamic* RED dan RED pada simulasi aliran TCP.

## 1.3 Pembatasan Masalah

Pembahasan dalam tugas akhir ini adalah dalam batas-batas sebagai berikut :

- 1) Simulasi dilakukan pada jaringan sederhana dengan konfigurasi topologi seperti gambar 1.1 untuk mempermudah pemahaman mengenai *congestion avoidance* dengan mekanisme RED dan DRED.



Gambar 1.1: Konfigurasi jaringan yang akan disimulasikan

- 2) Parameter hasil simulasi yang dianalisis adalah : *average queue size*, *throughput*, *packet loss*.
- 3) Analisis dikhususkan pada algoritma pendekatan *Dynamic RED* yang merupakan pengembangan dari RED.
- 4) Model yang digunakan hanya untuk trafik TCP Vegas saja, tidak membahas agen TCP yang lain maupun UDP.
- 5) Aplikasi pembangkit trafik yang digunakan adalah FTP, karena FTP merupakan aplikasi yang digunakan untuk mengirimkan data.
- 6) Simulator yang digunakan adalah NS2 (Network Simulator 2).
- 7) Aliran TCP difokuskan pada *single bottleneck link* untuk aplikasi *wired*.
- 8) Tidak membahas *interface* dengan layer di atas atau di bawah lapisan transport.
- 9) Tidak membahas implementasi algoritma RED dan DRED pada aliran TCP pada dunia nyata.
- 10) Tidak membahas algoritma *slow start*, *fast retransmit*, maupun *fast recovery*.

## 2.

### 1.4 Tujuan Penelitian

Tujuan penelitian terhadap masalah ini adalah sebagai berikut :

- 1) Membangun dan mengimplementasikan *Dynamic RED* untuk skema AQM pada aliran TCP.
- 2) Menganalisis parameter *Dynamic RED* yang mempengaruhi performansi AQM pada aliran TCP.
- 3) Menganalisis performansi hasil implementasi algoritma *Dynamic RED* dan membandingkannya dengan algoritma RED pada aliran TCP.

3.

### 1.5 Metodologi Penyelesaian Masalah

Masalah yang telah penulis rumuskan di atas akan dipecahkan dengan metode studi literatur terhadap bahasan-bahasan yang berkaitan dengan masalah utama, yakni:

1. Studi Literatur mengenai TCP, *Quality of Service* (QoS), algoritma RED dan *Dynamic RED*, dan simulator NS2.
2. Melakukan analisa atau identifikasi kebutuhan dari sistem yang akan dibuat.
3. Merancang desain model simulator dengan menggunakan aplikasi simulator NS2.
4. Menjalankan simulasi dan mengumpulkan output dari simulasi.
5. Melakukan analisis terhadap hasil simulasi.

4.

### 1.6 Sistematika Penulisan

Tugas Akhir ini disusun dengan sistematika pembahasan sebagai berikut :

#### **BAB I PENDAHULUAN**

Bab ini memaparkan latar belakang masalah, perumusan masalah yang akan dibahas, pembatasan masalah, tujuan yang ingin dicapai melalui penelitian ini, metode penyelesaian masalah dan sistematika pembahasan.

#### **BAB II LANDASAN TEORI**

Bab ini berisi uraian mengenai landasan teori yang akan digunakan dalam penyelesaian tugas akhir.

#### **BAB III ANALISIS DAN PERANCANGAN**

Bab ini menjelaskan mengenai analisa sistem yang akan dibuat dan perancangan sistem.

#### **BAB IV ANALISIS PERFORMANSI SISTEM**

Bab ini berisi tentang simulasi dan uji kerja sistem yang telah dirancang.

#### **BAB V PENUTUP**

Berisi kesimpulan dari hasil penelitian tugas akhir ini serta saran-saran untuk pengembangan lebih lanjut.

Telkom  
University

## 5. Penutup

### 5.1 Kesimpulan

Dari analisis terhadap hasil simulasi yang telah dilakukan dalam tugas akhir ini, didapatkan kesimpulan sebagai berikut :

1. Berdasarkan nilai surplus panjang antrian terhadap rata-rata antriannya dan rasio surplus terhadap ukuran buffernya, nilai parameter  $wq$  dapat berubah secara dinamis, sehingga nilai rata-rata antrian selanjutnya dapat meningkat atau menurun sesuai dengan nilai  $wq$ -nya. Setelah terjadi lonjakan perubahan nilai antrian yang tinggi, DRED mampu menurunkan nilai rata-rata panjang antrian mendekati nilai minimum threshold yang telah ditetapkan.
2. Nilai rata-rata panjang antrian selalu meningkat seiring dengan peningkatan jumlah trafik. Setelah ditambahkan dengan trafik baru, nilai rata-rata panjang antrian pada DRED memang sempat meningkat, akan tetapi dapat kembali menurun setelah periode waktu tertentu, tergantung pada jumlah trafik yang dibangkitkan.
3. *Throughput* dan *packet loss rate* selalu berbanding terbalik. Jika *packet loss rate* besar, maka *throughput*nya kecil, demikian sebaliknya. Berdasarkan hasil simulasi, meskipun nilai perbedaannya tidak terlalu besar, *packet loss rate* pada DRED selalu lebih kecil daripada RED dan *throughput* pada DRED lebih besar daripada RED.
4. Secara keseluruhan, performansi DRED lebih baik daripada RED.

### 5.2 Saran

Beberapa saran yang dapat diberikan untuk tujuan pengembangan penelitian selanjutnya antara lain :

1. Simulasi yang dilakukan masih sebatas simulasi jaringan sederhana pada TCP Vegas dengan aplikasi FTP. Untuk pengembangan selanjutnya, dapat dicobakan simulasi untuk jenis TCP yang berbeda dan dengan jaringan yang lebih kompleks
2. nilai parameter inputan yang digunakan pada simulasi sebaiknya mendekati kondisi sebenarnya.

Telkom  
University

## Referensi

- [1] S. Floyd, V. Jacobson, "Random Early Detection *gateways* for Congestion Avoidance", IEEE/ACM Transaction on Networking, August, 1993
- [2] A.A. Akintola, et al., "Modeling and Performance Analysis of Dynamic Random Early Detection (DRED) *Gateway* for Congestion Avoidance"
- [3] Braden et. al., RFC 2309, "Recommendations on *Queue* Management and Congestion Avoidance in the Internet", April, 1998
- [4] W. R. Stevens, RFC 2001, "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery", January, 1997
- [5] Ipuk Sri Hadiyati, Adiwijaya, Hafidudin, "Analisis Performansi Adaptive RED sebagai AQM pada Aliran TCP", STT Telkom, Bandung, 2006
- [6] B. Zheng, M. Atiquzzaman, "Study of Active *Queue* Management Using OPNET", USA.
- [7] E. Brent Kelly, "Quality of Service In Internet Protocol (IP) Networks", Brookline, 2002.
- [8] Tanenbaum, Andrew S, "Jaringan Komputer Edisi Bahasa Indonesia Jilid 2", Prenhallindo Jakarta, 1997

[9] [en.wikipedia.org/wiki/Congestion Avoidance Overview](http://en.wikipedia.org/wiki/Congestion_Avoidance_Overview)

[10] <http://www.isi.edu/nsnam/ns/>

[11]

[12]

[13]

[14]

[15]

[16]

[17]

[18]

[19]

[20]

[21]

[22]

[23]

[24]

[25]

[26]

[27]

