

BAB 5 HASIL DAN PEMBAHASAN

Bab ini akan menunjukkan hasil dan analisis yang dilakukan dalam pengerjaan tugas akhir. Ada dua pembahasan pada bab ini yaitu analisis dan perbandingan dari hasil analisis. Pembahasan analisis akan berisi analisis dari hasil performansi kedua aplikasi *file sharing* yang didapatkan dari pengujian. Sedangkan untuk pembahasan perbandingan akan berisi perbandingan dari hasil analisis yang telah dilakukan.

5.1 Analisis hasil pengujian

Pada subbab ini akan dibahas analisa hasil pengujian dari masing-masing aplikasi *file sharing* yang telah dilakukan sebelumnya. Dari hasil masing-masing aplikasi *file sharing*, akan dilakukan analisis. Selanjutnya dari hasil analisis masing-masing aplikasi *file sharing*, akan dibandingkan untuk mengetahui perbedaan performansi antara kedua aplikasi *file sharing* yang dibandingkan dalam penelitian ini.

5.1.1 Analisis hasil pengujian aplikasi myjxta

Dari pengujian aplikasi *file sharing* yang telah dilakukan pada sub bab 4.3.1, yaitu pengujian aplikasi myjxta didapatkan hasil pengujian yaitu nilai *throughput* dan rata-rata *delay* saat melakukan proses *file sharing* yang ditunjukkan pada tabel 5.1 yang selanjutnya dilakukan analisis hasil pengujian. Pengujian yang dilakukan pada pengujian aplikasi myjxta hanya menggunakan satu skema yaitu hanya menerima *file* dari satu *seeder* walaupun beberapa *seeder* telah melakukan *share* terhadap *file* yang diujikan. Berikut adalah hasil pengujian aplikasi myjxta pada Tabel 5.1.

Ukuran File	<i>Throughput</i> (Mb/s)	Rata-rata <i>Delay</i> (s)
50MB	10.99792216	0.000999251
100MB	11.06360217	0.00098824
200MB	11.06432392	0.000937512
512MB	11.06599133	0.000464059

Tabel 5.1 Hasil pengujian aplikasi myjxta

Dari Tabel 5.1 didapatkan hasil pengujian aplikasi myjxta terhadap performansi *file sharing* saat melakukan proses *file sharing* dengan ukuran *file* sebesar 50MB menghasilkan *throughput* sebesar 10.99792216 Mb/s dengan rata-rata *delay* 0.000999251s. Sedangkan untuk ukuran *file* yang lebih besar yaitu 100MB, 200MB, dan 512MB masing-masing menghasilkan *throughput* sebesar 11.06360217 Mb/s, 11.06432392 Mb/s, dan 11.05599133 Mb/s. dan rata-rata *delay* 0.00098824s, 0.000937512s, 0.000464059s. Hasil pengujian menunjukkan bahwa semakin besar ukuran *file* yang diujikan maka semakin besar *throughput* yang dihasilkan dan semakin kecil nilai rata-rata *delay* yang dihasilkan.

5.1.2 Analisis hasil pengujian aplikasi Gnucleus

Dari pengujian aplikasi *file sharing* yang telah dilakukan pada pengujian aplikasi Gnucleus didapatkan nilai *throughput* dan rata-rata *delay* saat melakukan proses *file sharing* yang ditunjukkan pada Tabel 5.2, Tabel 5.3, Tabel 5.4, dan Tabel 5.5 dan dilakukan analisis hasil pengujian. Pengujian yang dilakukan pada pengujian aplikasi Gnucleus menggunakan seluruh skema yang telah dibuat pada perancangan lingkungan pengujian sistem sehingga ada 14 skema yang dilakukan untuk pengujian aplikasi Gnucleus. Berikut adalah hasil pengujian aplikasi Gnucleus.

Tabel 5.2, Tabel 5.3, Tabel 5.4, dan Tabel 5.5 menunjukkan hasil dari pengujian yang dilakukan oleh aplikasi *file sharing* dalam melakukan proses *file sharing* yang skemanya telah diurutkan berdasarkan jumlah *seeder*. Hal itu dilakukan untuk mengetahui perbedaan performansi saat menggunakan jumlah *seeder* yang berbeda-beda dengan ukuran *file* yang sama. Nilai *throughput* terkecil yang dihasilkan selalu didapatkan dari skema A dengan menggunakan satu *seeder* kecuali pada pengujian dengan ukuran *file* 50MB yaitu nilai *throughput* terkecil dihasilkan dari skema E yang menggunakan 4 *seeder*. Sedangkan nilai *throughput* terbesar dihasilkan masing-masing oleh skema G pada ukuran *file* 50MB dan 100MB, skema O pada ukuran *file* 200MB, dan skema F pada ukuran *file* 512MB yang seluruhnya menggunakan paling sedikit tiga dari jumlah *seeder*. Perbandingan nilai *throughput* pada masing-masing skema dan ukuran *file* ditunjukkan dengan grafik pada Gambar 5.14.

	Skema	<i>Throughput</i> (Mb/s)	Rata-rata <i>Delay</i> (s)
50MB	A	4.291459308	0.002743954
	K	4.313200923	0.003136948
	B	4.436183904	0.002695017
	N	4.42433718	0.002701684
	L	4.314759727	0.002890415
	C	4.38940841	0.002722532
	D	4.306240624	0.0026807
	O	4.370657007	0.002735674
	M	4.36096158	0.002923292
	E	4.178750204	0.00277473
	F	4.407502677	0.002675204
	H	4.294984172	0.002957791
	G	4.487656887	0.002615402
	I	4.3300841	0.004111921

Tabel 5.2 Hasil pengujian dengan ukuran *file* 50MB

100MB	Skema	Throughput (Mb/s)	Rata-rata Delay (s)
	A	4.298981781	0.002717124
	K	4.327641683	0.002923605
	B	4.369657439	0.002737918
	N	4.332891241	0.002733904
	L	4.35340889	0.003171409
	C	4.385069254	0.002726274
	D	4.40532793	0.002689569
	O	4.371287145	0.002733756
	M	4.322555733	0.002760689
	E	4.361345602	0.002739698
	F	4.396036475	0.002714451
	H	4.324257311	0.002932958
	G	4.571834007	0.0025731
I	4.338748989	0.002996253	

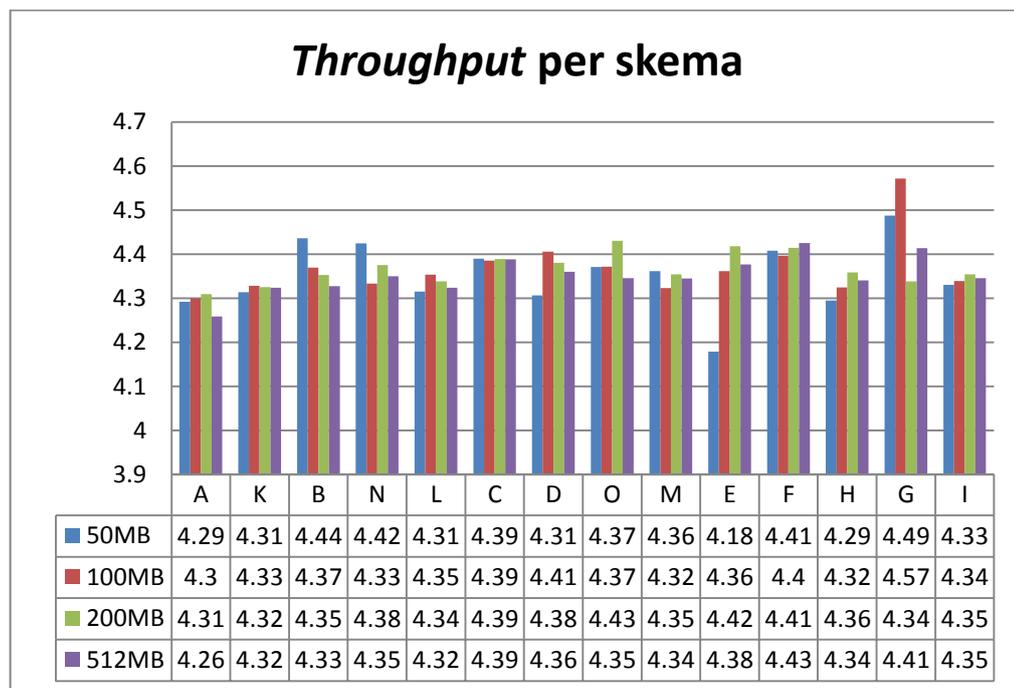
Tabel 5.3 Hasil pengujian dengan ukuran file 100MB

200MB	scheme	Throughput (Mb/s)	Rata-rata Delay (s)
	A	4.309058174	0.00271349
	K	4.32486041	0.002879559
	B	4.352947489	0.002749467
	N	4.375416366	0.002734673
	L	4.338209464	0.002869283
	C	4.389043361	0.002724936
	D	4.380396272	0.002692006
	O	4.430144824	0.002694137
	M	4.354128187	0.002788767
	E	4.417942669	0.002705603
	F	4.41419224	0.002696157
	H	4.358599262	0.002900037
	G	4.338135199	0.002702451
I	4.354128187	0.003097397	

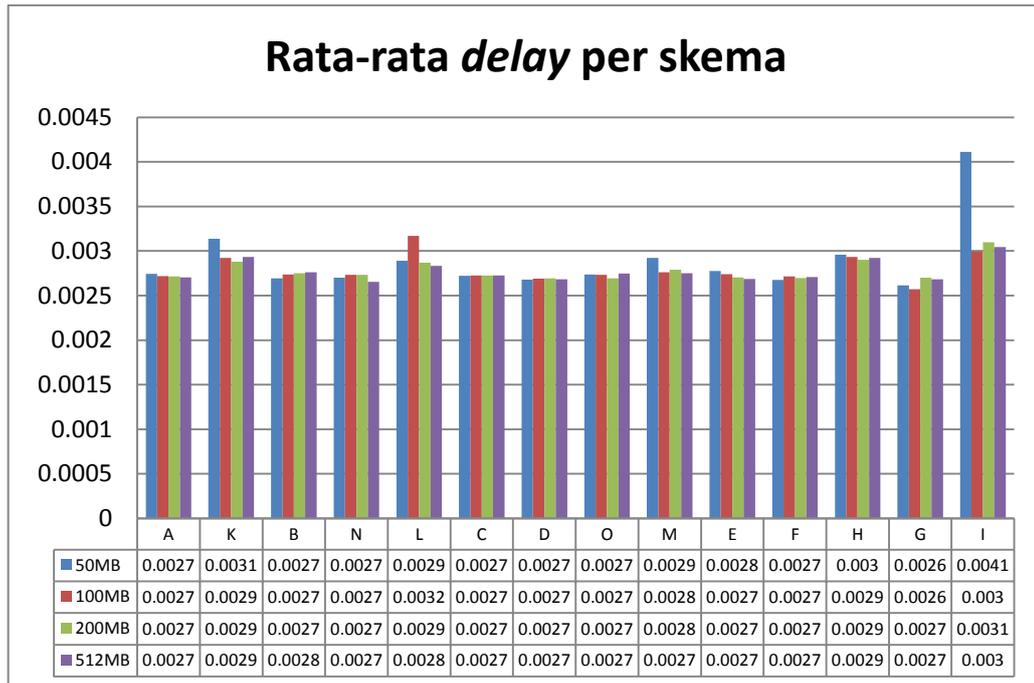
Tabel 5.4 Hasil pengujian dengan ukuran file 200MB

512MB	scheme	Throughput (Mb/s)	Rata-rata Delay (s)
	A	4.25860966	0.002705565
	K	4.323978006	0.002933598
	B	4.327620761	0.002763256
	N	4.349723811	0.002655898
	L	4.323610255	0.002833464
	C	4.38820623	0.002724831
	D	4.35959671	0.002681502
	M	4.345470722	0.002746467
	O	4.344436141	0.002749761
	E	4.376518442	0.002685236
	F	4.425373214	0.002708695
	H	4.340484038	0.002923446
	G	4.413563919	0.002682091
I	4.345470722	0.003043241	

Tabel 5.5 Hasil pengujian dengan ukuran file 512MB



Gambar 5.1 Grafik hasil pengujian terhadap throughput pada masing-masing skema dan ukuran file

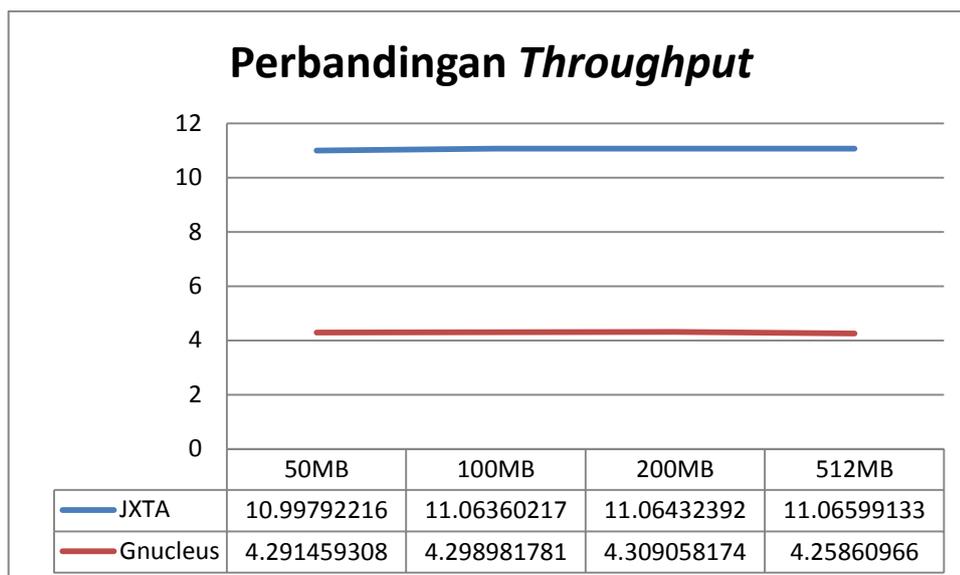


Gambar 5.2 Grafik hasil pengujian terhadap rata-rata *delay* pada masing-masing skema dan ukuran *file*

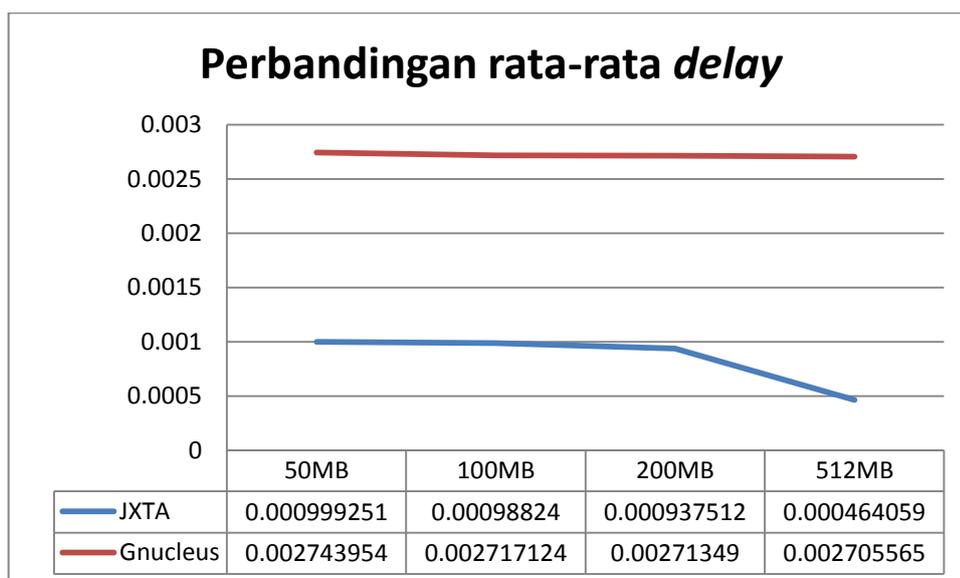
Untuk rata-rata *delay*, nilai terkecil masing-masing didapatkan dari skema G pada ukuran *file* 50MB dan 100MB, skema D pada ukuran *file* 200MB, dan skema N pada ukuran *file* 512MB. Sedangkan nilai terbesar seluruhnya didapatkan dari skema I kecuali pada pengujian dengan ukuran *file* 100MB yaitu pada skema L. Perbandingan nilai rata-rata *delay* pada masing-masing skema dan ukuran *file* ditunjukkan dengan grafik pada Gambar 5.15.

5.2 Perbandingan antar aplikasi

Subbab ini membahas perbandingan dari hasil pengujian performansi pada kedua aplikasi *file sharing* yang telah dilakukan sebelumnya. Hasil pengujian yang dijadikan perbandingan yaitu hasil dari pengujian aplikasi JXTA dan hasil dari pengujian aplikasi Gnucleus dengan menggunakan skema A. Berikut adalah perbandingan hasil pengujian kedua aplikasi *file sharing*.



Gambar 5.3 Grafik perbandingan *throughput* antara JXTA dan Gnutella



Gambar 5.4 Grafik perbandingan rata-rata *delay* antara JXTA dan Gnutella

Dari Gambar 5.15 terlihat perbedaan nilai *throughput* yang signifikan yang dihasilkan oleh kedua aplikasi. Begitu juga pada Gambar 5.16 terlihat nilai rata-rata *delay* yang dihasilkan oleh kedua aplikasi menunjukkan perbedaan yang signifikan. Faktor yang mempengaruhi perbedaan performansi terhadap proses *file sharing* adalah:

- JXTA memiliki jumlah *protocol* lebih banyak pada *core*-nya dan didesain untuk dapat bekerja melewati banyak medium dibandingkan dengan Gnutella sehingga membuat JXTA lebih kompleks dan lebih kuat performansinya. *Core* yang dimaksud memiliki beberapa protokol didalamnya, yaitu *discovery* berbasis LAN(*broadcast* melalui LAN), *discovery* melalui *invitation*(*peer* dapat

mengundang *peer* lain untuk bergabung di jaringan), *Cascade discovery*(dengan satu *peer* ditemukan akan mengarahkan pada *peer* lain melalui *proxy connection*), *discovery* melalui *rendezvous point*(seperti server yang menyimpan indeks *peer* sehingga dapat diketahui *peer* yang terhubung).

- Gnutella menggunakan metode *flooding* dalam melakukan pengiriman *query* pada setiap *peer* sehingga mengurangi performansi dari Gnutella karena *flooding* menyebabkan beban pada jaringan bertambah dan dapat menghabiskan *resource* yang disediakan oleh *peer*.

Dari pengujian yang dilakukan pada kedua aplikasi *file sharing* dapat diketahui mekanisme yang dilakukan oleh kedua aplikasi *file sharing* dalam membentuk jaringan *peer-to-peer* dan mekanisme untuk melakukan *file sharing* seperti yang ditunjukkan pada Tabel 5.6.

	JXTA	Gnutella
Arsitektur	<i>Any (Pure atau Hybrid) peer-to-peer</i>	<i>Any (Pure atau Hybrid) peer-to-peer</i>
Replikasi Data	Data didapatkan hanya dari satu <i>peer</i> sebagai <i>seeder</i> dengan nama <i>file</i> yang sama	Data didapatkan dari satu atau beberapa <i>peer</i> sebagai <i>seeder</i> dengan nama <i>file</i> yang sama
Pencarian Data	Menggunakan nama <i>file</i> sebagai kata kunci dan menghasilkan <i>file</i> dan profil <i>peer</i> pemilik <i>file</i>	Menggunakan nama <i>file</i> sebagai kata kunci dan menghasilkan <i>file</i> , jumlah <i>peer</i> yang memiliki <i>file</i> , dan alamat <i>peer-peer</i> yang memiliki <i>file</i> yang sama
Protokol Transport	TCP/IP	HTTP
Layer	Mengimplementasikan pada semua susunan pada model OSI <i>layer</i>	Didasarkan pada <i>session layer</i> dalam model OSI layer dan beroperasi pada satu <i>layer</i>
Pembaruan Informasi	Informasi setiap <i>peer</i> dan <i>index file</i> disimpan pada <i>peer</i> yang berperan sebagai <i>RDV/Relay</i> dan informasi baru akan diperbarui kepada <i>peer – peer</i> yang lain	Informasi diteruskan pada <i>peer</i> tetangga secara terus – menerus (<i>flooding</i>) melalui protokol TCP

Tabel 5.6 Perbandingan aplikasi JXTA dan Gnutella