

BAB IV

HASIL DAN PEMBAHASAN

Pembahasan yang dilakukan merupakan hasil dari percobaan terhadap parameter-parameter yang telah ditentukan. Setelah itu dilakukan analisis untuk mendapat perbandingan unjuk kerja protokol TCP Vegas dan UDP dengan menggunakan data *streaming*.

4.1 Kebutuhan Sistem

Sebelum dilakukan simulasi dan analisis perbandingan unjuk kerja protokol TCP Vegas dan UDP dengan menggunakan data *streaming* dibutuhkan perangkat keras dan perangkat lunak dengan kondisi tertentu agar simulasi dapat berjalan dengan baik. Adapun kebutuhan perangkat lunak dan perangkat keras sebagai berikut.

4.1.1 Kebutuhan Perangkat Keras

Syarat minimal perangkat keras yang digunakan untuk tugas akhir ini seperti pada Tabel 4.1:

Tabel 4.1. Kebutuhan Perangkat Keras

Perangkat Keras	Spesifikasi
Processor	Dual core
Memori	1GB RAM
Sistem Operasi	Ubuntu 12.04

4.1.2 Kebutuhan Perangkat Lunak

Pada Tabel 4.2 akan dijelaskan tentang beberapa kebutuhan perangkat lunak yang menunjang dalam tugas akhir ini.

Tabel 4.2. Kebutuhan Perangkat Lunak

Perangkat Lunak	Uraian
Network Simulator 2	Aplikasi yang digunakan untuk menjalankan proses simulasi
Perl	Aplikasi yang digunakan untuk mengolah file .tr yang merupakan data output dari simulasi.
Libreoffice Calc	Aplikasi yang digunakan untuk mengolah hasil dari perl dan membuat grafik dari data hasil simulasi.

4.2 Data Simulasi

Data simulasi ini digunakan untuk menjalankan percobaan dalam simulasi.

Data *streaming* yang akan dijalankan di atas TCP Vegas dan UDP adalah *VOIP* dan *Video*. Waktu yang digunakan untuk menjalankan simulasi di bawah ini selama 20 detik. Untuk mendapatkan hasil yang lebih akurat percobaan dilakukan sebanyak 5 kali di setiap data *streaming*. Tabel 4.3 dan Tabel 4.4 merupakan data-data yang akan digunakan:

Tabel 4.3. Paket *size* dan *bit rate*

Data Streaming	Data	Protokol	
		TCP Vegas	UDP
VOIP	Paket Size	160 Kb	160 Kb
	Bit rate	64 Kb	64 Kb
VIDEO	Paket Size	1400 Kb	1400 Kb
	Bit rate	256 Kb	256 Kb

Tabel 4.4. Ukuran *bandwidth*

Percobaan	Data Streaming	Bandwidth
1	VOIP	70 Kb
2		86 Kb
3		92 Kb
4		103 Kb
5		256 Kb
1	VIDEO	300 Kb
2		340 Kb
3		380 Kb
4		420 Kb
5		840 Kb

4.3 Hasil dan Pembahasan

Setelah melakukan percobaan sesuai dengan data pada Tabel 4.3 dan Tabel 4.4 didapatkan hasil seperti pada Tabel 4.5 dan Tabel 4.6:

Tabel 4.5. Hasil Perhitungan *Utilisasi bandwidth* dan *Packet Loss VOIP*

Perc	Bandwidth	Parameter			
		Utilisasi Bandwidth		Packet Loss	
		TCP Vegas	UDP	TCP Vegas	UDP
1	70 Kb	9,60%	85,67%	6,25%	0,11%
2	86 Kb	23,07%	69,80%	1,27%	0,21%
3	92 Kb	29,15%	65,39%	0,24%	0,32%
4	103 Kb	35,39%	58,78%	0,00%	0,00%
5	256 Kb	60,10%	23,65%	0,00%	0,00%

Tabel 4.6. Hasil Perhitungan *Latency* dan *Jitter VOIP*

Perc	Bandwidth	Parameter			
		Latency		Jitter	
		TCP Vegas	UDP	TCP Vegas	UDP
1	70 Kb	276,86 ms	288 ms	42,16 ms	36,11 ms
2	86 Kb	197,8 ms	184,54 ms	32,98 ms	29,94 ms
3	92 Kb	192,85 ms	188,73 ms	29,38 ms	29,12 ms
4	103 Kb	149,46 ms	147,71 ms	22,13 ms	20,02 ms
5	256 Kb	68,56 ms	67,98 ms	7,57 ms	6,99 ms

Pada tabel 4.5 dan 4.6 merupakan hasil perhitungan *Utilisasi bandwidth*, *packet loss*, *latency* dan *jitter* pada data *streaming VOIP* dengan *bandwidth* 70 Kb, 86 Kb, 92 Kb, 103 Kb dan 256 Kb.

Tabel 4.7. Hasil Perhitungan *Utilisasi bandwidth* dan *Packet Loss* Video

Perc	Bandwidth	Parameter			
		Utilisasi Bandwidth		Packet Loss	
		TCP Vegas	UDP	TCP Vegas	UDP
1	300 Kb	17,73%	77,12%	5,83%	3,79%
2	340 Kb	24,21%	70,12%	2,61%	0,47%
3	380 Kb	31,39%	62,86%	0,93%	0,33%
4	420 Kb	37,07%	57,33%	0,00%	0,00%
5	840 Kb	63,13%	28,67%	0,00%	0,00%

Tabel 4.8. Hasil Perhitungan *Latency* dan *Jitter Video*

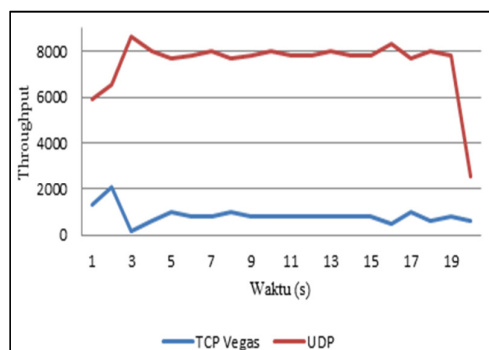
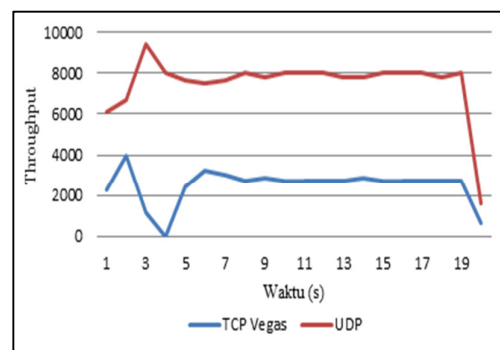
Perc	Bandwidth	Parameter			
		Latency		Jitter	
		TCP Vegas	UDP	TCP Vegas	UDP
1	300 Kb	341,33 ms	325,6 ms	57,81 ms	43,2 ms
2	340 Kb	296,13 ms	293,43 ms	40,94 ms	38,32 ms
3	380 Kb	273,51 ms	270,5 ms	37,02 ms	33,83 ms
4	420 Kb	245,5 ms	231,51 ms	27,22 ms	24,59 ms
5	840 Kb	106,15 ms	101,63 ms	12,41 ms	10,62 ms

Pada tabel 4.7 dan 4.8 merupakan hasil perhitungan *Utilisasi bandwidth*, *packet loss*, *latency* dan *jitter* pada data *streaming video* dengan *bandwidth* 300 Kb, 340 Kb, 380 Kb, 420 Kb dan 840 Kb.

Di bawah ini adalah hasil percobaan data *streaming VOIP* berdasarkan parameter uji *Utilisasi bandwidth*, *packet loss*, *latency* dan *jitter* yang disajikan dalam bentuk grafik, untuk memudahkan dalam membandingkan kedua protokol.

4.3.1 Utilisasi bandwidth

Sebelum didapatkan nilai dari *Utilisasi bandwidth* terlebih dahulu dicari nilai *throughput*, dibawah ini hasil dari nilai *throughput*.

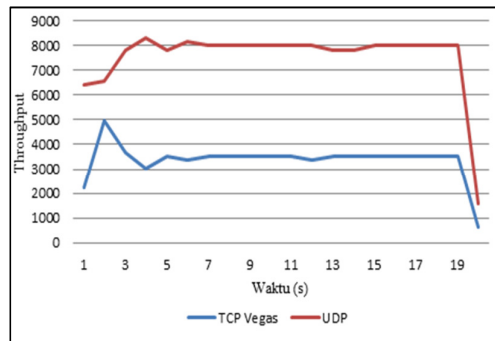
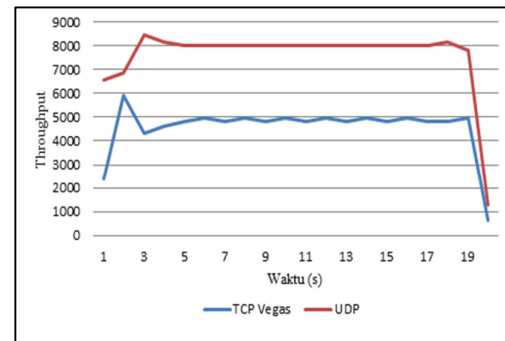
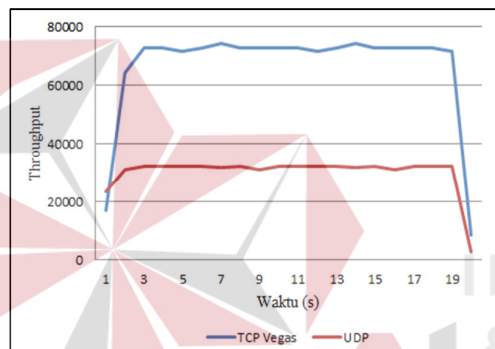
Gambar 4.1. *Throughput bandwidth 70Kb*Gambar 4.2. *Throughput bandwidth 86Kb*



INSTITUT BISNIS
& INFORMATIKA

stikom

SURABAYA

Gambar 4.3. *Throughput bandwidth 92Kb*Gambar 4.4. *Throughput bandwidth 103Kb*Gambar 4.5. *Throughput bandwidth 256 Kb*

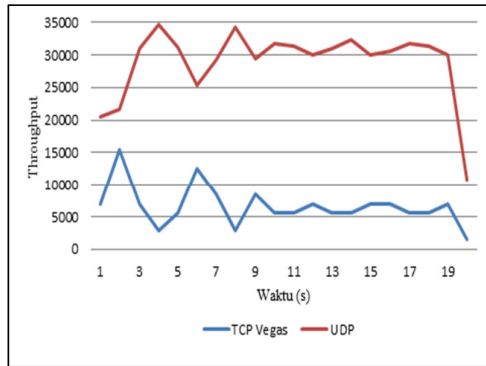
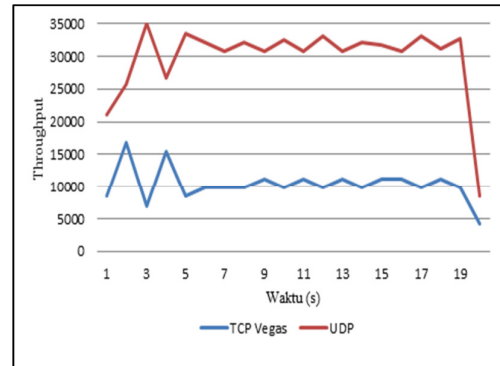
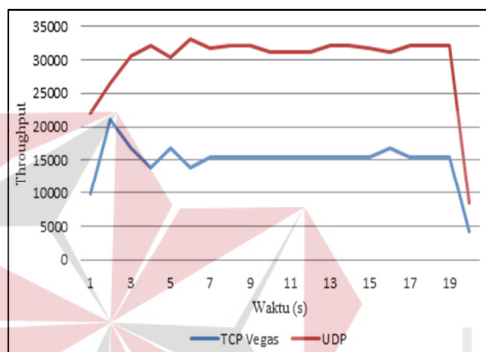
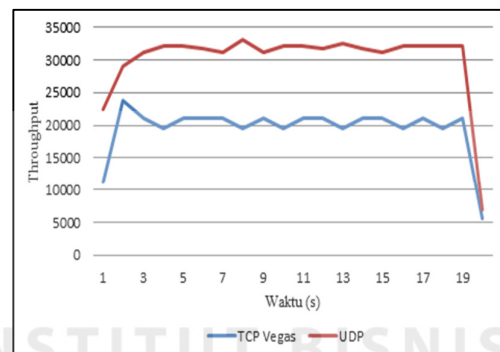
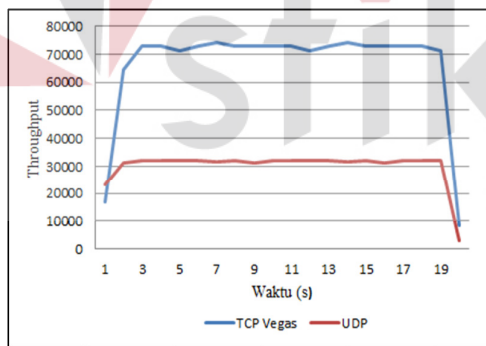
Gambar 4.1-Gambar 4.5 adalah grafik nilai *throughput* dari TCP Vegas dan UDP dengan data VOIP. Gambar 4.1 adalah nilai *throughput* dengan *bandwidth* 70 Kb. Gambar 4.2 nilai *throughput* dengan *bandwidth* 86 Kb. Gambar 4.3 adalah nilai *throughput* dengan *bandwidth* 92 Kb, Gambar 4.4 nilai *throughput* dengan *bandwidth* 103Kb. Sedangkan Gambar 4.5 nilai *throughput* dengan *bandwidth* 256 Kb.



INSTITUT BISNIS
& INFORMATIKA

stikom

SURABAYA

Gambar 4.6. *Throughput bandwidth 300Kb*Gambar 4.7. *Throughput bandwidth 340Kb*Gambar 4.8. *Throughput bandwidth 380Kb*Gambar 4.9. *Throughput bandwidth 420Kb*Gambar 4.10. *Throughput bandwidth 840Kb*

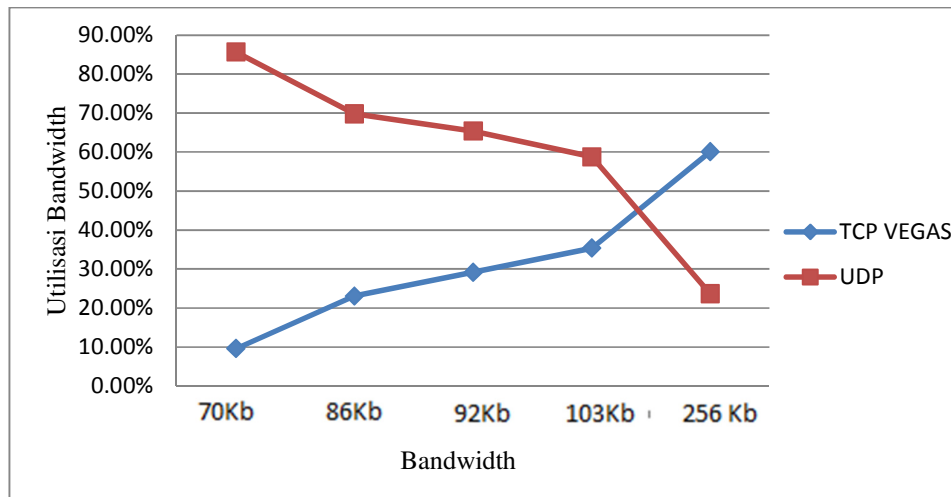
Gambar 4.6 – Gambar 4.10 adalah grafik nilai *throughput* dari TCP Vegas dan UDP dengan data *Video*. Gambar 4.6 adalah nilai *throughput* dengan *bandwidth* 300 Kb. Gambar 4.7 nilai *throughput* dengan *bandwidth* 340 Kb. Gambar 4.8 adalah nilai *throughput* dengan *bandwidth* 380 Kb, Gambar 4.9 nilai

throughput dengan *bandwidth* 420 Kb sedangkan Gambar 4.10 merupakan nilai *throughput* dengan *bandwidth* 840 Kb.

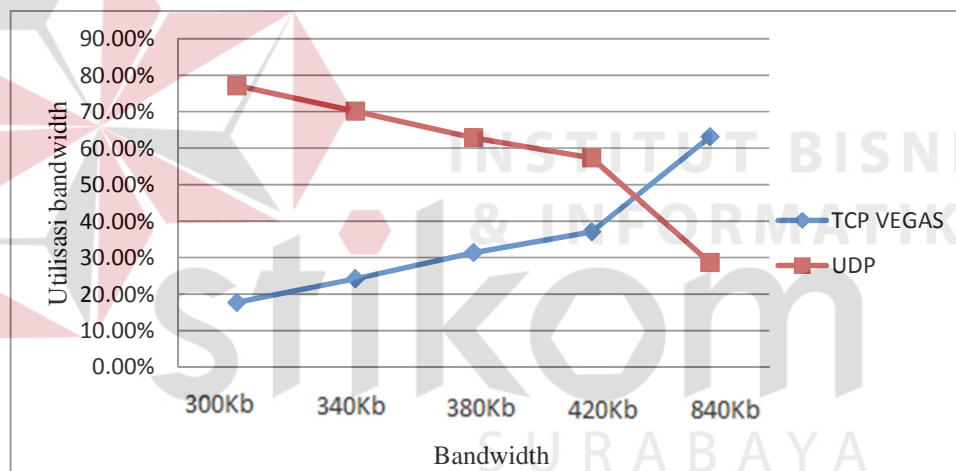
Grafik *throughput* pada protokol UDP dan TCP Vegas menunjukkan bahwa nilai *throughput* protokol UDP memiliki nilai konstan 8000 Bps untuk data *streaming* VoIP dan 32000 Bps untuk data *streaming video* selama pengiriman berlangsung. *Throughput* TCP Vegas pada percobaan pertama sampai keempat pada data *streaming* VoIP (Gambar 4.1-Gambar 4.4) masing-masing bernilai sekitar 750, 2750, 3500 dan 4875 Bps sedangkan untuk data *streaming video* (Gambar 4.6-Gambar 4.9) masing-masing bernilai 5500, 10500, 15500 dan 20500 Bps.

Seperti terlihat pada Gambar 4.1 – Gambar 4.8, nilai karakteristik *throughput* pada TCP Vegas berbeda dengan karakteristik *throughput* pada UDP. Pada TCP Vegas pertambahan nilai *throughput* berbanding lurus dengan kenaikan *bandwidth*. Hal ini disebabkan karena aplikasi VOIP dan *Video* streaming pada protokol UDP dalam simulasi digunakan *constant bit rate* (CBR) sehingga *throughput* pada protokol UDP mengidentifikasi nilai yang konstan. Sedangkan pada protokol TCP Vegas nilai *throughput* bertambah seiring dengan pertambahan *bandwidth*, hal ini disebabkan adanya algoritma *congestion control* pada TCP Vegas yang mana protokol berusaha menyelesaikan *throughput* dengan kapasitas *link* (*bandwidth*) .

Setelah didapatkan nilai *throughput* selanjutnya dihitung dengan persamaan 2.1 untuk mendapatkan nilai *Utilisasi bandwidth*. Rata-rata *Utilisasi bandwidth* ditunjukkan dalam Gambar 4.11.



Gambar 4.11. Grafik perbandingan *Utilisasi bandwidth VOIP*



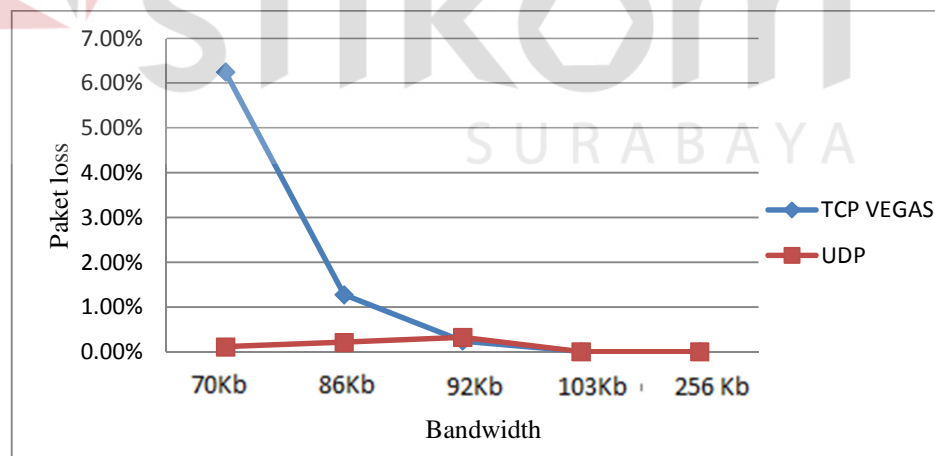
Gambar 4.12. Grafik perbandingan *Utilisasi bandwidth Video*

Gambar 4.11 menunjukkan grafik perbandingan *utilisasi bandwidth* dari protokol TCP Vegas dan UDP dengan menggunakan data *streaming VOIP* sedangkan Gambar 4.12 nilai *utilisasi bandwidth* dengan menggunakan data *streaming Video*. Pada saat bandwidth kecil *utilisasi bandwidth* UDP nampak lebih besar daripada utilisasi pada TCP Vegas. Namun pada saat bandwidth

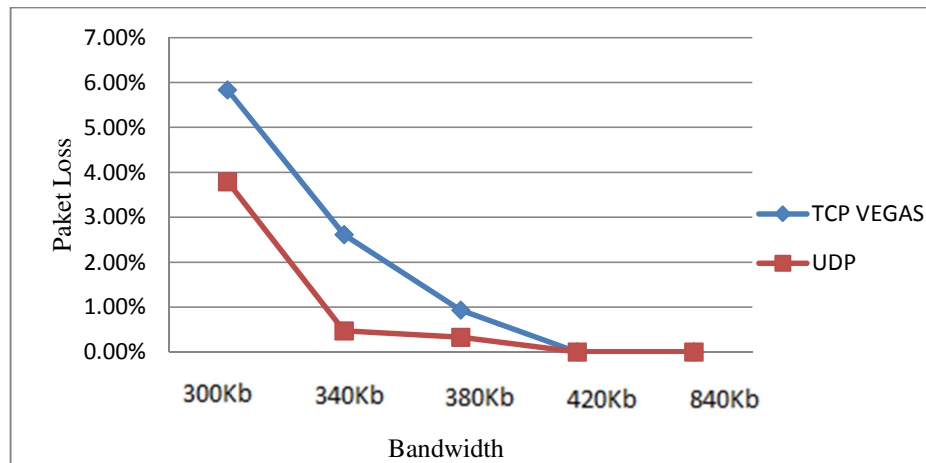
bottleneck link semakin besar maka nampak *utilisasi bandwidth* TCP Vegas lebih besar daripada UDP.

Hal ini disebabkan karena UDP menggunakan menggunakan paket CBR yang punya kecepatan konstan sehingga ketika bandwidth pada bottleneck link besar kecepatan bandwidth tetap konstan, dengan demikian *utilisasi bandwidth* pada UDP akan semakin kecil ketika bandwidth bertambah. Karena UDP tidak memiliki mekanisme congestion control maka pengiriman data akan tetap konstan. Sementara itu TCP Vegas menggunakan congestion control dalam pengiriman data. Pada saat bandwidth bottleneck link diperbesar mekanisme congestion control mengatur kecepatan pengiriman data mengikuti besarnya bandwidth. Dengan demikian pada TCP Vegas saat bandwidth diperbesar maka *utilisasi bandwidth* akan semakin meningkat.

4.3.2 Packet loss



Gambar 4.13. Grafik Perbandingan *Packet loss* VOIP

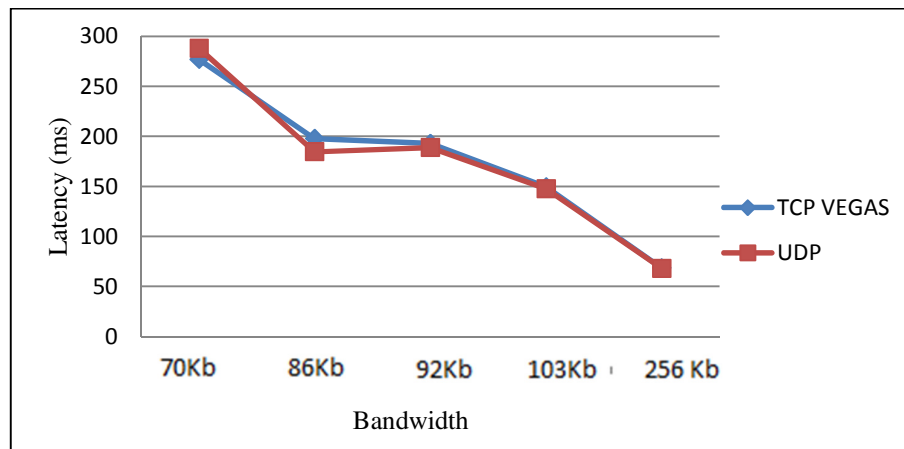


Gambar 4.14. Grafik perbandingan *Packet loss* Video

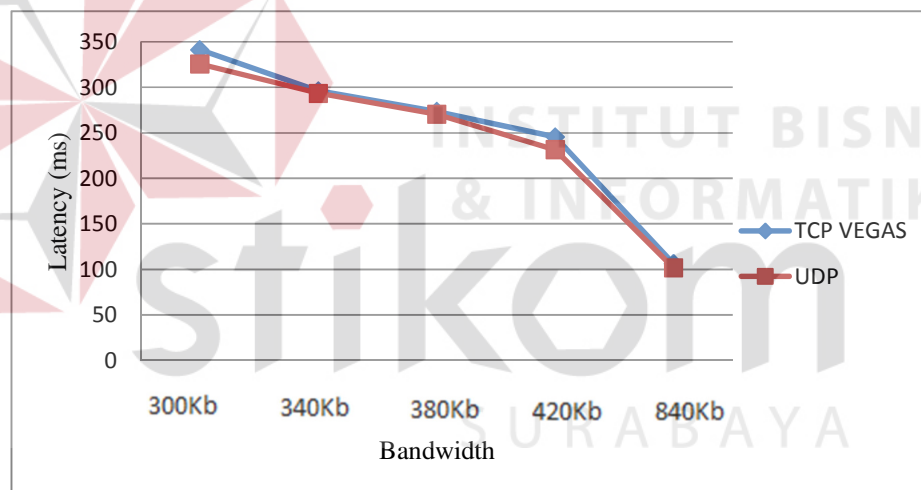
Gambar 4.13 menunjukkan grafik perbandingan *Packet loss* dari protokol TCP Vegas dan UDP dengan menggunakan data *streaming VOIP* sedangkan Gambar 4.14 menunjukkan perbandingan *packet loss* dengan menggunakan data *streaming Video*. Berdasarkan hasil pengujian didapatkan bahwa nilai *packet loss* pada TCP Vegas lebih tinggi dibandingkan nilai *packet loss* pada UDP. Hal ini dikarenakan kapasitas *bandwidth* pada *bottleneck link* banyak digunakan oleh UDP selama proses pengiriman. Sehingga TCP Vegas hanya menempati sebagian kecil total *bandwidth* yang tersedia. Misalnya untuk uji coba dengan *bandwidth* 70 Kb *Utilisasi bandwidth* TCP Vegas hanya 9,6% sedangkan *utilisasi* UDP mencapai 85,67%. Hal ini ditunjukkan pada Tabel 4.5.

Menurut *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON)* nilai perbandingan *packet loss* pada VOIP dan Video masuk pada kategori bagus karena besarnya *packet loss* pada VoIP masih dibawah 7 %. Sedangkan untuk Video dibawah 6%.

4.3.3 Latency



Gambar 4.15. Grafik perbandingan *Latency* VOIP



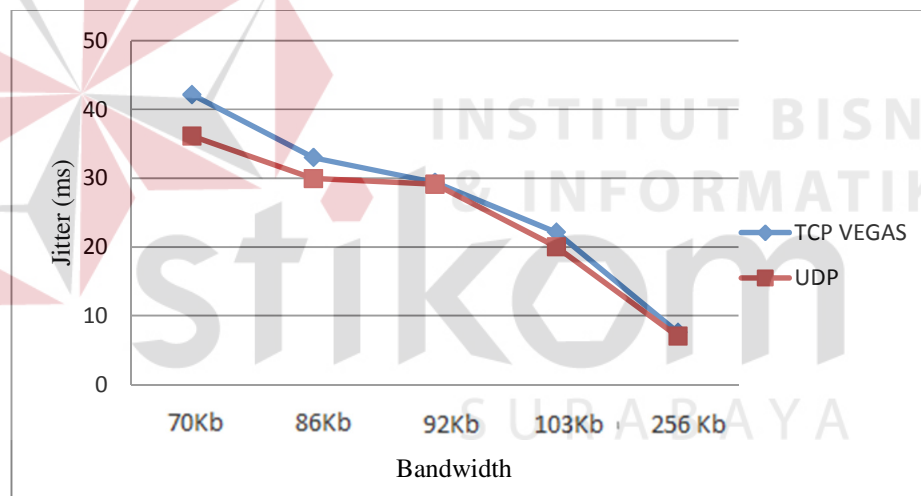
Gambar 4.16. Grafik perbandingan *Latency* Video

Gambar 4.15 menunjukkan grafik perbandingan *latency* dari protokol TCP Vegas dan UDP dengan menggunakan data *streaming VOIP*. Sedangkan Gambar 4.16 menunjukkan grafik perbandingan *latency* dengan menggunakan data *streaming Video*. Berdasarkan hasil pengujian didapatkan nilai *latency* pada TCP Vegas lebih tinggi dibandingkan nilai *latency* pada UDP. Gambar 4.1-Gambar 4.8 menunjukkan bahwa nilai *throughput* UDP lebih besar daripada TCP Vegas

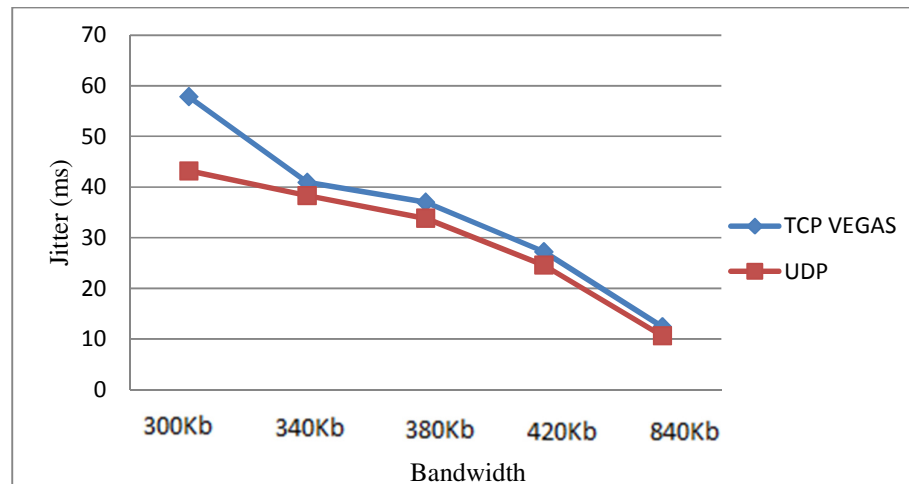
sehingga secara keseluruhan latency TCP Vegas menjadi lebih tinggi karena paket atau segmen TCP menunggu lama pada buffer node *bottleneck link*. (The VINT Project, 2011) Paket data TCP akan dibuang ketika kapasitas buffer pada node telah penuh sehingga menyebabkan *packet loss* TCP Vegas tinggi.

Menurut *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON)* nilai perbandingan *latency* pada *VOIP* masuk pada kategori bagus karena besarnya *latency* masih dibawah 300 ms. Sedangkan untuk *Video* tergolong mempunyai kategori sedang karena nilai *latency* dibawah 350 ms.

4.3.4 Jitter



Gambar 4.17. Grafik perbandingan *Jitter* VOIP



Gambar 4.18. Grafik perbandingan *Jitter* Video

Gambar 4.17 menunjukkan grafik perbandingan *jitter* dari protokol TCP Vegas dan UDP dengan menggunakan data *streaming VOIP*. Sedangkan gambar 4.18 perbandingan *jitter* dengan menggunakan data *streaming Video*. Berdasarkan hasil pengujian didapatkan nilai *jitter* pada TCP Vegas lebih tinggi dibandingkan nilai *jitter* pada UDP. Hal ini dikarenakan besarnya interferensi yang terjadi dari protokol UDP sehingga delay TCP Vegas mengalami variasi yang tinggi. Delay paket sebelumnya kecil karena ada *bandwidth* yang bisa digunakan oleh TCP Vegas sedangkan paket selanjutnya harus menunggu lebih lama untuk proses pengiriman karena berada dalam antrian pada buffer. Hal inilah yang menyebabkan *jitter* TCP Vegas lebih tinggi daripada UDP.

Menurut *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON)* nilai perbandingan *jitter* pada *VOIP* dan *Video* masuk pada kategori bagus karena besarnya *jitter* masih dibawah 60ms.