ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika

ISSN(p): 2338-8323 | ISSN(e): 2459-9638 | Vol. 9 | No. 3 | Halaman 716 - 730
DOI : http://dx.doi.org/10.26760/elkomika.v9i3.716 Juli 2021

Algoritma *Epsilon Greedy* pada *Reinforcement Learning* untuk Modulasi Adaptif Komunikasi *Vehicle to Infrastructure* (V2I)

NAZMIA KURNIAWATI[1], YULI KURNIA NINGSIH[2], SOFIA DEBI PUSPA[3], TRI SWASONO ADI[4]

[1,2,4]Jurusan Teknik Elektro, Universitas Trisakti, Indonesia
[3]Jurusan Teknik Mesin, Universitas Trisakti, Indonesia
Email : nazmia.kurniawati@trisakti.ac.id

Received 5 April 2021 **|** *Revised* 24 April 2021 **|** *Accepted* 3 Mei 2021

ABSTRAK

Komunikasi Vehicle to Infrastructure (V2I) memungkinkan kendaraan dapat terhubung ke berbagai macam infrastruktur. Dengan kondisi kendaraan yang bergerak, maka kondisi lingkungan yang dilewati mempengaruhi parameter komunikasi. Implementasi modulasi adaptif pada skema V2I memperbolehkan sistem menggunakan skema modulasi yang berbeda untuk mengakomodasi perubahan kondisi lingkungan. Pada penelitian ini digunakan skema modulasi QPSK, 8PSK, dan 16-QAM dengan memanfaatkan reinforcement learning dan algoritma epsilon greedy untuk menentukan skema modulasi yang digunakan berdasarkan level AWGN. Dari hasil simulasi dengan kondisi nilai epsilon yang divariasikan dari 0.1 hingga 0.5 didapatkan bahwa semakin tinggi nilai epsilon maka semakin sering agen tidak memilih skema modulasi dengan reward tertinggi.

***Kata kunci**: Reinforcement learning, Modulasi Adaptif, AWGN*

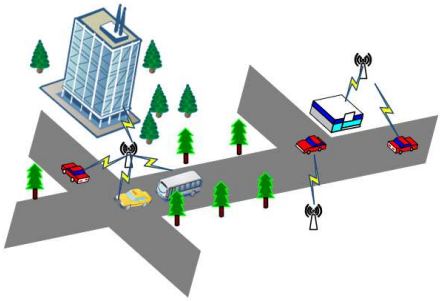
ABSTRACT

Vehicle to Infrastructure (V2I) communication allows vehicles to be connected to various infrastructures. Under the scenario of a moving vehicle, the environmental conditions which is passed by the vehicle will affect the communication parameters. The adaptive modulation implementation in the V2I scheme allows the system to use different modulation schemes to accommodate changing environmental conditions. In this study, the QPSK, 8PSK, and 16-QAM modulation schemes were used by utilizing reinforcement learning and the epsilon greedy algorithm to determine the modulation scheme used based on AWGN level. From the simulation results with the conditions of the epsilon value varying from 0.1 to 0.5, it is found that the higher the epsilon value, the more often the agent does not choose the modulation scheme with the highest reward.

***Keywords**: Reinforcement learning, Adaptive Modulation, AWGN*

1. PENDAHULUAN

Sistem transportasi cerdas merupakan bagian penting dalam pengembangan sebuah kota cerdas (*smart city*) **(Halegoua, 2020).** Pada sistem transportasi cerdas, kendaraan dapat berkomunikasi dengan berbagai infrastruktur atau kendaraan lain. *Vehicle-to-Infrastructure* (V2I) *communication* merupakan salah satu bagian dari sistem komunikasi pada sistem transportasi cerdas. Pada V2I, kendaraan dapat berkomunikasi dengan infrastruktur di sekelilingnya seperti lampu lalu-lintas, kamera pengawas, atau infrastruktur pada suatu bangunan **(Wietfeld & Ide, 2015).** Dengan memanfaatkan kanal *wireless*, informasi dikirimkan melalui jaringan *ad-hoc* sehingga proses pertukaran informasi dapat dilakukan secara fleksibel. Melalui sistem ini, kendaraan dan infrastruktur dapat saling bertukar informasi mengenai *time of arrival* kendaraan, informasi ketersediaan parkir pada gedung, iklan suatu produk, informasi navigasi dan lalu lintas **(Cronin, 2015).** Dengan memanfaatkan teknologi ini, diharapkan jumlah kecelakaan yang diakibatkan oleh kelalaian manusia dapat berkurang hingga 80% **(Thomas, 2016).**



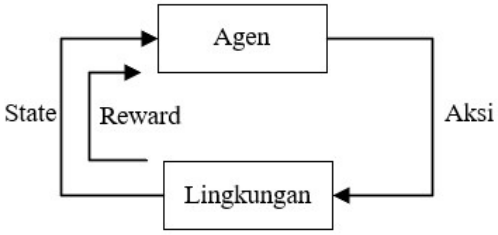
Gambar 1. Skema V2I

Dengan kondisi kendaraan yang bergerak, maka kondisi lingkungan yang dilewati pun akan berubah. Hal tersebut berpengaruh terhadap level *noise* yang diterima **(Skrucany, dkk, 2017)**. *Noise* yang tinggi akan menyebabkan kemungkinan diterimanya *bit error* oleh *receiver* semakin tinggi **(Pandey, dkk, 2013).** Hal tersebut menyebabkan semakin tingginya nilai *Signal to Noise Ratio* (SNR) yang dibutuhkan untuk menjaga nilai *error* tetap berada pada level yang diizinkan. Salah satu strategi untuk mengakomodasi hal tersebut adalah dengan menggunakan skema modulasi yang berbeda mengikuti kondisi *noise* yang lingkungan **(Eska, 2018).** Semakin tinggi skema modulasi maka semakin tinggi nilai SNR yang dibutuhkan **(Dangi & Porwal, 2015).** Hal tersebut berakibat pada semakin rendahnya level *noise* yang diizinkan. Namun keuntungan dari skema tersebut adalah semakin banyak informasi yang dapat dibawa oleh satu simbol **(Singya, dkk, 2021)**.

Modulasi adaptif merupakan sebuah teknologi yang memungkinkan sistem untuk memilih skema modulasi yang paling baik sesuai dengan kondisi kanal **(Masood, 2013)**. Saat kanal dalam kondisi bagus atau nilai *noise* rendah maka modulasi dengan *data rate* lebih tinggi dapat digunakan. Sedangkan saat kondisi kanal sedang buruk maka modulasi dengan *data rate* lebih rendah digunakan untuk menghindari terjadinya *packet drop* yang tinggi. Dengan mengimplementasikan skema modulasi adaptif, maka penggunaan *bandwidth* dapat dioptimalkan dan level sensitifitas terhadap perubahan lingkungan dapat diturunkan **(Novfitri, dkk, 2018)**. Dengan demikian kualitas koneksi dapat dijaga walaupun kondisi lingkungan yang dilewati berubah-ubah.

Reinforcement learning merupakan bagian dari *machine learning* di mana *learning process* terjadi ketika agen berinteraksi dengan lingkungan **(Ravinchandiran, 2018)**. Tidak seperti

machine learning yang membutuhkan *dataset* untuk proses *training*, pada *reinforcement learning* agen akan mengeksplorasi lingkungannya dan membuat keputusan berdasarkan nilai *reward* dan *punishment* yang diberikan ketika agen melakukan suatu aksi **(Lowe & Ziemke, 2013).** Setelah proses *trial* dan *error*, agen akan mempelajari aksi apa yang harus dilakukan untuk mendapatkan *reward* dengan nilai tertinggi **(Sutton & Barto, 2015).** Sehingga agen akan memiliki kecenderungan untuk mengambil aksi tersebut jika berada pada suatu kondisi tertentu.



Gambar 2. *Reinforcement Learning* Loop

Epsilon greedy merupakan salah satu algoritma yang digunakan untuk *learning process.* Algoritma ini menyeimbangkan proses eksploitasi dan eksplorasi berdasarkan nilai *epsilon* **(dos Santos Mignon & de Azevedo da Rocha, 2017)**. Ketika agen melakukan eksplorasi lingkungan, agen akan mengambil aksi acak tanpa mempedulikan nilai *reward* atau *punishment* yang didapatkan. Sedangkan ketika agen berada dalam mode eksploitasi, maka agen akan melakukan aksi yang memberikan nilai *reward* tertinggi. *Epsilon* dalam hal ini merupakan parameter yang menentukan apakah agen akan melakukan eksplorasi atau eksploitasi **(Nieuwdorp, 2017)**.

Pada penelitian ini dilakukan implementasi algoritma *epsilon-greedy* pada *reinforcement learning* sebagai pembuat keputusan untuk pemilihan skema modulasi pada V2I. Pada skema ini, infrastruktur bertindak sebagai pengirim (*Transmitter*/Tx) yang memiliki kemampuan untuk mengubah skema modulasi. Sementara itu kendaraan yang bergerak merupakan penerima yang juga difungsikan sebagai agen. Model kanal *Additive White Gaussian Noise* (AWGN) digunakan untuk menentukan kondisi lingkungan yang dilewati kendaraan. Pada **(Rochmatika, dkk, 2018)** diusulkan *Binary Phase Shift Keying* (BPSK), *Quadrature Phase Shift Keying* (QPSK), dan 16-*Quadrature Amplitude Modulation* (16-QAM) sebagai modulasi untuk skema modulasi adaptif. Namun pada model kanal AWGN probabilitas *error* untuk BPSK dan QPSK tidak memiliki perbedaan pada perhitungannya. Sehingga penggunaan kedua modulasi tersebut tidak dapat digunakan untuk membedakan kondisi *noise* pada lingkungan. Oleh karena itu pada penelitian ini diusulkan penggunaan QPSK, 8PSK, dan 16-QAM sebagai skema modulasi untuk membedakan kondisi *noise* pada lingkungan.

Struktur *paper* ini adalah sebagai berikut, pada bagian pendahuluan dipaparkan ide dasar mengenai *paper* ini. Di bagian kedua dijelaskan mengenai pengimplementasian algoritma *epsilon-greedy* untuk skema modulasi adaptif pada V2I. Di bagian selanjutnya dijelaskan mengenai hasil yang didapat. Di bagian terakhir diambil kesimpulan mengenai penelitian ini.

2. METODE

Berdasarkan **(Sassi, dkk, 2012)**, nilai maksimum *bit error* yang diperbolehkan untuk sistem komunikasi pada kendaraan adalah 10^{-3}. Dengan menggunakan model kanal AWGN dan nilai 10^{-3} sebagai *threshold* probabilitas *error*, nilai Eb/No dikalkulasi sebagai dasar untuk mengkategorikan level *noise* menjadi *low, medium,* dan *high*. QPSK, 8PSK, dan 16-QAM

digunakan untuk skema modulasi adaptif dengan level *noise* lingkungan sebagai penentunya. Skema modulasi yang lebih tinggi akan dimaksimalkan untuk kondisi lingkungan dengan level *noise* yang rendah. Sedangkan skema modulasi dengan *data rate* yang lebih rendah digunakan untuk lingkungan dengan level *noise* yang tinggi.

Setelah skema modulasi ditentukan, selanjutnya dilakukan implementasi *epsilon-greedy* pada *reinforcement learning*. Untuk mengasumsikan kondisi kendaraan bergerak melewati lingkungan yang memiliki level *noise* yang berbeda, simulasi dilakukan secara episodik dengan jumlah episode sebanyak 1000 kali. Hal tersebut digunakan untuk menggambarkan kendaraan yang bergerak melewati 1000 lingkungan yang berbeda-beda level *noise*-nya. Berdasarkan **(Novfitri, dkk, 2018)**, rekomendasi kecepatan kendaraan untuk simulasi adalah 44.43 hingga 88.8 meter per jam.

2.1 Menentukan Nilai Eb/No

Parameter *Energy per Bit to Spectral Noise Density* (Eb/No) digunakan untuk membedakan kondisi setiap lingkungan. Untuk menghitung nilai Eb/No, Persamaan (1) digunakan untuk mengubah fungsi Q menjadi fungsi *error* (erfc) **(Bliss & Govindasamy, 2013).**

$$Q(x) = \frac{1}{2} erfc\left(\frac{x}{\sqrt{2}}\right) \tag{1}$$

Persamaan (2) digunakan untuk menghitung probabilitas *error* pada modulasi QPSK **(Ippolito Jr., 2017)**.

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_o}}\right) \tag{2}$$

Dengan memodifikasi Persamaan (2) dengan Persamaan (1), maka didapatkan persamaan untuk menghitung probabilitas *error* pada modulasi QPSK adalah:

$$P_b = \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{N_0}}\right) \tag{3}$$

Persamaan (4) digunakan untuk menghitung probabilitas *error* pada 8PSK dengan fungsi Q **(Sighal, dkk, 2013)**.

$$P_b = \frac{1}{log_2 M} 2Q\left(\sqrt{2\frac{E_b}{N_0} log_2 M \;\; sin\frac{\pi}{M}}\right) \tag{4}$$

Dengan memodifikasi Persamaan (4) dengan Persamaan (1), maka persamaan untuk menghitung probabilitas *error* 8PSK menjadi:

$$P_b = \frac{1}{3} erfc\left(\sqrt{3\frac{E_b}{N_0} \;\; sin\left(\frac{180}{8}\right)^o}\right) \tag{5}$$

Berdasarkan **(Oyetola, dkk, 2018)**, untuk menghitung probabilitas *error* 16-QAM, digunakan Persamaan (6).

$$P_b = \frac{4}{log_2 M}\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3log_2 M}{M-1}\frac{E_b}{N_0}}\right) \quad (6)$$

Dengan mengubah fungsi Q pada Persamaan (6) menjadi fungsi *error* pada Persamaan (1), maka persamaan untuk menghitung probabilitas *error* 16-QAM menjadi :

$$P_b = \frac{3}{8} erfc\left(\sqrt{\frac{2}{5}\frac{E_b}{N_0}}\right) \quad (7)$$

Dengan menggunakan Persamaan (3), (5), dan (7) dengan nilai Pb = 10^{-3}, maka didapatkan nilai *threshold* Eb/No untuk setiap modulasi adalah: 6.78 dB untuk QPSK, 10 dB untuk 8PSK, dan 10.52 dB untuk 16-QAM. Tabel 1 merangkum nilai Eb/No untuk setiap modulasi dalam satuan desibel (dB)

Tabel 1. Nilai *Threshold* Eb/No Setiap Modulasi

Modulasi	E_b/N_o (dB)
QPSK	6.78
8PSK	10
16-QAM	10.52

Berdasarkan hasil perhitungan, semakin tinggi skema modulasi maka semakin tinggi nilai Eb/No yang dibutuhkan untuk menjaga probabilitas *error* tidak melebihi 10^{-3}. Dengan asumsi nilai Eb konstan maka dibutuhkan level *noise* yang lebih rendah agar probabilitas *error* terjaga pada nilai 10^{-3}.

2.2 Menentukan *Threshold* untuk Setiap Lingkungan

Dengan menggunakan nilai pada Tabel 1 sebagai acuan *threshold* Eb/No, kondisi lingkungan dikategorikan menjadi tiga kondisi. Ketika nilai Eb/No kurang dari 6.78 dB, lingkungan diasumsikan berada pada kondisi *high noise*. Sementara itu saat Eb/No berada di antara 6.78 dan 10 dB kondisi lingkungan diasumsikan berada pada kondisi *medium noise*. Sedangkan saat Eb/No berada pada nilai melebihi 10 dB diasumsikan lingkungan berada pada kondisi *low noise*. Dengan demikian nilai Eb/No untuk setiap kondisi lingkungan *noise* dapat dirangkum seperti yang ditunjukkan pada Tabel 2.

Tabel 2. Nilai Eb/No Setiap Lingkungan

Noise	**Nilai (dB)**
High	Eb/No < 6.78
Medium	6.78 ≤ Eb/No ≤ 10
Low	Eb/No > 10

2.3 Menentukan Skema Modulasi Setiap Lingkungan

Setelah mendefiniskan nilai Eb/No untuk setiap kondisi lingkungan, langkah selanjutnya adalah menentukan skema modulasi untuk setiap lingkungan dengan menggunakan Tabel 1 dan 2 sebagai acuan.

Saat Eb/No berada pada nilai kurang dari 6.78 dB, *threshold* Eb/No hanya terpenuhi untuk modulasi QPSK. Oleh karena itu pada kondisi lingkungan *high noise* hanya skema modulasi QPSK yang diizinkan untuk dipergunakan. Ketika Eb/No berada pada antara 6.78 dan 10 dB, nilai tersebut memenuhi *threshold* QPSK dan 8PSK. Oleh karena itu ketika level *noise* berada pada kondisi *medium noise*, skema modulasi yang diperbolehkan untuk digunakan adalah 8PSK dan 16-QAM. Sedangkan saat Eb/No berada di atas 10 dB maka *threshold* untuk skema modulasi QPSK, 8PSK, dan 16-QAM terpenuhi. Oleh karena itu saat *noise* level berada pada kondisi *low noise*, ketiga skema modulasi diperbolehkan untuk digunakan. Tabel 3 menunjukkan *mapping* skema modulasi yang diperbolehkan untuk setiap kondisi lingkungan.

Tabel 3. *Mapping* Skema Modulasi Setiap Lingkungan

		Skema Modulasi		
		16-QAM	**8PSK**	**QPSK**
Noise	***High***	X	X	V
	Medium	X	V	V
	Low	V	V	V

Keterangan:
X : tidak diperbolehkan
V : diperbolehkan

Pada kondisi *low noise* dan *medium noise*, ada lebih dari satu skema modulasi yang diperbolehkan untuk digunakan. Oleh karena itu prioritas diberikan untuk mengontrol pemilihan skema modulasi. Skema modulasi dengan *data rate* yang lebih tinggi diberikan nilai prioritas yang lebih tinggi dibandingkan dengan skema modulasi dengan *data rate* yang lebih rendah. Sehingga tanda "V" pada Tabel 3 diubah menjadi nilai prioritas di mana semakin kecil angkanya maka semakin tinggi prioritas penggunaan.

Tabel 4. *Mapping* Prioritas Skema Modulasi

		Skema Modulasi		
		16-QAM	**8PSK**	**QPSK**
Noise	***High***	X	X	1
	Medium	X	1	2
	Low	1	2	3

Keterangan:
X : tidak diperbolehkan
1 : prioritas pertama/tertinggi
2 : prioritas kedua
3 : prioritas ketiga/terendah

Pada kondisi *low noise*, penggunaan skema modulasi 16-QAM lebih diprioritaskan dibandingkan penggunaan skema modulasi 8PSK dan penggunaan skema modulasi QPSK merupakan prioritas terakhir. Hal tersebut disebabkan 16-QAM memiliki *data rate* yang paling tinggi dibanding dua skema modulasi lainnya. Hal tersebut sejalan dengan kondisi lingkungan pada *medium noise*. 8PSK diberikan prioritas yang lebih tinggi karena memiliki *data rate* yang lebih tinggi dibandingkan QPSK.

2.4 Implementasi *Reinforcement Learning*

Reinforcement learning melakukan proses *learning* berdasarkan prinsip *reward* dan *punishment*. *Reward* diberikan jika agen mengambil aksi yang benar dan *punishment* jika aksi yang diambil salah. Dengan menggunakan Tabel 4 sebagai acuan, nilai *reward* and *punishment* ditentukan.

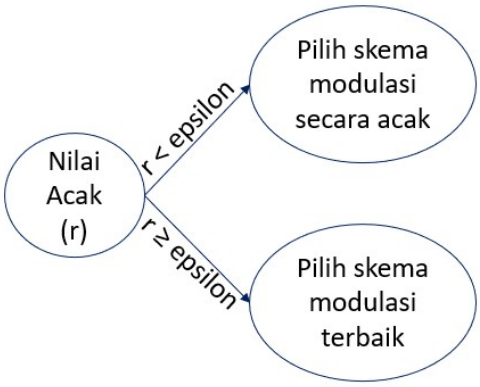
Pada kondisi *high noise*, skema modulasi yang boleh dipergunakan hanya QPSK. Maka jika agen mengambil aksi untuk menggunakan skema modulasi yang lain, maka agen akan diberikan *punishment*. Hal serupa juga dilakukan pada kondisi *medium noise*. *Punishment* akan diberikan kepada agen jika skema modulasi 16-QAM dipilih. Pada kondisi *noise* yang sama, 8PSK memiliki prioritas lebih tinggi dibandingkan QPSK. Oleh karena itu *reward* untuk agen jika memilih 8PSK akan lebih besar dibandingkan dengan QPSK. Pada kondisi *low noise*, *reward* yang paling besar akan diberikan jika agen memilih skema modulasi 16-QAM sedangkan jika QPSK yang dipilih maka agen akan diberikan *reward* dengan nilai yang paling kecil. Tabel 5 menampilkan nilai *reward* dan *punishment* untuk setiap skema modulasi di masing-masing kondisi *noise*.

Tabel 5. Nilai *Reward* Dan *Punishment*

		Skema Modulasi		
		16-QAM	**8PSK**	**QPSK**
Noise	***High***	0	0	0.9
	Medium	0	0.9	0.3
	Low	0.9	0.3	0.1

Nilai *punishment* yang diberikan adalah 0. Sehingga ketika agen memilih skema modulasi 16-QAM atau 8PSK pada kondisi lingkungan dengan *high noise* atau 16-QAM pada kondisi *medium noise* maka agen akan diberikan nilai 0 untuk aksi tersebut. Sedangkan jika agen memilih skema modulasi 8PSK saat kondisi *medium noise* atau 16-QAM pada kondisi *low noise* maka agen akan mendapatkan *reward* yang tinggi. Dengan hal tersebut maka agen akan belajar aksi apa yang dapat memberikan nilai terbaik saat berada di suatu kondisi. Sehingga agen memiliki kecenderungan untuk mengambil aksi terbaik setiap kali berada di kondisi tertentu.

Pada algoritma *epsilon greedy*, agen berada pada mode eksplorasi atau eksploitasi ditentukan oleh nilai *epsilon*. Untuk mengontrol agen agar dapat memasuki mode ekplorasi atau eksploitasi, sebuah nilai acak akan dibangkitkan. Jika nilai acak lebih besar dari *epsilon*, maka agen akan berada pada mode eksploitasi. Pada mode ini agen akan mengambil aksi yang memiliki nilai *reward* tertinggi. Sedangkan jika nilai acak yang dibangkitkan lebih rendah dari *epsilon*, maka agen akan berada pada mode eksplorasi. Pada mode ini agen akan mengambil aksi secara acak. Gambar 3 mengilustrasikan cara kerja algoritma *epsilon greedy*.

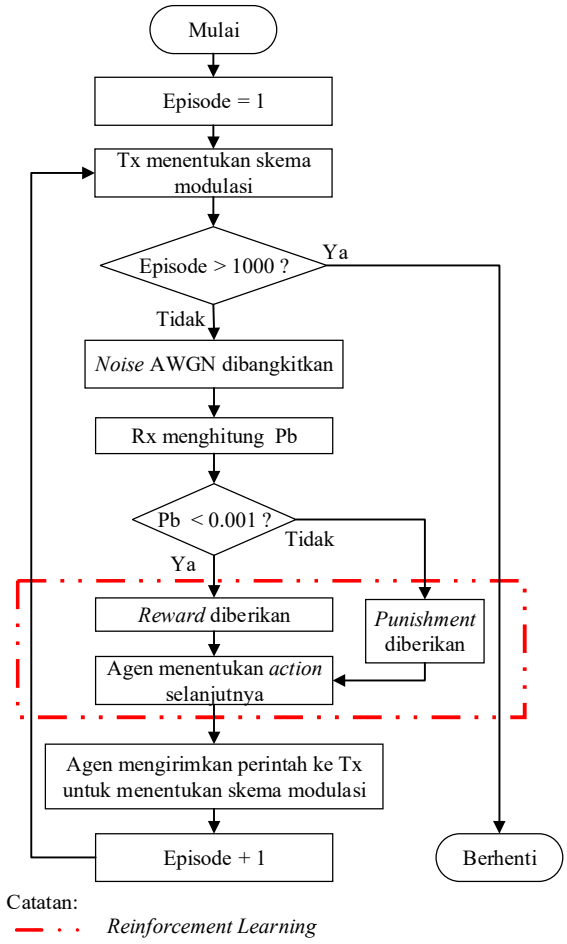


Gambar 3. Cara Kerja Algoritma *Epsilon Greedy*

Nilai acak (r) akan dibangkitkan dengan nilai antara 0 dan 1. Angka ini kemudian dibandingkan dengan nilai *epsilon*. Jika r bernilai lebih kecil dari *epsilon* maka agen akan memilih skema modulasi secara acak apapun kondisi lingkungannya. Agen bisa mendapatkan *reward* atau

punishment tergantung skema modulasi yang dipilihnya. Sementara itu ketika r yang dibangkitkan bernilai lebih besar dari *epsilon* maka agen akan memilih skema modulasi terbaik berdasarkan nilai pada Tabel 5. Untuk mengetahui pengaruh *epsilon* terhadap hasil pengambilan keputusan maka pada simulasi akan digunakan nilai *epsilon* yang berbeda-beda. Nilai *epsilon* yang digunakan pada simulasi adalah 0.1, 0.2, 0.3, 0.4, dan 0.5.

Berikut ini adalah alur simulasi yang dibuat. Untuk menyimulasikan kendaraan yang bergerak melalui berbagai kondisi lingkungan maka algoritma dibuat dengan situasi episodik. Setiap satu episode, kendaraan melalui satu kondisi lingkungan. Program di-*loop* sebanyak seribu kali untuk menyimulasikan kendaraan yang bergerak melewati seribu kondisi lingkungan dalam sekali perjalanan. *Flowchart* di bawah ini menunjukkan algoritma yang dibuat.



Gambar 4. Alur Simulasi

Pada episode 1, pengirim atau Tx secara acak menentukan skema modulasi yang digunakan. Kemudian program melakukan perhitungan jumlah episode. Jika episode lebih dari seribu, maka simulasi akan dihentikan. Jika episode masih bernilai kurang dari seribu, maka simulasi akan dilanjutnya ke tahap selanjutnya. *Noise* AWGN dibangkitkan untuk mensimulasikan kondisi lingkungan yang dilewati oleh kendaraan dengan nilai yang termasuk ke dalam level *low, medium*, atau *high noise*. Kendaraan sebagai penerima kemudian menghitung nilai Pb yang diterima. Jika probabilitas *error* lebih dari 10^{-3}, maka kendaraan yang juga bertindak sebagai agen akan diberikan *punishment*. Sedangkan jika nilai probabilitas *error* kurang dari

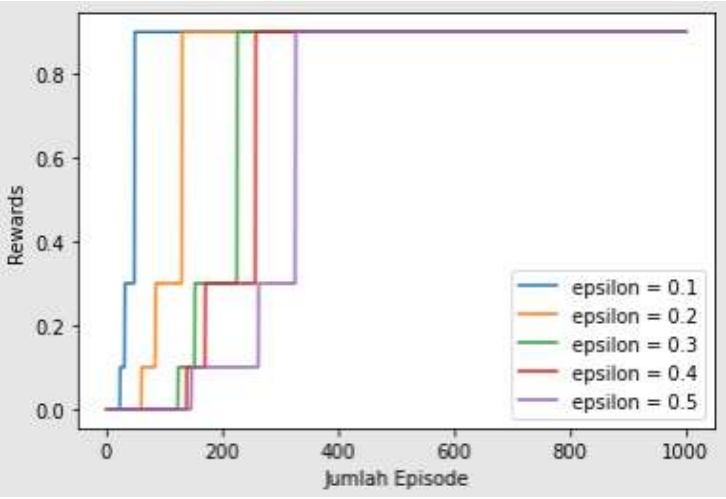
10^{-3} maka *reward* akan diberikan dengan nilai seperti yang ditunjukkan oleh Tabel 5. Selanjutnya agen dengan *reinforcement learning* akan menentukan skema modulasi apa yang seharusnya digunakan saat berada pada kondisi tersebut. Selanjutnya agen mengirimkan perintah ke Tx untuk menggunakan skema modulasi seperti yang diperintahkan oleh agen. Hal tersebut dilakukan secara terus menerus hingga episode berjumlah seribu. Setiap seribu *loop*, nilai *epsilon* diubah mulai dari 0.1 hingga 0.5. Setelah simulasi selesai, selanjutnya dilakukan analisis terhadap hasil yang didapatkan.

3. HASIL DAN PEMBAHASAN

Pada bagian ini akan dibahas hasil simulasi dengan nilai *epsilon* yang bervariasi dari 0.1 hingga 0.5. Hasil simulasi yang ditampilkan menggunakan empat skenario. Skenario pertama kendaraan bergerak melewati lingkungan *low, medium*, dan *high noise*. Pada skenario kedua kendaraan melewati lingkungan *low noise*. Skenario selanjutnya, kendaraan melewati lingkungan *medium noise*. Pada skenario terakhir, kendaraan melewati lingkungan *high noise.*

3.1 Hasil Simulasi Seluruh Kondisi *Noise*

Gambar 5 menunjukkan hasil simulasi untuk seluruh kondisi *noise*. Dari grafik dapat dilihat bahwa agen mendapatkan *punishment* dengan nilai 0 dan *reward* yang bervariasi, antara 0.1, 0.3, dan 0.9.



Gambar 5. Hasil Simulasi Seluruh Level *Noise*

Saat *epsilon* bernilai 0.1, agen mendapatkan nilai 0 sebanyak 23 kali, *reward* 0.1 sebanyak 8 kali, *reward* 0.3 sebanyak 17 kali, dan *reward* 0.9 sebanyak 952 kali. Ketika *epsilon* bernilai 0.2 agen mendapatkan *punishment* 0 sebanyak 60 kali, *reward* 0.1 sebanyak 24 kali, *reward* 0.3 sebanyak 46 kali, dan *reward* 0.9 sebanyak 870 kali. Sementara itu saat *epsilon* bernilai 0.3, *punishment* bernilai 0 didapatkan agen sebanyak 123 kali, *reward* 0.1 sebanyak 29 kali, *reward* 0.3 sebanyak 73 kali, dan *reward* 0.9 didapatkan sebanyak 775 kali. Sedangkan saat *epsilon* bernilai 0.4, agen mendapatkan *punishment* sebanyak 138 kali, *reward* 0.1 sebanyak 32 kali, *reward* 0.3 sebanyak 87 kali, dan *reward* 0.9 sebanyak 743 kali. Tabel 6 merangkum jumlah *reward* dan *punishment* yang didapatkan oleh agen.

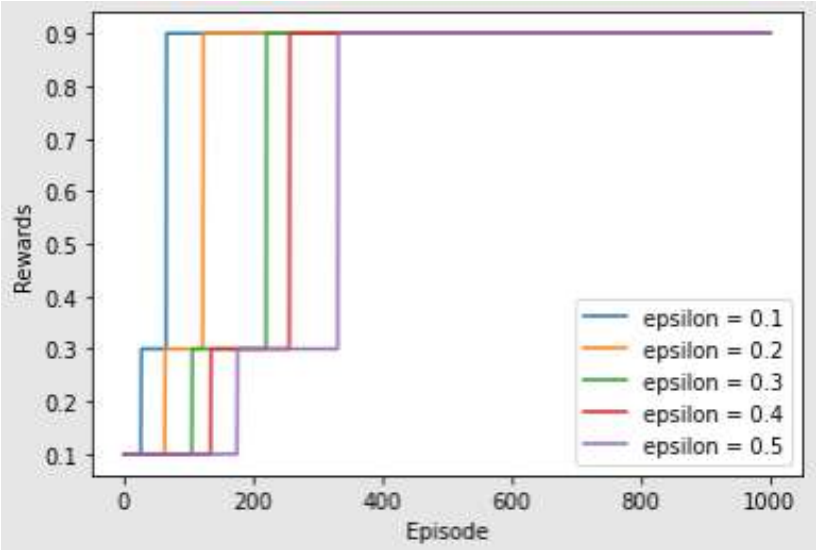
Tabel 6. Jumlah *Reward* Dan *Punishment* yang Didapatkan Agen

		***Rewards* (jumlah episode)**			
		0.9	**0.3**	**0.1**	**0**
Epsilon	**0.1**	952	17	8	23
	0.2	870	46	24	60
	0.3	775	73	29	123
	0.4	743	87	32	138
	0.5	674	64	116	146

Dari hasil simulasi didapatkan semakin tinggi nilai *epsilon* maka jumlah *punishment* yang didapatkan agen semakin banyak dan *reward* tertinggi semakin sedikit. Hal tersebut disebabkan oleh semakin besarnya batas antara mode eksplorasi dan eksploitasi. Ketika nilai *epsilon* semakin besar maka agen akan semakin sering berada pada mode eksplorasi. Sehingga semakin sering agen memilih skema modulasi secara acak tanpa mempedulikan kondisi lingkungan.

3.2 Hasil Simulasi Lingkungan *Low Noise*

Gambar 6 menunjukkan hasil simulasi pada kondisi *low noise*. Pada simulasi ini agen melewati 1000 tempat dengan nilai *noise* yang berbeda namun termasuk ke dalam kondisi lingkungan *low noise*. Berdasarkan Tabel 3, pada kondisi *low noise* agen diperbolehkan menggunakan ketiga skema modulasi. Namun lebih diprioritaskan untuk menggunakan skema modulasi yang memiliki *data rate* tertinggi, sesuai dengan Tabel 5, yaitu 16-QAM.



Gambar 6. Hasil Simulasi Kondisi *Low Noise*

Dari grafik dapat dilihat bahwa semakin tinggi nilai *epsilon*, semakin sering agen mengambil aksi yang tidak memberikan nilai *reward* tertinggi. Tabel 7 menunjukkan berapa kali *reward* yang didapatkan oleh agen dalam 1000 kali percobaan dengan nilai *epsilon* yang berbeda.

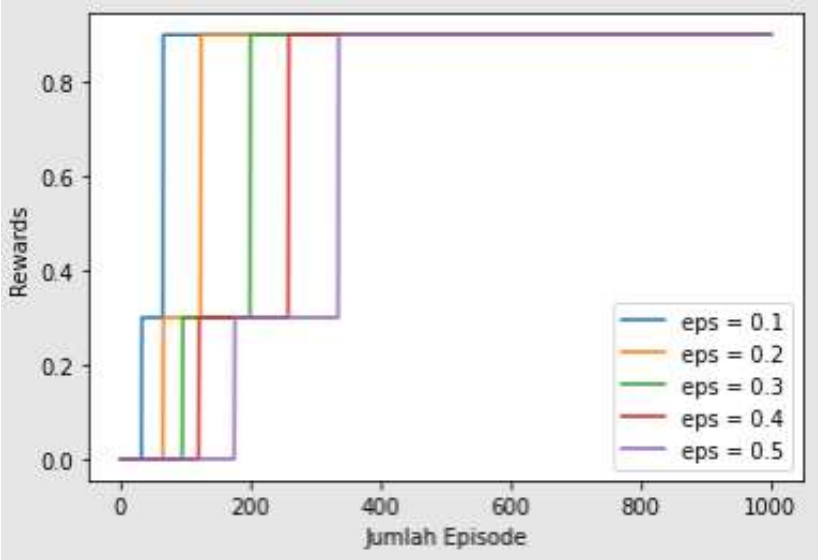
Tabel 7. *Reward* yang didapatkan Agen dalam 1000 Episode pada Kondisi *Low Noise*

		***Reward*s (jumlah episode)**		
		0.9	**0.3**	**0.1**
Epsilon	**0.1**	935	39	26
	0.2	878	59	63
	0.3	780	115	105
	0.4	744	122	134
	0.5	669	156	175

Dari hasil simulasi pada kondisi *low noise* menunjukkan hasil yang mirip dengan yang didapatkan pada simulasi dengan kondisi lingkungan yang berbeda-beda. Pada kondisi *low noise*, agen semakin sering mengambil aksi memilih modulasi dengan *data rate* yang lebih rendah ketika nilai *epsilon* semakin tinggi. Saat *epsilon* 0.1, agen tidak memilih 16-QAM sebanyak 65 kali. Saat *epsilon* bernilai 0.2 skema modulasi tidak dipilih sebanyak 122 kali dan terus meningkat hingga pada *epsilon* 0.5 agen tidak memilih skema modulasi 16-QAM sebanyak 331 kali. Hal tersebut terjadi karena gap antara mode eksplorasi dan eksploitasi semakin besar. Sehingga agen akan semakin sering memilih skema modulasi secara acak dibandingkan memilih skema modulasi terbaik.

3.3 Hasil Simulasi Lingkungan *Medium Noise*

Gambar 7 menunjukkan hasil simulasi dengan 1000 kali percobaan pada kondisi lingkungan *medium noise*. Seperti pada kondisi *low noise*, pada simulasi ini kendaraan diasumsikan melewati 1000 tempat dengan kondisi AWGN yang berbeda namun masih termasuk dalam kondisi *medium noise*.



Gambar 7. Hasil Simulasi Kondisi *Medium Noise*

Dari grafik dapat dilihat bahwa hasil yang didapatkan mirip dengan Gambar dan Gambar . Semakin tinggi nilai *epsilon*, semakin banyak jumlah episode di mana agen tidak memilih skema modulasi terbaik. Tabel 8 menunjukkan jumlah *reward* dan *punishment* untuk nilai yang berbeda selama 1000 kali simulasi.

Tabel 8. *Reward* yang didapatkan Agen dalam 1000 Episode pada Kondisi M*edium Noise*

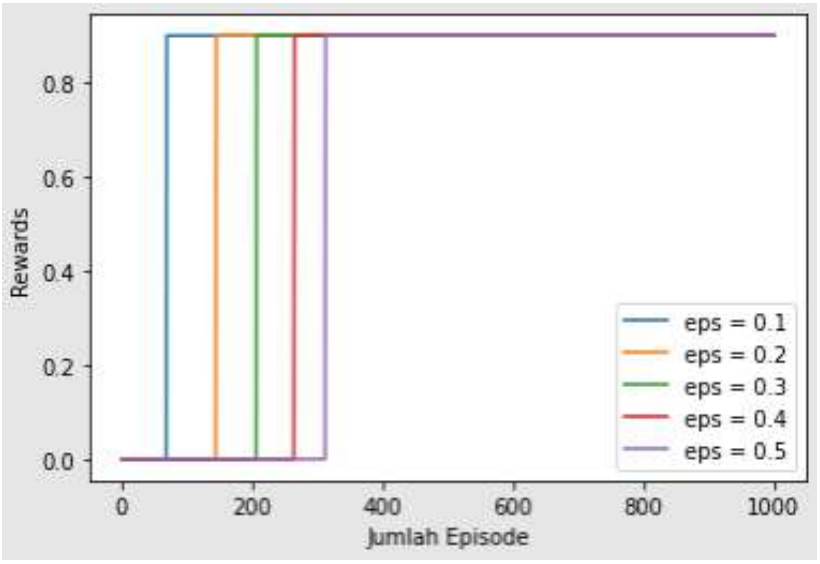
		***Reward*s (jumlah episode)**		
		0.9	**0.3**	**0**
Epsilon	**0.1**	935	33	32
	0.2	877	58	65
	0.3	801	104	95
	0.4	742	138	120
	0.5	665	160	175

Dari Tabel di atas diketahui bahwa saat *epsilon* 0.1 agen mendapatkan *reward* maksimal sebanyak 935 kali dan terus menurun hingga pada *epsilon* bernilai 0.5 agen hanya 665 kali mendapatkan *reward* tertinggi.

Berdasarkan Tabel 3 saat kondisi *medium noise*, agen hanya diperbolehkan menggunakan skema modulasi 8PSK dan QPSK. Namun dari hasil simulasi yang ditunjukan pada Tabel 8 didapatkan hasil bahwa agen juga memilih skema modulasi 16-QAM. Sehingga mendapatkan *punishment* berupa nilai 0. Hal tersebut terjadi karena saat *epsilon* semakin tinggi maka probabilitas agen berada pada mode eksplorasi pun semakin besar. Sehingga agen memilih skema modulasi secara acak tanpa mempertimbangkan apakah akan mendapatkan *reward* atau *punishment*.

3.4 Hasil Simulasi Lingkungan *High Noise*

Gambar 8 menunjukkan hasil simulasi dari 1000 kali percobaan pemilihan skema modulasi oleh agen pada kondisi lingkungan *high noise*.



Gambar 8. Hasil Simulasi Kondisi High *Noise*

Dari gambar didapatkan hasil yang menyerupai dengan grafik pada Gambar . Agen mendapatkan *punishment* yang semakin banyak saat nilai *epsilon* semakin tinggi. Tabel 9 merangkum jumlah *reward* dan *punishment* yang didapatkan oleh agen.

Tabel 9. *Reward* yang Didapatkan Agen dalam 1000 Episode pada Kondisi *High Noise*

		***Reward*s (jumlah episode)**	
		0.9	**0.3**
Epsilon	**0.1**	932	68
	0.2	856	144
	0.3	794	206
	0.4	736	264
	0.5	688	312

Berdasarkan Tabel 3, saat kondisi *high noise* agen hanya diperbolehkan memilih skema modulasi QPSK karena jika dua skema modulasi lain yang dipilih maka nilai probabilitas *error* akan melebihi 10^{-3}. Hal tersebut dapat mengakibatkan tingginya *packet drop*. Namun dari Tabel di atas diketahui bahwa agen mendapatkan *punishment* sebanyak 144 kali saat kondisi *epsilon* 0.2 dan terus meningkat hingga mencapai 312 kali saat *epsilon* bernilai 0.5. Hal tersebut disebabkan oleh semakin besarnya kemungkinan agen memilih skema modulasi secara acak saat nilai *epsilon* semakin tinggi.

4. KESIMPULAN

Dari hasil simulasi, *reinforcement learning* dengan algoritma *epsilon greedy* dapat diimplementasikan untuk skema modulasi adaptif pada V2I. Dengan kondisi nilai *epsilon* yang berbeda-beda mulai dari 0.1 hingga 0.5 didapatkan hasil bahwa semakin tinggi nilai *epsilon* maka semakin sering agen tidak memilih skema modulasi dengan *reward* tertinggi. Hal tersebut disebabkan semakin tinggi nilai *epsilon*, semakin sering agen memilih skema modulasi secara acak.

DAFTAR RUJUKAN

Bliss, D. W., & Govindasamy, S. (2013). *Adaptive Wireless Communications (MIMO Channels and Networks)* (1st Editio). Cambridge University Press.

Cronin, B. (2015). *Connected Vehicle Benefits*. Bureau of Transportation Statistics. https://www.its.dot.gov/factsheets/pdf/ConnectedVehicleBenefits.pdf

Dangi, M., & Porwal, M. K. (2015). Analyses of SNR Threshold for Minimum BER in Various Modulations Schemes and Development Of an Adaptive Modulation Scheme. *IJISET - International Journal of Innovative Science, Engineering & Technology*, *2*(3), 139–142.

dos Santos Mignon, A., & de Azevedo da Rocha, R. L. (2017). An Adaptive Implementation of ε-Greedy in Reinforcement Learning. *Procedia Computer Science*, *109*, 1146–1151. https://doi.org/10.1016/j.procs.2017.05.431

Eska, A. C. (2018). Adaptive Modulation and Coding (AMC) around Building Environment for MS Communication at The Train. *EMITTER International Journal of Engineering Technology*, *6*(2), 386–394. https://doi.org/10.24003/emitter.v6i2.279

Halegoua, G. R. (2020). *Smart Cities*. The MIT Press.

Ippolito Jr., L. J. (2017). *Satellite Communications Systems Engineering: Atmospheric Effects, Satellite Link Design and System Performance* (2nd Editio). Wiley.

Lowe, R., & Ziemke, T. (2013). Exploring the relationship of reward and punishment in reinforcement learning. *2013 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, 140–147. https://doi.org/10.1109/ADPRL.2013.6615000

Masood, R. F. (2013). Adaptive Modulation (QPSK, QAM). *ArXiv Preprint ArXiv:1302.7145*. http://arxiv.org/abs/1302.7145

Nieuwdorp, T. (2017). *Dare to Discover: The Effect of the Exploration Strategy on an Agent's Performance*. Radboud University Nijmegen.

Novfitri, A., Suryani, T., & Suwadi. (2018). Performance Analysis of Vehicle-to-Vehicle Communication with Adaptive Modulation. *2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, 187–191. https://doi.org/10.1109/EECCIS.2018.8692895

Oyetola, O. K., Okubanjo, A. A., Okandeji, A. A., Alao, P. O., Osifeko, M. O., & Olasunkanmi, O. G. (2018). Symbol Error Probability Of 16-QAM System Over AWGN and Rayleigh Fading Channels. *African Journal of Science & Nature*, *7*, 29–39.

Pandey, R., Awasthi, A., & Srivastava, V. (2013). Comparison between Bit Error Rate And Signal To Noise Ratio in OFDM Using LSE Algorithm. *Conference on Advances in Communication and Control Systems 2013 (CAC2S 2013)*, 463–466.

Ravinchandiran, S. (2018). *Hands-On Reinforcement Learning with Python*. Packt Publishing Ltd.

Rochmatika, R. A., Suryani, T., & Wirawan. (2018). Performance of Adaptive Modulation over Frequency Selective Fading Channel in VANET Environment. *2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 400–405.

Sassi, A., Charfi, F., Kamoun, L., Elhillali, Y., & Rivenq, A. (2012). OFDM Transmission Performance Evaluation in V2X Communication. *International Journal of Computer Science Issues*, *9*(2), 141–148. http://arxiv.org/abs/1410.8039

Sighal, M., Agarwal, M., Trikha, M., & Sharma, N. (2013). Bit Error Rate Performance of Gray Coded 8-PSK. *MIT International Journal of Electronics and Communication Engineering*, *3*(1), 20–24.

Singya, P. K., Shaik, P., Kumar, N., Bhatia, V., & Alouini, M.-S. (2021). A Survey on Higher-Order QAM Constellations: Technical Challenges, Recent Advances, and Future Trends.

IEEE Open Journal of the Communications Society, 1–1. https://doi.org/10.1109/OJCOMS.2021.3067384

Skrucany, T., Sarkan, B., FigluzTomasz, Synak, F., & Vrabel, J. (2017). Measuring of noise emitted by moving vehicles. *Dynamics of Civil Engineering and Transport Structures and Wind Engineering – DYN-WIND'2017*, *107*. https://doi.org/10.1051/matecconf/2017107000

Sutton, R. S., & Barto, A. G. (2015). *Reinforcement Learning: An Introduction*. The MIT Press.

Thomas, B. (2016). Proposed rule would mandate vehicle-to-vehicle (V2V) communication on light vehicles, allowing cars to "talk" to each other to avoid crashes. *National Highway Traffic Safety Information*.

Wietfeld, C., & Ide, C. (2015). Vehicle-to-infrastructure communications. In *Vehicular Communications and Networks,* (pp. 3–28). Elsevier. https://doi.org/10.1016/B978-1-78242-211-2.00001-5