

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil implementasi, pengujian, dan analisis yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut:

1. Arsitektur SDN dengan agen adaptif berhasil diterapkan pada skenario topologi yang merepresentasikan kondisi jaringan FIK UPNVJ. Agen DQN dan DDPG terintegrasi secara tertanam pada Ryu Controller melalui siklus kendali tertutup 5 detik dan menjalankan fungsi perutean otomatis di lingkungan emulasi Mininet dengan protokol OpenFlow v1.3. Kedua agen dilatih 300 episode pada lingkungan, ruang keadaan, dan fungsi imbalan yang identik, dan keduanya mencapai konvergensi: DDPG stabil pada kebijakan pembagian rasio mendekati seimbang dengan *checkpoint* terbaik pada langkah ke-530, sedangkan DQN menyelesaikan peluruhan epsilon penuh dengan *reward* stabil positif. Dengan demikian, penelitian ini tidak menghasilkan arsitektur SDN baru, melainkan menerapkan arsitektur yang telah mapan pada skenario topologi kampus yang dirancang khusus, dengan agen pembelajaran mendalam sebagai modul pengambil keputusan perutean.
2. Berdasarkan evaluasi *Quality of Service*, skenario SDN-DDPG menghasilkan kinerja terbaik dengan rata-rata *packet loss* 9,59%, latensi 60,57 ms, dan jitter 0,97 ms, sehingga meraih indeks kualitas TIPHON tertinggi sebesar 3,50. Sementara itu, jaringan konvensional tanpa SDN, SDN-Dijkstra, dan SDN-DQN terpaku pada indeks 3,25 dengan *packet loss* rata-rata pada kisaran 20,18–22,51% akibat keterikatan pada jalur tunggal. Ditinjau dari ambang ITU-T Y.1541 Kelas 0, seluruh skenario memenuhi ambang *delay* dan variasi *delay*, namun tidak satu pun memenuhi ambang *packet loss* ($\leq 0,1\%$), menandakan kondisi beban puncak pada topologi ini masih berada di luar target kelas layanan *real-time* paling ketat. Uji ANOVA yang dikonfirmasi oleh Kruskal-Wallis menunjukkan perbedaan antar-skenario signifikan pada kondisi *burst* untuk latensi, *packet loss*, dan jitter ($p < 0,05$), sedangkan pada beban rendah tidak terdapat perbedaan nyata sesuai prediksi bahwa strategi perutean baru berpengaruh ketika beban mendekati kapasitas jalur.
3. Dalam Kesesuaian karakteristik ruang aksi terhadap kondisi lonjakan trafik dapat diuraikan melalui tiga karakteristik yang saling berkaitan. DQN berkarakteristik ruang aksi diskret agen hanya memilih satu jalur pada setiap keputusan, sehingga hanya satu *gateway* yang aktif pada satu waktu. DDPG berkarakteristik ruang aksi kontinu agen menghasilkan nilai rasio pembagian trafik yang diterjemahkan menjadi bobot *bucket* pada *group*

table OpenFlow, sehingga kedua *gateway* dapat aktif bersamaan. Adapun topologi FIK berkarakteristik dua jalur *gateway* redundan berkapasitas 30 Mbps per jalur, dengan beban *burst* mencapai ± 57 Mbps melampaui kapasitas satu jalur, namun masih tertampung oleh kapasitas gabungan keduanya. Dari ketiga karakteristik tersebut, ruang aksi kontinu paling sesuai dengan topologi FIK: ketika permintaan melampaui kapasitas satu jalur namun masih tertampung kapasitas gabungan, hanya agen yang mampu membagi trafik secara proporsional yang dapat memanfaatkannya secara utuh. Hal ini terkonfirmasi oleh data pengujian pada kondisi *burst*, DDPG mencatat *packet loss* 28,08% dengan throughput agregat sisi *controller* menembus 49,7 Mbps, sementara DQN tertahan pada 59,67% dan 27,4 Mbps, praktis setara dengan pendekatan statis. Ketidaksesuaian DQN bukan menunjukkan kegagalan algoritma, melainkan bentuk keputusannya yang tidak menjawab persoalan yang menuntut pemanfaatan dua jalur sekaligus; karakteristik tersebut justru sesuai untuk kelas persoalan lain seperti *failover*, sebagaimana Dijkstra tetap sesuai untuk kondisi beban ringan dan stabil. Dengan demikian, kesesuaian algoritma bersifat kontekstual: bukan penggunaan pendekatan pembelajaran mendalam itu sendiri yang menentukan, melainkan kesesuaian antara karakteristik ruang aksi dengan karakteristik topologi dan pola beban yang dihadapi.

5.2 Saran

Untuk pengembangan penelitian selanjutnya, disarankan beberapa hal sebagai berikut:

1. Melakukan implementasi pada perangkat keras fisik atau *testbed* nyata (misalnya *switch* yang mendukung OpenFlow) untuk memvalidasi hasil simulasi terhadap kondisi jaringan produksi di lingkungan kampus.
2. Menguji algoritma DRL yang lebih mutakhir seperti TD3, PPO, atau SAC sebagai pembanding terhadap DQN dan DDPG khususnya untuk mengatasi *overestimation bias* pada estimasi *Q-value* sekaligus mengevaluasi *overhead* kontrol (beban komunikasi *controller-switch* dan waktu inferensi agen) sebagai bagian dari penilaian kinerja secara menyeluruh.
3. Memperluas skala topologi dan variasi pola trafik (misalnya trafik campuran *video conference*, LMS, dan unduhan masif sesuai karakteristik jaringan FIK) agar model teruji pada kondisi yang lebih realistis.
4. Melakukan pengujian pada skenario kegagalan tautan (*failover*) untuk menguji apakah ruang aksi diskret lebih sesuai pada kelas persoalan

pemilihan jalur tunggal, sehingga peta kesesuaian antar-algoritma menjadi lebih lengkap.

5. Memperbaiki instrumentasi pengukuran *packet loss* pada sisi controller misalnya melalui pembacaan statistik antrean agar seluruh komponen fungsi imbalan berkontribusi efektif terhadap proses pembelajaran agen, sekaligus menambah jumlah pengulangan pengujian untuk memperkuat daya uji statistik.
6. Menambahkan mekanisme pengawasan (*human-in-the-loop*) dan prioritas trafik pada agen sesuai masukan pengelola jaringan, agar keputusan agen tetap terkontrol terutama untuk layanan kritikal seperti *video conference*.