

## BAB 6 PENGUJIAN DAN ANALISIS

Pada tahap pengujian ini akan dilakukan pengujian pada setiap kontroler, baik kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan juga kontroler ONOS. Dengan tujuan mengetahui tingkat performansi dari kontroler setiap kontroler yang akan diuji. Untuk pengujian performansi akan digunakan dua parameter. Parameter yang akan digunakan adalah parameter *throughput* dan parameter *latency*. Dimana parameter *throughput* merupakan banyaknya *flow* yang mampu direspon oleh kontroler dalam satu satuan waktu. Sedangkan parameter *latency* merupakan waktu yang dibutuhkan oleh kontroler dalam memberikan respon.

Pengujian pada setiap kontroler akan dilakukan oleh simulator Cbench yang dimana akan mengirimkan sebanyak-banyaknya paket menuju kontroler. Dengan berdasar pada parameter yang digunakan, yaitu parameter *throughput* dan parameter *latency*. Maka pengujian kelima kontroler tersebut akan dilakukan dengan empat skenario pengujian. Skenario pertama adalah pengujian *throughput* dengan jumlah *switch* dan *host* bervariasi. Pada skenario kedua adalah pengujian *throughput* dengan jumlah *switch* bervariasi. Pada skenario ketiga pengujian *latency* dengan jumlah *switch* dan *host* bervariasi. Selanjutnya pada skenario keempat pengujian *latency* dengan jumlah *switch* bervariasi.

### 6.1 Pengujian Throughput

Pada pengujian *throughput* simulator Cbench akan mensimulasikan sejumlah *switch* dan *host* yang akan mengirimkan sebanyak-banyaknya data secara acak dan ditujukan pada kontroler yang kemudian Cbench akan menghitung jumlah *flow* pada setiap detik yang dapat ditangani oleh kontroler.

#### 6.1.1 Pengujian Throughput Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch dan Host bervariasi

Dengan menerapkan skenario pengujian yang sudah dijelaskan pada bab sebelumnya, pada pengujian ini Cbench akan diberikan jumlah *switch* dan *host* yang bervariasi untuk mengetahui *throughput* kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS. Dimana dalam pengujian akan diberikan 20 *switch* hingga 120 *switch*, dengan jumlah *host* 400 hingga 12800 *host*. Dan pengujian akan dilakukan sebanyak 5 kali literasi atau perulangan dengan setiap literasi mempunyai waktu pengujian selama 1000 *ms* secara *default*. Dan perintah yang digunakan pada simulator Cbench adalah

```
$Cbench -c localHost -p 6653 -l 5 -M <jumlah Host> -s <jumlah Switch> -t
```

Untuk *port* yang digunakan pada setiap pengujian kontroler menggunakan *default port* pada masing-masing kontroler. Pada pengujian ini kontroler Floodlight menggunakan *default port* 6653, kontroler Maestro menggunakan *default port* 6633, kontroler RYU menggunakan *default port* 6633, kontroler POX menggunakan *default port* 6633, dan pada kontroler ONOS juga menggunakan *default port* 6633. Dan berikut penjelasan mengenai perintah pada Cbench.

**Tabel 6. 1** Perintah pada Cbench

-c	Alamat IP yang digunakan oleh kontroler
-p	Port yang digunakan oleh kontroler
-m	Waktu yang dibutuhkan dalam satu literasi
-l	Jumlah literasi
-M	Jumlah <i>Host</i>
-s	Jumlah <i>Switch</i>
-t	Mode Throughput

Pada skenario pengujian *throughput* ini Cbench akan berjalan pada mode *throughput* yang artinya akan menggunakan perintah *-t*. Hasil akhir yang diberikan oleh Cbench merupakan hasil minimal, maksimal, dan hasil rata-rata respon yang diberikan oleh kontroler Floodlight.

### **6.1.2 Pengujian Throughput Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah switch bervariasi**

Dengan menerapkan skenario pengujian yang sudah dijelaskan pada bab sebelumnya, Cbench akan diberikan jumlah *switch* yang bervariasi untuk mengetahui *throughput* kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan juga kontroler ONOS. Dalam pengujian ini menggunakan 20 *switch* hingga 120 *switch*, dengan jumlah *host* yang sama dalam setiap pengujian, yaitu 12800 *host*. Pengujian ini akan dilakukan sebanyak 5 kali literasi, yang dimana setiap literasi mempunyai waktu pengujian selama 1000 *ms* secara *default*.

## **6.2 Pengujian Latency**

Pada pengujian *latency* simulator Cbench akan mensimulasikan sejumlah *switch* dan *host* yang akan mengirimkan sebanyak-banyaknya data untuk menuju ke kontroler dan selanjutnya Cbench akan menghitung waktu yang dibutuhkan kontroler dalam memberikan respon terhadap data yang dikirimkan oleh simulator dalam setiap detiknya.

### **6.2.1 Pengujian Latency Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch dan Host bervariasi**

Pada pengujian *latency* ini Cbench akan diberikan jumlah *switch* dan *host* yang bervariasi untuk mengetahui *latency* pada kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS. Sama halnya dalam pengujian *throughput*, dalam pengujian ini juga akan diberikan 20 *switch* hingga 120 *switch*, dengan jumlah *host* 400 hingga 12800 *host*. Pengujian juga akan dilakukan sebanyak 5 kali literasi atau perulangan dengan setiap literasi mempunyai waktu pengujian selama 1000 *ms* secara *default*. Perintah yang digunakan pada simulator Cbench adalah:

`$Cbench -c localHost -p 6653 -l 5 -M <jumlah Host>-s <jumlah Switch>.`

Penggunaan nilai *port* pada setiap pengujian kontroler menggunakan *default port* pada masing-masing kontroler. Pada pengujian ini kontroler Floodlight menggunakan *default port* 6653, kontroler Maestro menggunakan *default port* 6633, kontroler RYU menggunakan *default port* 6633, kontroler POX menggunakan *default port* 6633, dan pada kontroler ONOS juga menggunakan *default port* 6633.

### **6.2.2 Pengujian Latency Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch bervariasi**

Pada pengujian *latency* dengan jumlah *switch* bervariasi, Cbench akan diberikan jumlah *switch* yang bervariasi untuk mengetahui *throughput* kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS. Dimana akan diberikan juga 20 *switch* hingga 120 *switch*, dengan jumlah *host* yang ditetapkan sebanyak 12800 *host*. Pengujian akan dilakukan sebanyak 5 kali literasi atau perulangan dengan setiap literasi mempunyai waktu pengujian selama 1000 *ms* secara *default*.

Dalam pengujian yang dilakukan Cbench akan menampilkan hasil dari setiap tahap dalam pengujian. Hasil yang akan ditampilkan oleh Cbench berupa jumlah minimum, jumlah maksimal, dan juga jumlah rata-rata waktu yang dibutuhkan kontroler dalam memberikan waktu respon.

## **6.3 HASIL PENGUJIAN DAN ANALISIS**

Dalam hal ini yang akan dilakukan adalah membahas serta menganalisis hasil pengujian yang telah dilakukan. Baik pengujian *throughput* dan *latency* dengan jumlah *switch* dan *host* bervariasi terhadap kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOX. Serta pengujian *throughput* dan *latency* dengan jumlah *switch* yang bervariasi terhadap kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOX, sebagaimana mengikuti skenario yang telah dibuat sebelum melakukan pengujian yang dijelaskan dalam bab implementasi dan pengujian. Hasil dari pengujian semua kontroler akan dianalisis hingga diketahui tingkat performa antara kontroler Floodlight dan kontroler Maestro.

### **6.3.1 Analisis Hasil Pengujian**

Pengujian yang telah dilakukan pada penelitian ini adalah pengujian terhadap performansi kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS dalam penanganan atau merespon paket data. Dimana pada pengujian ini digunakan dua parameter yang berbeda, yaitu parameter *throughput* dan parameter *latency*.

Pengujian yang dilakukan terbagi menjadi 20 tahap. Adapun tahap pengujian yang dilakukan sebagai berikut:

1. *Throughput* Floodlight dengan jumlah *switch* dan *host* bervariasi
2. *Troughtput* Maestro dengan jumlah *switch* dan *host* bervariasi

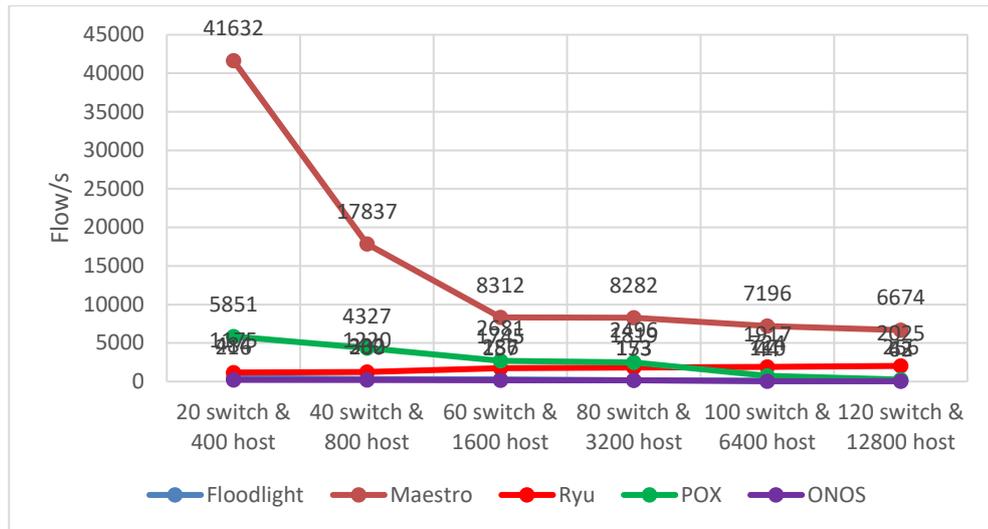
3. *Throughput* RYU dengan jumlah *switch* dan *host* bervariasi
4. *Throughput* POX dengan jumlah *switch* dan *host* bervariasi
5. *Throughput* ONOS dengan jumlah *switch* dan *host* bervariasi
6. *Throughput* Floodlight dengan jumlah *switch* bervariasi
7. *Throughput* Maestro dengan jumlah *switch* bervariasi
8. *Throughput* RYU dengan jumlah *switch* bervariasi
9. *Throughput* POX dengan jumlah *switch* bervariasi
10. *Throughput* ONOS dengan jumlah *switch* bervariasi
11. *Latency* Floodlight dengan jumlah *switch* dan *host* bervariasi
12. *Latency* Maestro dengan jumlah *switch* dan *host* bervariasi
13. *Latency* RYU dengan jumlah *switch* dan *host* bervariasi
14. *Latency* POX dengan jumlah *switch* dan *host* bervariasi
15. *Latency* ONOS dengan jumlah *switch* dan *host* bervariasi
16. *Latency* Floodlight dengan jumlah *switch* bervariasi
17. *Latency* Maestro dengan jumlah *switch* bervariasi
18. *Latency* RYU dengan jumlah *switch* bervariasi
19. *Latency* POX dengan jumlah *switch* bervariasi
20. *Latency* ONOS dengan jumlah *switch* bervariasi

### **6.3.1.1 Hasil Perbandingan *Throughput* Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch dan Host bervariasi**

Berikut ini adalah hasil dari perbandingan jumlah rata-rata *throughput* kontroler Floodlight, *throughput* kontroler Maestro, *throughput* kontroler RYU, *throughput* kontroler POX, dan *throughput* kontroler ONOS dimana pengujian ini diberikan skenario dengan jumlah *switch* dan jumlah *host* yang bervariasi. Berikut adalah tabel dan grafik hasil :

**Tabel 6. 2** Hasil perbandingan *Throughput* Floodlight, Maestro, RYU, POX dan ONOS dengan jumlah Switch dan Host bervariasi

nSwitch & nHost	Floodlight	Maestro	RYU	POX	ONOS
20 switch & 400 host	484	41632	1175	5851	216
40 switch & 800 host	332	17837	1220	4327	200
60 switch & 1600 host	286	8312	1745	2681	157
80 switch & 3200 host	173	8282	1819	2496	153
100 switch & 6400 host	110	7196	1917	724	44
120 switch & 12800 host	62	6674	2025	256	43



**Gambar 6.1** Grafik hasil perbandingan Throughput Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch dan Host bervariasi

**Sumber:** Penulis

Seperti yang terlihat pada **Tabel 6.1** dan juga pada **Gambar 6.1** dimana merupakan hasil rata-rata *flow/s* pengujian yang telah dilakukan. Dengan skenario menggunakan jumlah *switch* dan *Host* bervariasi. Nilai *switch* yang diberikan yaitu 20 *switch* hingga 120 *switch*, sedangkan nilai *host* yang diberikan yaitu 400 *Host* hingga 12800. Maka jika dilihat perbandingan kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS, nilai *Throughput* kontroler Maestro lebih besar dalam jumlah *flow* yang dapat ditangani setiap detiknya yaitu dengan nilai tertinggi 41632 *flow/s* pada penetapan 20 *switch* & 400 *host*. Meski dalam jumlah *switch* dan *host* yang lebih banyak pada kontroler Maestro mengakibatkan jumlah *flow* yang dapat ditangani kontroler Maestro menjadi menurun yaitu dengan nilai terendah 6674 *flow/s* pada penetapan 120 *switch* & 12800 *hosts*, tetapi penanganan *flow* yang diberikan kontroler Maestro setiap detiknya masih tergolong lebih besar dibanding jumlah *flow* yang dapat ditangani oleh kontroler-kontroler lainnya.

Pada kontroler Floodlight, kontroler POX, dan kontroler ONOS, jumlah *flow* yang ditangani setiap detiknya mengalami penurunan jika jumlah *host* dan jumlah *switch* ditambah. Dimana nilai tertinggi kontroler Floodlight adalah 484 *flow/s*, kontroler RYU 1175 *flow/s*, kontroler ONOS 216 *flow/s* pada penetapan 20 *switch* & 400 *host*. Berbeda dengan kontroler RYU yang memiliki peningkatan *flow* yang ditangani setiap detiknya dalam jumlah *host* dan jumlah *switch* ditambah, meski dengan jumlah *flow* yang ditangani tidak begitu besar yaitu pada penetapan 20 *switch* & 400 *host* adalah 1175 *flow/s* dan pada penetapan 120 *switch* & 12800 *hosts* jumlah *flow* yang mampu ditangani adalah 2025 *flow/s*.

Penurunan nilai *throughput* atau jumlah *flow* pada saat jumlah *switch* dan jumlah *host* ditambah berkaitan dengan *link* pada *infrastructure layer*, yang dimana *memory* yang akan terpakai pada kontroler juga akan semakin besar.

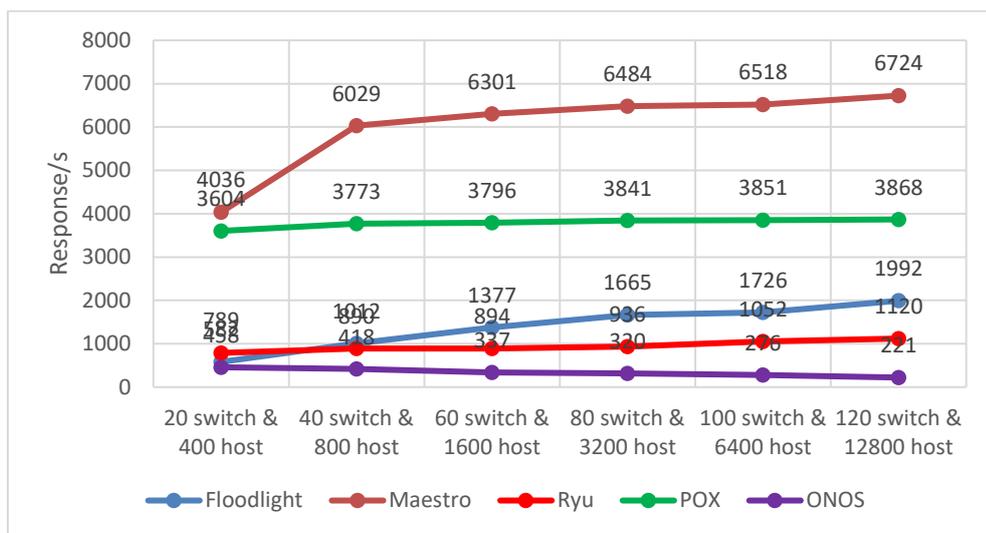
Sehingga hal tersebut dapat mempengaruhi keterlambatan kontroler dalam menangani jumlah *flow* pada setiap detik.

### 6.3.1.2 Hasil Perbandingan Latency Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch dan Host bervariasi

Berikut ini adalah hasil dari perbandingan jumlah rata-rata *Latency* kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS dimana pengujian ini diberikan skenario dengan jumlah *Switch* dan jumlah *Host* yang bervariasi. Berikut adalah tabel dan grafik hasilnya :

**Tabel 6.3** Hasil perbandingan *Latency* Floodlight dan *Latency* Maestro dengan jumlah *Switch* dan *Host* bervariasi

nSwitch & nHost	Floodlight	Maestro	RYU	POX	ONOS
20 switch & 400 host	582	4036	789	3604	458
40 switch & 800 host	1012	6029	890	3773	418
60 switch & 1600 host	1377	6301	894	3796	337
80 switch & 3200 host	1665	6484	936	3841	320
100 switch & 6400 host	1726	6518	1052	3851	276
120 switch & 12800 host	1992	6724	1120	3868	221



**Gambar 6.2** Grafik hasil perbandingan Latency Floodlight dan Latency Maestro dengan jumlah Switch dan Host bervariasi

Sumber: Penulis

Seperti yang terlihat pada **Tabel 6.3** dan juga pada **Gambar 6.2** dimana merupakan hasil rata-rata dari respon kontroler atau respon time kontroler. Dengan skenario menggunakan jumlah *Switch* dan *Host* bervariasi. Nilai *Switch* yang diberikan yaitu 20 *Switch* hingga 120 *Switch*, sedangkan nilai *Host* yang diberikan yaitu 400 *Host* hingga 12800 *host*. Maka jika dilihat perbandingan kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS, nilai respon kontroler Maestro lebih besar dalam jumlah respon yang dapat diberikan setiap detik yaitu pada penetapan 20 *switch* & 400 *host*

adalah 4036 ms, dan pada . penetapan 120 switch & 12800 host adalah 6724 ms. Meski dapat diketahui bahwa semakin banyak jumlah *host* dan jumlah *switch* diberikan pada setiap kontroler, semakin besar juga jumlah respon yang dapat diberikan oleh kontroler dalam setiap detiknya.

Namun pada kontroler ONOS jumlah respon yang dapat diberikan dalam setiap detiknya mengalami penurunan jika jumlah *host* dan jumlah *switch* semakin banyak yaitu pada penetapan 20 switch & 400 host adalah 458 ms, dan pada . penetapan 120 switch & 12800 host adalah 221 ms. Berbeda dengan kontroler Floodlight, kontroler Maestro, kontroler RYU, dan kontroler POX, dimana memiliki jumlah respon yang dapat diberikan dalam menangani *flow* lebih besar jika jumlah *host* dan jumlah *switch* semakin banyak.

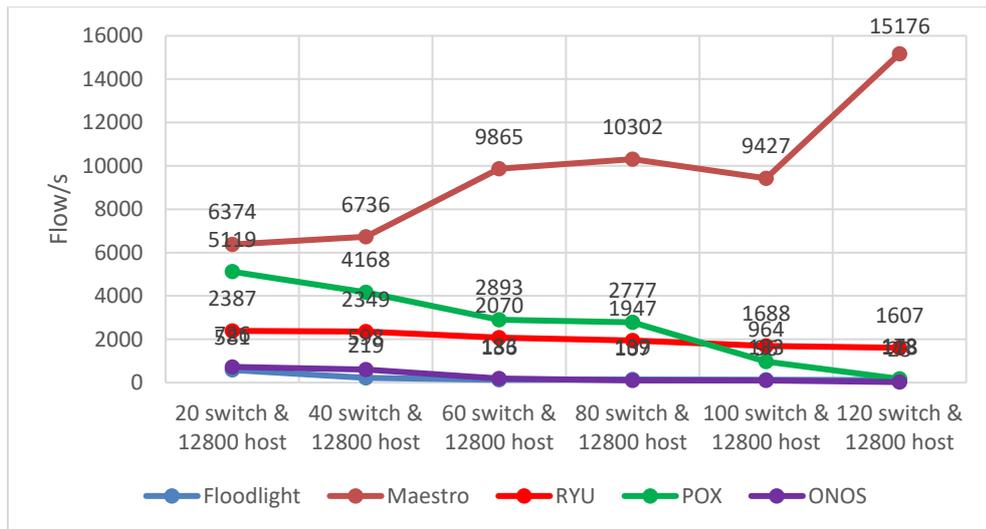
Pada dasarnya dalam pengujian jumlah *latency* atau respon yang dapat diberikan oleh kontroler dalam setiap detiknya semakin besar disaat semakin banyaknya jumlah *switch* yang diberikan. Sedangkan dalam pengujian ini kontroler ONOS memberikan performa yang berbeda dengan kontroler lainnya. Kontroler ONOX memberikan jumlah respon yang menurun ketika jumlah *switch* semakin banyak.

### 6.3.1.3 Hasil Perbandingan Throughput Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch bervariasi

Berikut ini adalah hasil dari perbandingan jumlah rata-rata *throughput* kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS, dimana pengujian ini diberikan skenario dengan jumlah *switch* yang bervariasi dan jumlah *switch* tetap. Berikut adalah tabel dan grafik hasilnya:

**Tabel 6.4** Hasil perbandingan *Throughput* Floodlight, Maestro, RYU, POX dan ONOS dengan jumlah *Switch* bervariasi

nSwitch & nHost	Floodlight	Maestro	RYU	POX	ONOS
20 switch & 12800 host	581	6374	2387	5119	726
40 switch & 12800 host	219	6736	2349	4168	598
60 switch & 12800 host	136	9865	2070	2893	183
80 switch & 12800 host	139	10302	1947	2777	107
100 switch & 12800 host	133	9427	1688	964	96
120 switch & 12800 host	118	15176	1607	173	23



**Gambar 6.3** Grafik hasil perbandingan Throughput Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch bervariasi

**Sumber:** Penulis

Seperti yang terlihat pada **Tabel 6.4** dan juga pada **Gambar 6.3** dimana merupakan hasil rata-rata *flow/s* pengujian yang telah dilakukan. Dengan skenario menggunakan jumlah *Switch* bervariasi. Nilai *Switch* yang diberikan yaitu 20 *Switch* hingga 120 *Switch*, sedangkan nilai *Host* yang diberikan yaitu 12800 *host*. Maka jika dilihat perbandingan kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS, nilai kontroler Maestro lebih besar dalam jumlah *flow* yang dapat ditangani setiap detiknya yaitu pada penetapan 20 *switch* & 400 *host* adalah 6374 *flow/s*, dan pada penetapan 120 *switch* & 12800 *host* adalah 15176 *flow/s*. Meski pada pemberian jumlah *host* 12800 dengan jumlah *switch* 100, jumlah *flow* yang ditangani kontroler Maestro mengalami penurunan dengan nilai 9427 *flow/s*, namun jumlah *flow* pada *switch* 120 dengan 12800 *switch* kontroler Maestro memiliki jumlah *flow* yang dapat ditangani meningkat kembali dengan nilai 15176 *flow/s*.

Sedangkan jumlah *flow* pada setiap detik yang dapat ditangani oleh kontroler Floodlight, kontroler RYU, kontroler POX, dan kontroler ONOX mengalami penurunan jika jumlah *switch* bertambah. Meski pada kontroler POX memiliki jumlah *flow* pada setiap detik yang dapat ditangani tergolong cukup besar, namun ketika jumlah *switch* bertambah jumlah *flow* yang dapat ditangani semakin menurun. Pada penetapan 20 *switch* & 12800 *host* kontroler Floodlight memiliki nilai 581 *flow/s*, pada kontroler RYU memiliki nilai 2387 *flow/s*, POX dengan nilai 5119 *flow/s*, dan ONOS dengan nilai 726 *flow/s*. Kemudian pada penetapan 120 *switch* & 12800 *host* nilai kontroler Floodlight menurun pada 118 *flow/s*, kontroler RYU menurun pada nilai 1607 *flow/s*, pada POX 173 *flow/s*, dan pada kontroler ONOS nilai menurun hingga 23 *flow/s*.

Penurunan dan peningkatan nilai *throughput* atau jumlah *flow* pada saat jumlah *switch* ditambah berkaitan dengan *link* pada *infrastructure layer*, yang dimana *memory* yang akan terpakai pada kontroler juga akan semakin besar. Ketika menggunakan jumlah *switch* yang jumlahnya bervariasi maka tentu *data*

traffic akan dilewati beberapa *link*, yang digunakan untuk menghubungkan *switch* dengan kontroler. Sehingga hal tersebut dapat mempengaruhi keterlambatan kontroler dalam menangani jumlah *flow* pada setiap detiknya.

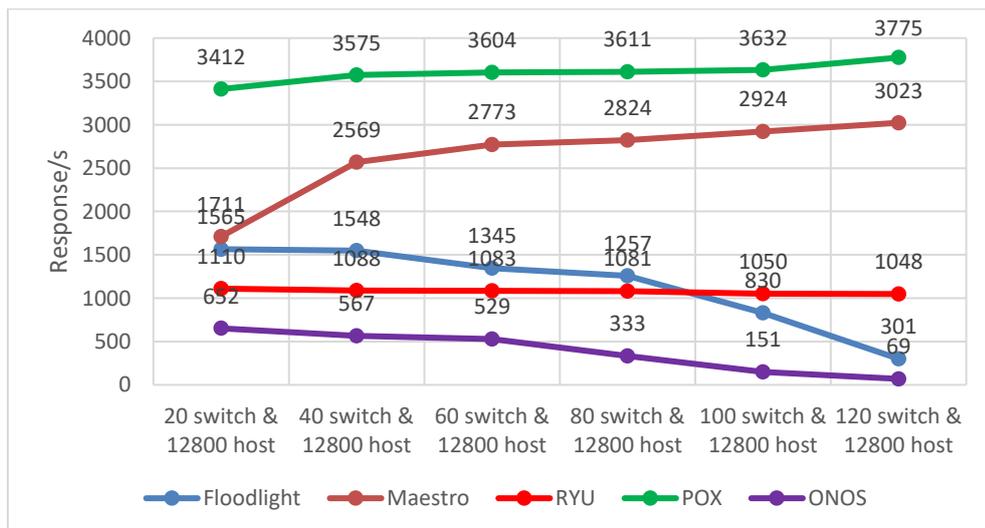
Karena pada dasarnya bertambahnya jumlah *switch* lebih dari *threshold* mengurangi total *throughput*, disebabkan karena bertambahnya jumlah *resource* atau memori untuk menangani proses input output. Sedangkan dalam pengujian ini kontroler Maestro memberikan performa yang berbeda dengan kontroler lainnya, kontroler Maestro memberikan jumlah *flow* pada setiap detik yang dapat ditangani semakin meningkat ketika jumlah *switch* yang diberikan semakin banyak.

#### 6.3.1.4 Hasil Perbandingan Latency Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch bervariasi

Berikut ini adalah hasil dari perbandingan jumlah rata-rata *latency* kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS dimana pengujian ini diberikan skenario dengan jumlah *switch* yang bervariasi. Berikut adalah tabel dan grafik hasilnya:

**Tabel 6.5** Hasil perbandingan Latency Floodlight, Maestro, RYU, POX dan ONOS dengan jumlah Switch bervariasi

nSwitch & nHost	Floodlight	Maestro	RYU	POX	ONOS
20 switch & 12800 host	1565	1711	1110	3412	652
40 switch & 12800 host	1548	2569	1088	3575	567
60 switch & 12800 host	1345	2773	1083	3604	529
80 switch & 12800 host	1257	2824	1081	3611	333
100 switch & 12800 host	830	2924	1050	3632	151
120 switch & 12800 host	301	3023	1048	3775	69



**Gambar 6.4** Grafik hasil perbandingan Latency Floodlight, Maestro, RYU, POX, dan ONOS dengan jumlah Switch bervariasi

Sumber: Penulis

Seperti yang terlihat pada **Tabel 6.5** dan juga pada **Gambar 6.4** dimana merupakan hasil rata-rata dari respon kontroler atau respon time kontroler. Dengan skenario menggunakan jumlah *switch* bervariasi. Nilai *switch* yang diberikan yaitu 20 *switch* hingga 120 *switch*, sedangkan nilai *host* yang diberikan yaitu 12800 *host*. Maka jika dilihat perbandingan kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOS, nilai kontroler POX lebih besar dalam jumlah respon yang dapat ditangani setiap detiknya. Pada jumlah 20 *switch* dengan 12800 *host* kontroler POX memiliki nilai respon yang lebih tinggi yaitu 3412 *ms*, dengan nilai 3775 *ms* pada penetapan 120 *switch* & 12800 *host*. Dibandingkan dengan kontroler Maestro 1711 *ms* pada 20 *switch* hingga 120 *switch* dengan nilai 3023 *ms* pada 120 *switch* & 12800 *host*, kontroler Floodlight 1565 *ms* pada 20 *switch* hingga 120 *switch* dengan nilai 301 *ms* pada 120 *switch* & 12800 *host*, kontroler RYU 1110 *ms* pada 20 *switch* hingga 120 *switch* dengan nilai 1048 *ms* pada 120 *switch* & 12800 *host*, dan kontroler ONOS 652 *ms* pada 20 *switch* hingga 120 *switch* dengan nilai 69 *ms* pada 120 *switch* & 12800 *host*.

Meski dapat diketahui jumlah respon yang diberikan oleh kontroler Maestro dalam setiap detiknya meningkat jika semakin banyak jumlah *switch* yang terlibat. Namun jumlah respon yang dapat diberikan oleh kontroler POX dalam setiap detiknya lebih besar dibanding dengan jumlah respon yang diberikan oleh kontroler Maestro. Berbeda dengan kontroler Floodlight, kontroler RYU, dan kontroler ONOS yang memiliki jumlah respon menurun jika jumlah *switch* yang terlibat semakin banyak. Dimana dalam pengujian jumlah *latency* semestinya dipengaruhi jumlah *switch* yang terlibat, karena semakin banyaknya jumlah *switch* jumlah respon yang dapat diberikan oleh kontroler dalam setiap detiknya juga akan semakin besar.