

## **Analisa Kendali Trafik Data Internet Melalui Jaringan ATM Dengan Menggunakan Kendali Aliran TCP dan Layanan Informasi ABR**

**Rendy Munadi, Dadang Gunawan, dan Sar Sardy**

Fakultas Teknik, Universitas Indonesia, Kampus Baru UI – Depok.16424

Telp : 6221-78888430 Fax : 6221-78887861

Email: rendymunadi@bdg.centrin.net.id, guna@yexa.eng.ui.ac.id, spsopto@bit.net.id

### **Abstrak**

Pertumbuhan trafik pengguna internet dari waktu ke waktu mengalami perkembangan yang menakjubkan, oleh karena itu penelitian untuk meningkatkan unjuk kerja jaringan terus dilakukan guna memberikan solusi terhadap penanganan lonjakan trafik yang terjadi dan tuntutan akan kualitas pelayanan oleh end user. Jaringan yang telah direkomendasikan sebagai jaringan transport kecepatan tinggi adalah jaringan Asynchronous Transfer Mode (ATM) yang mampu mentransfer data dalam bentuk paket yang berukuran kecil yang disebut sel. Jika trafik data internet dipandang sebagai trafik Transmission Control Protocol / Internet Protocol (TCP/IP), maka penggunaan TCP sebagai protokol transport yang andal adalah sangat tepat.

Dalam tulisan ini dipaparkan analisa kendali trafik data internet di Jaringan ATM dengan menggunakan kendali aliran TCP dan layanan informasi Available Bit Rate (ABR). Tinjauan parameter yang dilakukan meliputi : kecepatan pengiriman data oleh sumber, besarnya throughput dan efisiensi jaringan, panjang antrian yang terjadi dan tingkat fairness yang diberikan terhadap jumlah sumber TCP yang bervariasi, ukuran paket yang berbeda dan perubahan kapasitas buffer switch.

Dari hasil penelitian yang dilakukan melalui simulasi dapat ditunjukkan bahwa kecepatan pengiriman data oleh sumber sesuai dengan kapasitas saluran dan banyaknya sumber yang aktif, sedangkan throughput, efisiensi dan tingkat fairness akan maksimum untuk ukuran paket TCP = 1024 byte. Panjang antrian paket berukuran 512 byte yang melebihi kapasitas buffer akan menurunkan besarnya throughput dan efisiensi masing-masing 3% dan 4%.

**Kata kunci:** Trafik internet, TCP/IP, Jaringan ATM, layanan ABR, Buffer Switch

### **Abstract**

The growth of internet traffic user from recent years experiences fast growing, therefore the research to improve of network performance will be continue for give solution to handling of enormous traffic that happened and demand of service quality by end user. The network has been recommended as transport of network high speed is Asynchronous Transfer Mode (ATM) Network. The ATM network capable to transfer data in the form of packet format with small size and fixed length is called as cell. If the data traffic of internet is viewed as TCP/IP traffic, hence usage of TCP as transport protocol is very appropriate.

In this paper proposes the control analysis data traffic of internet over ATM network by using control TCP flow and information service of Available Bit Rate (ABR). The parameter metrics such as rate of data send by source, throughput and efficiency of the network, length of queue in buffer of switch and fairness levels. These parameters are depends of variation of source number, different of packet size and changes of buffer capacity.

From simulation results can be indicated that rate of data send by source as according to channel capacities and number of active sources, while high fairness for all sources and high throughput can be achieved when packet size of TCP 1024 bytes. For the packet size of TCP 512 bytes, length of queue exceeding capacities of buffer, so degradation of throughput and efficiency around 3% and 4%.

**Keywords:** Internet traffic, TCP/IP, ATM network, ABR service, Buffer Switch

## 1. Pendahuluan

Dengan berkembangnya penggunaan Internet yang sangat menakjubkan, maka layanan yang sesuai untuk Jaringan ATM adalah layanan *Available Bit Rate* (ABR) dan *Guaranteed Frame Rate* (GFR), dengan memandang bahwa trafik dari pengguna Internet sebagai trafik data TCP/IP [1,2]. Pengguna dari layanan tersebut tentu ingin mendapatkan kualitas pelayanan yang memadai, oleh karena itu *Network Provider* harus dapat memenuhi akan keinginan tersebut.

Layanan ABR dan GFR di Jaringan ATM terkonsentrasi pada *switch* ATMnya sendiri, khususnya di *buffer*. Oleh karena itu pemecahan masalahnya merupakan bagian dari manajemen *buffer*, sehingga dengan layanan-layanan tersebut diharapkan dapat meningkatkan kinerja jaringan yang meliputi : *throughput*, dan efisiensi. Hal ini sesuai dengan karakteristik layanan-layanan tersebut, yaitu bahwa trafik data TCP/IP akan melalui proses segmentasi dan enkapsulasi, baik disisi sumber pengirim maupun disisi penerima, sehingga dapat memenuhi kualitas ataupun jaminan pelayanan yang dikehendaki pengguna [1,4].

Teknik yang digunakan dalam layanan informasi ABR, yaitu dengan memanfaatkan informasi umpan-balik yang ada pada sel RM (*Resource Management cell*) dan penggunaan algoritma *switch* ERICA (*Explicit Rate Indication for Congestion Avoidance*) terhadap trafik data TCP/IP yang ter-*aggregate* dalam *Virtual Channel* (VC) di jaringan ATM [3,4].

Layanan informasi ABR dirancang untuk menangani trafik data di jaringan ATM, yang memberikan jaminan pelayanan (kehilangan sel minimum) ke pemakai yang menghendaki kebutuhan *throughput* yang bervariasi. Prinsipnya dalam layanan ini adalah memanfaatkan sisa *bandwidth* di jaringan setelah layanan lain, dengan jaminan *bandwidth* dipenuhi. Jaringan secara aktif memberikan informasi ke

pengirim dan memonitor kondisi yang terjadi selama hubungan berlangsung.

Penelitian yang telah dilakukan adanya pembatasan jumlah sumber TCP yang aktif [1,2]. Permasalahan yang akan timbul, adanya peningkatan terhadap jumlah sumber TCP yang aktif sehingga pengaturan kecepatan pengiriman oleh sumber TCP sulit dilakukan, disamping itu adanya kecenderungan *switch* untuk melakukan pembuangan sel jika kondisi kongesti terdeteksi.

Pertanyaannya adalah mampukah *switch* mempertahankan kinerjanya.

Untuk mengatasi permasalahan diatas perlu dilakukan penelitian melalui simulasi terhadap penambahan jumlah sumber TCP yang aktif dalam layanan informasi ABR dengan cara bertahap, disamping membuat batasan ukuran *buffer*, yaitu dengan menentukan maksimum *buffer size* dan *threshold*-nya, sehingga tidak terjadi pembuangan sel oleh *buffer* atau kondisi kongesti dapat dihindari, dengan kata lain kinerja dari *switch* tetap dipertahankan. Sebagai parameter yang terpenting untuk jenis layanan ini adalah besarnya *throughput* yang mampu dicapai oleh keadaan beban trafik yang selalu berubah [3].

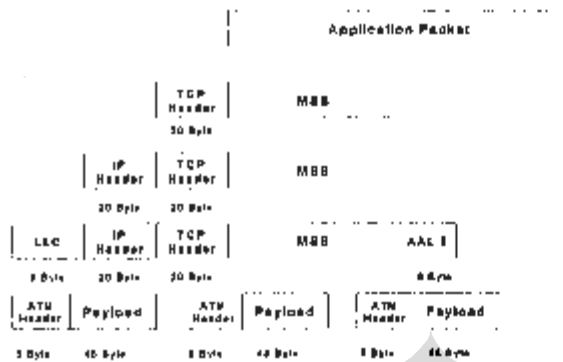
## 2. Mekanisme Pengendalian

### 2.1. Integrasi TCP/IP dan Jaringan ATM

*Request For Common* (RFC) dari Internet no. 2225 tertuang spesifikasi *interoperability* terhadap *transport* TCP/IP di Jaringan ATM. Gambar.1 menjelaskan struktur dari *Protocol Data Unit* (PDU) pada tiap lapisnya. Paket-paket lapis aplikasi dipecah menjadi *segment-segment* TCP. Ukuran *Maximum Segment Size* (MSS) didasarkan pada *Maximum Transmission Unit* (MTU) dari lapis *link*, satu segmen TCP tidak kurang dari panjang *byte* MSS.

Lapis TCP menambahkan 20 *byte* sebagai *header* dari segmennya dan melewatkannya di lapis IP dengan tambahan 20 *byte* lagi pada *header*-nya. Paket-paket IP ini kemudian diproses oleh lapis

interworking IP-ATM. Oleh lapis *Logical Link Layer* (LLC) ditambahkan pula *header* 8 byte yang berisi protokol ID guna ke lapis yang lebih tinggi.



**Gambar 1.**  
TCP/IP di Jaringan ATM: *Protocol Data Unit* [2].

Pada lapis protokol, *header* inilah yang membedakan paket-paket IP dengan paket lainnya. Selanjutnya paket tersebut dilewatkan ke lapis *ATM Adaptation Layer-5* (AAL-5) dan mendapatkan 8 byte pada *trailer*-nya guna membentuk *frame* AAL-5. Di lapis AAL-5 inilah *frame* akan dipecah menjadi sel-sel ATM yang berisi 48 byte sebagai *payload* dan 5 byte sebagai *header* [5].

**2.2. Mekanisme Kendali Aliran TCP**

Dalam hubungan TCP digunakan *window* pengendali aliran dari ujung ke ujung guna membatasi jumlah paket yang dikirim oleh sumber. *Window* pengirim merupakan *window* penerima minimum (*Wrcvr*) dan *window variable* kongesti (*CWND*). Ketika kehilangan paket terjadi di hubungan TCP, sumber tidak menerima *Acknowledge* (ACK) dan batas *timeout*-nya. Sumber mengetahui nilai *window* kongestinya, sehingga untuk mencegah kehilangan paket maka perlu di-*setting* nilai *variable threshold* yaitu setengah dari nilai *window*-nya. Sumber kemudian mengirim kembali paket yang hilang dan menambah *CWND* dengan 1 setiap waktu paket di *Acknowledge*. Fase ini dikenal dengan "fase penambahan secara eksponensial". Proses ini berlangsung hingga *window* sama dengan nilai *Slow Start Threshold*-nya (*SSTHRESH*). Setelah itu nilai *window w*

akan bertambah sebesar 1/w untuk setiap paket yang di-ACK dan fase ini disebut "fase penambahan linier" [5].

Dalam penelitian ini digunakan mekanisme *fast retransmission and recovery* (FRR), ketika TCP tujuan menerima segmen yang bukan sebenarnya (*out of order segment*), maka TCP tujuan akan segera mengirim *Acknowledge* duplikat ke pengirim. Ketika pengirim menerima 3 paket duplikat ACKs, maka pengirim mengerti bahwa segmen dengan indikasi ACKs sudah hilang dan dengan segera pengiriman ulang segmen yang hilang dilakukan. Ini berarti bahwa kongesti ditandai dengan penerimaan 3 paket duplikat ACKs tadi. Pengirim kemudian menurunkan *CWND*nya menjadi setengahnya (plus 3 segmen) dan mengamankan setengah dari nilai *CWND* semula dalam *SSTHRESH* (perkiraan dari nilai mantap yang baru) [1,5].

**2.3. Mekanisme Kendali Layanan Informasi ABR.**

Mekanisme dasar dari pengendalian kongesti di layanan ABR, yaitu dengan kebutuhan akan kerjasama antar *Source End System* (SES) dan *Destination End System* (DES) dalam hal kendali kecepatan ke jaringan yaitu nilai *Allowed Cell Rate* (ACR) yang harus diberikan ke sumber. SES menghasilkan sel RM dalam setiap  $N_{rm-1}(=31)$  sel data terkirim, hal ini untuk mencegah jaringan dari kondisi kongesti.

**Tabel 1.**  
Sifat sumber berdasar bit CI dan NI [7]

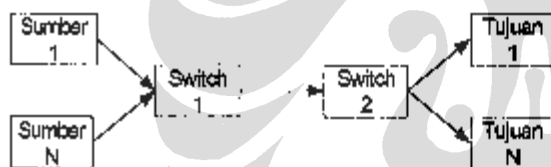
NI	CI	Nilai
0	0	$ACR \leftarrow \min(ER, ACR + RIF \times PCR, PCR)$
0	1	$ACR \leftarrow \min(ER, ACR - ACR \times RDF)$
1	0	$ACR \leftarrow \min(ER, ACR)$
1	1	$ACR \leftarrow \min(ER, ACR - ACR \times RDF)$

Sel RM mengandung *Current Cell Rate* (CCR), *Peak Cell Rate* (PCR), *Explicit Rate* (ER), dan parameter lainnya seperti bit *Congestion Indication* (CI) dan bit *No Increase* (NI). Aturan dasar untuk nilai ACR dapat dilihat di Tabel 1. Ketika SES menerima sel RM arah balik, maka SES

akan menyesuaikan kecepatannya guna mengangkut informasi di dalam sel RM. Jika bit  $CI=0$ , maka SES dapat secara linier menambah kecepatannya dengan faktor *rate*. Ketika SES menerima sel RM arah balik, maka SES akan menyesuaikan kecepatannya guna mengangkut informasi *Rate Increase Factor* (RIF) hingga mencapai nilai ER di sel RM tapi tidak melebihi nilai kecepatan puncak (PCR). Jika bit  $CI=1$ , maka SES harus menurunkan secara eksponensial kecepatannya dengan menggunakan nilai *Rate Decrease Factor* (RDF). Nilai RIF dan RDF ditentukan pada saat pembangunan hubungan akan dilakukan. Jika kecepatan masih lebih besar dari ER, maka SES harus terus menurunkan kecepatannya hingga menuju ER tersebut, tetapi tidak lebih rendah dari MCR [5,6,7].

### 3. Pemodelan dan Simulasi

Konfigurasi sistem yang akan dianalisa adalah seperti pada Gambar.2. Konfigurasi terdiri dari N buah sumber TCP yang identik, dua buah *switch* ATM.



Gambar 2.  
Pemodelan simulasi

#### 3.1. Sumber TCP

Sumber TCP membangkitkan trafik secara tetap, dimana sumber selalu memiliki data yang akan dikirimkan apabila diijinkan oleh *window*-nya. TCP MSS diset 512 *byte* dan 1024 *byte* (ekivalen dengan 12 dan 23 sel ATM). Maksimum *window* di penerima diset 64 Kbyte. Kecepatan *link* 155.52 Mbps. Waktu *delay* ACK = 200 ms. Waktu proses setiap pengiriman dan penerimaan paket = 0.3 ms. Minimum RTO = 500 ms. Parameter layanan informasi ABR yang diset antara lain: PCR=155.52 Mbps, MCR=0 Mbps

#### 3.2. Switch ATM

*Switch* menerapkan mekanisme ERICA dengan target utilitas 0.9 dan *scheduling* yang digunakan adalah FIFO. Pada analisa ini diasumsikan setiap sumber TCP terhubung pada sebuah VCI. Seluruh *link* berjalan pada 155.52 Mbps dan *delay* total propagasi 278 mikrodetik, dengan pembagian sebagai berikut ; 122.68 mikrodetik *delay* akses tiap sumber TCP ke *switch* dan *switch* ke TCP penerima, dan 32.71 mikrodetik *delay* pada *bottleneck link*.

### 4. Hasil Simulasi dan Analisa

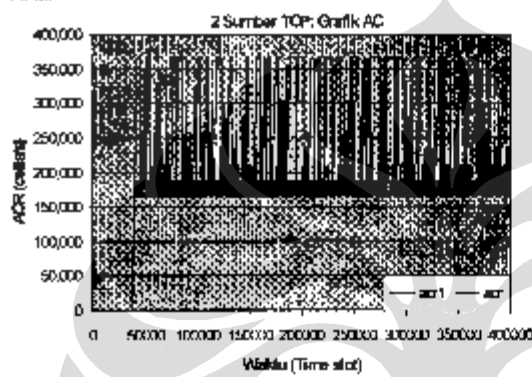
Analisa dilakukan untuk melihat parameter kinerja yang meliputi karakteristik-karakteristik dari kecepatan pengiriman data oleh sumber, *throughput* dan panjang antrian. Disamping itu melihat nilai dari total *throughput*, efisiensi dan *indeks fairness* yang mampu dicapai melalui simulasi dengan memasukkan variabel jumlah sumber TCP, besar ukuran paket dan ukuran *buffer* di *switch*.

#### 4.1. Karakteristik Kecepatan Sumber.

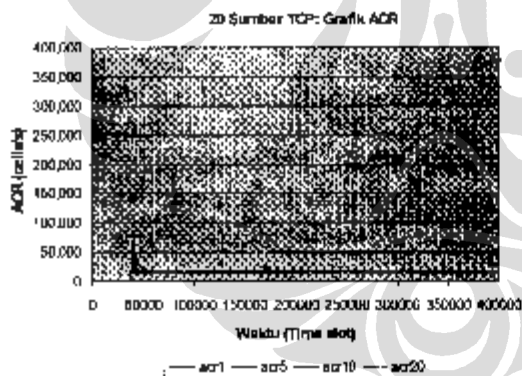
Gambar.3 merupakan hasil simulasi dari karakteristik kecepatan pengiriman data oleh sumber yang dikenal sebagai *Allowed Cell Rate* (ACR). Kecepatan sumber dengan ukuran paket TCP 512 *byte* dan digunakan ukuran *buffer switch* maksimum dan *threshold*-nya (10000,8000) sel, diperoleh :

- Dengan jumlah sumber 2 TCP, diperlihatkan karakteristik kecepatan sumber yang berbeda (*acr1* dan *acr2*), dengan fluktuasi yang sangat besar sepanjang waktu simulasi (seperti pada Gambar.3.a). Besarnya nilai ACR disekitar harga 165.000 sel/det atau sebesar 70 Mbps untuk masing-masing sumber, yang berarti sesuai dengan nilai *fairshare*-nya.
- Untuk 20 sumber TCP, dengan mengambil *sampling* terhadap karakteristik ACR pada sumber 1, 5, 10 dan 20 yaitu : *acr1*, *acr5*, *acr10* dan *acr20* (seperti pada Gambar.3.b), maka nilai ACR yang dicapai sebesar 16.500 sel/det atau sebesar 7.0 Mbps.

Adanya variasi kecepatan sumber yang dihasilkan dipengaruhi oleh faktor RIF dan RDF, dimana ketika sumber menerima informasi bahwa jaringan dalam keadaan bebas kongesti (dengan disetnya bit NI=0 dan CI=0) maka sumber akan menaikkan nilai ACR-nya sebesar PCR x RIF. Demikian sebaliknya, jika sumber menerima informasi bahwa keadaan jaringan kongesti (dengan disetnya bit CI=1), maka sumber akan menurunkan nilai ACR-nya sebesar ACR x RDF.



(a)



(b)

Gambar 3.

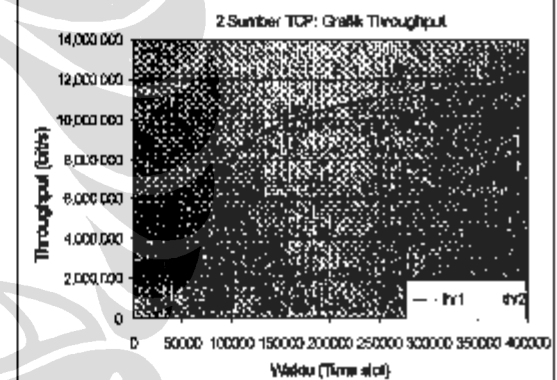
Karakteristik ACR untuk 2 sumber TCP (a) dan 20 sumber TCP (b)

4.2. Karakteristik Throughput.

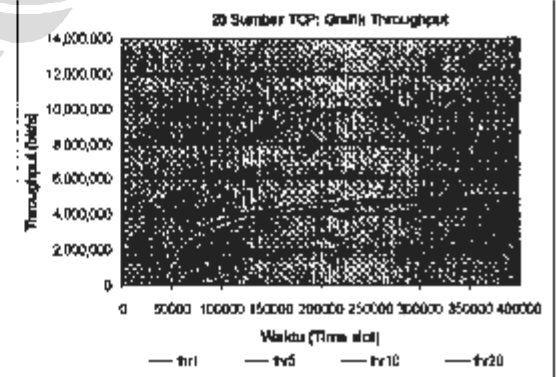
Gambar.4 di bawah menunjukkan karakteristik *throughput* untuk sumber yang berbeda, yaitu dengan jumlah sumber 2 dan 20. Dengan jumlah sumber 2 TCP, diperlihatkan karakteristik *throughput* yang sama ( $thr1 = thr2$ ) sepanjang waktu simulasi, dengan nilai *throughput* maksimum 12Mbps (seperti pada Gambar.4a). Sedangkan pada jumlah sumber

20 TCP, dengan mengambil *sampling* terhadap karakteristik *throughput* pada sumber 1, 5, 10 dan 20 yaitu :  $thr1$ ,  $thr5$ ,  $thr10$  dan  $thr20$  tampak berbeda (seperti Gambar.4b), adapun *throughput* maksimum yang dapat dicapai sepanjang waktu simulasi adalah 5 Mbps.

Jadi nampak perbedaan yang sangat drastis jika kita melihat dari jumlah *throughput* yang dicapai oleh 2 sumber TCP adalah sekitar 24 Mbps, sedangkan untuk jumlah sumber 20 TCP dengan asumsi *throughput* rata-rata tiap sumber 5 Mbps, maka diperoleh jumlah *throughput* untuk 20 sumber TCP adalah 100 Mbps. Dengan demikian jika kita menyediakan *link* yang besar, maka sebaiknya *link* tersebut diperuntukkan untuk jumlah *user* yang banyak, sehingga *occupancy link* tersebut akan tinggi



(a)



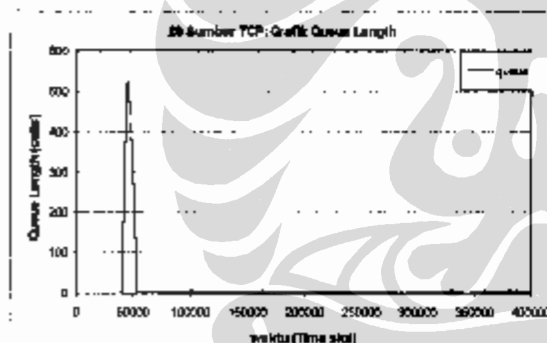
(b)

Gambar 4.

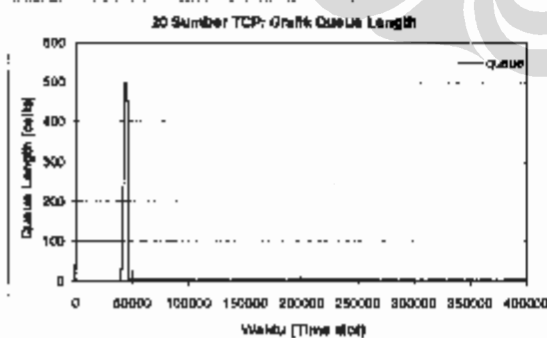
Karakteristik *throughput* untuk 2 sumber TCP (a) dan 20 sumber TCP (b)

### 4.3. Karakteristik Panjang Antrian

Gambar.5a menunjukkan karakteristik panjang antrian yang terjadi di *buffer* dengan menggunakan 20 sumber TCP,  $MSS=512$  byte dan ukuran *buffer* maksimum, *threshold* (500,500) sel, terlihat adanya sel yang di buang akibat panjang antrian yang melebihi ukuran *buffer* yang diijinkan, hal ini diperlihatkan dengan terpotongnya grafik panjang antrian (*queue length*) pada skala 500 sel. Terjadinya lonjakan antrian pada *switch* ( *time slot* ke-50000 atau detik ke 0.14 ) disebabkan pada selang waktu tersebut terjadi pengiriman data yang cepat dari sumber TCP (sebagai *fasa slow start*). Pada fase tersebut sumber mengirimkan pakatnya dua kali dari pada paket sebelumnya saat menerima ACK. Setelah mencapai nilai *SSTHERSH*, TCP akan memasuki *phase congestion avoidance* dengan mengirimkan satu paket setiap menerima ACK. Pada fase ini, panjang antrian mulai dapat dikontrol oleh *switch*.



(a)



(b)

**Gambar 5.**

Karakteristik panjang antrian dengan 20 sumber TCP, (a) ukuran *buffer* (500,500) sel, (b) ukuran *buffer* (500,300) sel

Untuk mengurangi besarnya sel yang dibuang pada *switch*, maka dapat dilakukan dengan mengeset nilai *threshold* pada *buffer* yaitu menjadi (500,300) sel (Gambar.5b). Penentuan nilai *threshold* dimaksudkan agar *switch* dapat lebih awal untuk mendeteksi jika akan terjadi suatu antrian yang berlebih, sehingga sumber akan segera menurunkan nilai ACR-nya sebesar  $ACR \times RDF$  hingga *Minimum Cell Rate* (MCR).

### 4.4. Total Throughput dan Efisiensi

Total *Throughput* menunjukkan jumlah *byte* yang terkirim oleh semua sumber dengan sukses ke tujuan selama waktu simulasi/transmisi. Untuk melihat performansi *throughput* dihitung juga nilai efisiensi yang didapatkan dengan membandingkan jumlah masing-masing *throughput* terhadap besar *throughput* maksimum yang mungkin.

Perhitungan terhadap Total *Throughput* maksimum tentu memperhitungkan *header-header* dari TCP (20 *byte*), IP (20 *byte*), LLC (8 *byte*), AAL5 (8 *byte*), dan beban sel RM 6%. Untuk  $MSS=512$  *byte* akan menghasilkan 12 sel ATM, sehingga *total throughput* maksimum yang mungkin adalah 105.91 Mbps. Perhitungan yang sama dilakukan untuk  $MSS=1024$  *byte*, sehingga dihasilkan 110.52 Mbps.

Dari hasil simulasi *total throughput* dan efisiensi maksimum yang dicapai untuk jumlah 100 sumber TCP ( $MSS=512$  *byte*), 106.52 Mbps dan 100.57 % (Tabel.2) dan untuk  $MSS=1024$  *byte* adalah 110.99 Mbps dan 100.42 % (Tabel.3).

Tabel.2

Total *Throughput* dan Efisiensi untuk paket dengan MSS =512 byte

Buffer size(cells) (Max,Threshold)	Jumlah sumber	Total Throughput (bit/det)	Efisiensi (%)
(500,500)	10	99,215,348.028	93.679
	20	96,055,905.315	90.696
	40	100,688,413.470	95.070
	60	101,770,877.410	96.092
	80	105,336,068.770	99.458
	100	106,517,914.360	100.574
(1000,1000)	10	99,215,348.028	93.679
	20	99,756,451.921	94.190
	40	100,688,413.470	95.070
	60	101,770,877.410	96.092
	80	105,336,068.770	99.458
	100	106,517,914.360	100.574
(10000,8000)	10	99,215,348.028	93.679
	20	99,756,451.921	94.190
	40	100,688,413.470	95.070
	60	101,770,877.410	96.092
	80	105,336,068.770	99.458
	100	106,517,914.360	100.574

Sedangkan untuk ukuran *buffer* yang kecil sangat dimungkinkan terjadinya pembuangan sel akibat panjang antrian yang melebihi batas maksimum *buffer*, sehingga terjadi penurunan nilai total *throughput*-nya. Seperti terjadi pada ukuran *buffer* (500,500), paket dengan MSS=512 byte dan jumlah sumber 20, nilai total *throughput* menjadi 96.055 Mbps, yang sebelumnya 99.756 Mbps pada ukuran *buffer* (1000,1000)sel maupun (10000,8000)sel (Tabel.2).

Dari hasil simulasi, didapatkan selama ukuran *buffer* dapat menangani antrian sel yang terjadi, maka besar total *throughput* untuk jumlah sumber tertentu cenderung tidak berubah walaupun ukuran *buffer* yang digunakan terus diperbesar.

#### 4.5. Tingkat *Fairness*

Tingkat keadilan (*indeks fairness*) menunjukkan keadilan pada tiap sumber TCP dalam pembagian *bandwidth*. Pada analisa ini, *indeks fairness* diukur berdasarkan performansi *throughput*,

Tabel.3

Total *Throughput* dan Efisiensi untuk paket dengan MSS =1024 byte

Buffer size(cells) (Max,Threshold)	Jumlah Sumber	Total Throughput (bit/det)	Efisiensi (%)
(500,500)	10	103,275,001.560	93.445
	20	104,058,824.590	94.154
	40	104,906,761.990	94.921
	60	105,797,231.990	95.727
	80	109,252,715.450	98.853
	100	110,988,874.160	100.424
(1000,1000)	10	103,275,001.560	93.445
	20	104,058,824.590	94.154
	40	104,906,761.990	94.921
	60	105,797,231.990	95.727
	80	109,252,715.450	98.853
	100	110,988,874.160	100.424
(10000,8000)	10	103,275,001.560	93.445
	20	104,058,824.590	94.154
	40	104,906,761.990	94.921
	60	105,797,231.990	95.727
	80	109,252,715.450	98.853
	100	110,988,874.160	100.424

dimana *indeks fairness* akan tinggi jika setiap sumber menghasilkan nilai *throughput* yang sama atau hampir sama, sedangkan penurunan nilai *indeks fairness* menunjukkan adanya ketidakadilan pada tiap sumbernya dikarenakan adanya selisih perolehan besar *throughput* pada masing-masing sumber yang cukup besar.

Dari hasil simulasi, penambahan jumlah sumber menyebabkan adanya sedikit penurunan nilai *indeks fairness* baik pada paket TCP MSS= 512 byte, maupun MSS=1024 byte. Sedangkan penurunan *indeks fairness* yang sangat drastis terjadi pada jumlah sumber 20 untuk ukuran paket MSS = 512 byte, yaitu menjadi 0.852 yang sebelumnya diatas 0.99. Hal ini disebabkan pengaruh dari penurunan nilai total *throughput* pada *buffer switch* ukuran kecil. Berikut diperlihatkan nilai *indeks fairness* pada berbagai ukuran *buffer* (Tabel 4).



Tabel. 4

Tingkat *Fairness* untuk paket dengan MSS =512 byte dan MSS=1024 byte.

Buffer size (cells) (Max,Threshold)	Jumlah Sumber	Fairness Indeks (MSS=512byte)	Fairness Indeks (MSS=1024byte)
(500,500)	2	0.99999986	0.99965593
	5	0.99999911	0.99386279
	10	0.99997729	0.99327476
	20	0.85227335	0.99987633
	40	0.99918401	0.99930918
	60	0.99800984	0.99963295
	80	0.99555417	0.99606638
	100	0.99309402	0.99256291
(1000,1000)	2	0.99999986	0.99965593
	5	0.99999911	0.99386279
	10	0.99997729	0.99327476
	20	0.99987138	0.99987633
	40	0.99918401	0.99930918
	60	0.99800984	0.99963295
	80	0.99555417	0.99606638
	100	0.99309402	0.99256291
(10000,8000)	2	0.99999986	0.99965593
	5	0.99999911	0.99386279
	10	0.99997729	0.99327476
	20	0.99987138	0.99987633
	40	0.99918401	0.99930918
	60	0.99800984	0.99963295
	80	0.99555417	0.99606638
	100	0.99309402	0.99256291

## 5. Kesimpulan

Beberapa kesimpulan yang dapat diambil dari hasil penelitian ini melalui simulasi adalah :

1. Total *throughput* dan efisiensi untuk paket TCP 512 byte, dengan jumlah sumber 20 mengalami penurunan masing-masing 3% dan 4% pada ukuran *buffer* maksimum, *threshold* (500,500) sel.
2. Nilai *Indeks Fairness* untuk semua sumber TCP dengan ukuran paket MSS=1024 byte dapat mencapai nilai maksimumnya, yaitu diatas 0.99
3. Untuk mengurangi pembuangan sel di *buffer switch* pada saat jaringan kongesti perlu dilakukan *setting* terlebih dahulu terhadap nilai *threshold* di *buffer*-nya, seperti yang telah dilakukan pada jumlah

sumber 20 dengan ukuran *buffer* maksimum = 500 sel dan *threshold* = 500 sel menjadi *threshold* = 300 sel.

## Daftar Acuan

1. K. Djemame, M. Kara. " *A new rate coordination protocol for TCP and ABR congestion Control* ", School of Computer Studies, University of Leeds. 2000.
2. Cai, Xiangrong, " *The Performance of TCP over ATM ABR and UBR Services* ", Juli 2000.
3. Sonia Fahmi, Raj Jain, Rohit Goyal and Bobby Vandalore, " *Design and Simulation of ATM-ABR End System Congestion Control* ", Transaction of the Society for Computer Simulation, Vol 78, Issue 3, March/April 2002.
4. Sonia Fahmi, Raj Jain, Rohit Goyal and Bobby Vandalore, " *On Determining the Fair Bandwidth Share for ABR Connection in ATM Network* ", Accepted to appear in Journal of high speed network, 2002
5. S. Kalyanaraman, Raj Jain, Sonia Fahmy, and Rohit Goyal. " *Buffer Requirements for TCP/IP over ABR* ". The Ohio State University. 1997.
6. Prihanto Harry. " *Adaptive Thresholds of Buffer to Solve the Beat-Down Problem of Rate Control in ATM Network* " .IEICE, Trans. Comm, Vol.E.83-B, No.2 February 2000.
7. Raj Jainm S.Kalyanaraman, " *Source behavior for ATM ABR traffic Management* ".IEEE, Communication Magazine, November 1996.