

## Desain Perilaku *Home Position* Pada Robot Soccer Dan Kursi Bergerak Otonom Menggunakan Algoritma *Q-Learning*

Andi Hasad<sup>1</sup>, Yusuf Hanafi<sup>2</sup>, Risa Sartika<sup>3</sup>

Program Studi Teknik Elektro Universitas Islam "45" Bekasi

<sup>1</sup>andihasad@gmail.com, <sup>2</sup>yusufhanafi611@gmail.com, <sup>3</sup>risasartika1996@gmail.com

### Abstrak

Robot *soccer* merupakan robot mobil tipe beroda yang memiliki kerumitan dalam rancang bangun robot. Kerumitan dalam rancang bangun robot, membuat pengembangan riset di bidang robotika semakin meningkat. Penelitian ini membahas perilaku *home position* (kembali ke titik semula) menggunakan algoritma *Q-learning* yang diimplementasikan pada robot *soccer* dan kursi bergerak otonom. Sistem perilaku yang diimplementasikan, robot dapat bergerak secara acak dalam menelusuri area kotak warna untuk mencapai titik target. Di sisi lain, menggunakan kursi sebagai objek yang akan dikendalikan secara acak dengan algoritma *Q-Learning* yang dipadukan dengan *camera webcam* sebagai sensor pendeteksi warna area pergerakan robot. Dari hasil pengujian yang telah dilakukan, hasil perilaku optimal robot untuk mencapai titik target yaitu ungu-kuning-biru dengan nilai  $Q = 5.13$  dan nilai tanggapan waktunya yaitu *time peak* ( $t_p$ ) = iterasi 12, *time rise* ( $t_r$ ) = iterasi 8, *time settling* ( $t_s$ ) = iterasi 9, *time delay* = iterasi 6, dan nilai *error steady state* = 0, sedangkan pada kursi bergerak otonom dengan nilai  $Q = 5.13$ , jumlah iterasi 9, jumlah langkah sedikit dan mendapatkan respon tanggapan waktu sebagai berikut: *time rise* ( $t_r$ ) = 9 iterasi, *time peak* ( $t_p$ ) = 7 iterasi, *time settling* ( $t_s$ ) = 9 iterasi, *delay time* ( $t_d$ ) = 6 iterasi dan nilai *error steady state* adalah nol (0) pada iterasi ke 9.

**Kata Kunci:** *Q-learning, Robot Soccer, Raspberry Pi, Camera Webcam, Kursi Bergerak Otonom*

### Abstract

*Robot soccer is a type of wheeled car robot that has complexity in the design of the robot. The complexity in the design of the robot makes research development in the field of robotics increasing. This study discusses home position behavior (back to its original point) using the Q-learning algorithm which is implemented on robot soccer. Behavior system is implemented, the robot can move randomly in tracing the color box area to reach the target point. On the other side, using a chair as an object that will be randomly controlled with the Q-Learning algorithm combined with a webcam camera as a sensor to detect the color of the robot's movement area. Color acts as an initial identification so that the robot easily recognizes and knows the target point and area of its movement. From the results of tests that have been done, the optimal behavior of robots to reach the target point is purple-yellow-blue with a value of  $Q = 5.13$  and the time response value is *time peak* ( $t_p$ ) = iteration 12, *time rise* ( $t_r$ ) = iteration 8, *time settling* ( $t_s$ ) = iteration 9, *time delay* = iteration 6, and the value of the steady state error = 0, while in an autonomous moving chair with a value of  $Q = 5.13$ , the number of iterations 9, the number of steps is small and the response time response as follows: *time rise* ( $t_r$ ) = 9 iterations, *time peak* ( $t_p$ ) = 7 iteration, *time settling* ( $t_s$ ) = 9 iterations, *delay time* ( $t_d$ ) = 6 iterations and the steady state error value is zero (0) on the 9th iteration.*

**Keywords :** *Q-learning, Robot Soccer, Raspberry Pi, Camera Webcam, Autonomous Moving Chair*

## PENDAHULUAN

Robot sepak bola merupakan robot mobil tipe beroda yang dirancang agar memiliki fleksibilitas gerakan yang baik dalam berpindah-pindah tempat, seperti bergerak maju, mundur, serong, geser, dan *manuver*. Secara konstruksi, selain robot ini dirancang untuk menghindari rintangan, robot ini juga didesain untuk dapat kembali ke posisi awal setelah robot bergerak dan berpindah posisi di dalam area lingkungannya. Supaya robot dapat kembali ke posisi awal (*Homing*-nya) dengan pergerakannya yang secara acak, maka robot perlu suatu kemampuan untuk menyelesaikan misinya. Oleh karena itu, robot membutuhkan sistem pembacaan yang berbasis kamera. Hal ini dikarenakan objek yang dicari dan digunakan di dalam area lingkungannya tidak bisa hanya mengandalkan sensor selain kamera, seperti untuk mengenali warna yang akan dianalogikan sebagai lapangan[1].

Robot bergerak diharuskan untuk bernavigasi dari satu tempat ke tempat lain secara otomatis untuk mencapai titik target. Robot tersebut ditugaskan untuk mengenal area pergerakannya, sehingga proses pembelajarannya dalam mengenali area dapat terlaksana dengan baik. Metode yang digunakan adalah salah

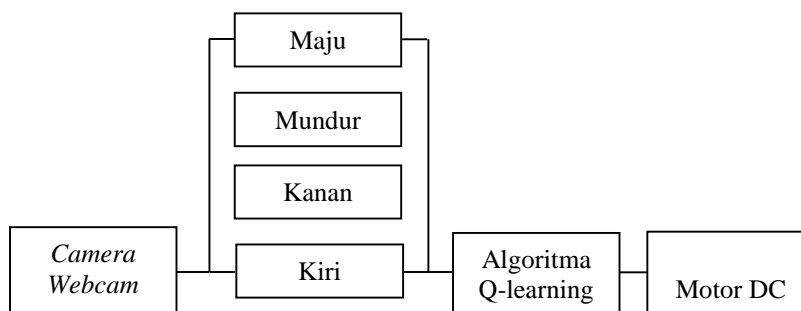
satu algoritma dalam *Reinforcement Learning* (RL) yaitu *Q-Learning*. Pada pembelajaran RL, robot belajar melalui interaksi dengan lingkungannya. Robot menerima kondisi (*state*) lingkungannya dan memilih aksi yang tepat untuk diberikan ke lingkungannya. Kemudian keadaan lingkungan berubah dan robot menerima penghargaan (*reward*) berupa nilai dari lingkungannya. Tujuan belajar disini adalah memilih aksi yang dapat memaksimalkan penghargaan yang diterima dan robot dapat mencapai titik target dengan jalur tercepat[2].

*Q-Learning* menjawab pertanyaan terhadap koordinasi behavior (tingkah laku) robot. Algoritma *Q-Learning* merupakan sebuah simulasi dasar teknik stokastik (model matematika untuk mengukur nilai yang berubah terhadap waktu dengan derajat kepastian yang tidak stabil) yang menyediakan cara untuk menghubungkan *state*, *action* dan *reward* berupa nilai Q dalam tabel. Dalam aplikasinya, kontroler sederhana mencari dalam tabel dan memilih *decision* (keputusan) dengan nilai terbaik dalam mencapai target[3].

## METODE PENELITIAN

### Arsitektur Perilaku Robot Soccer

Perilaku robot *soccer* pada proses penelusuran area untuk mencapai titik target, menggunakan pola aksi secara acak dalam pergerakannya. Perilaku aksi ini menggunakan, perilaku aksi maju, perilaku aksi mundur, perilaku belok kanan, dan perilaku belok kiri.



Gambar 1. Sistem Arsitektur Perilaku

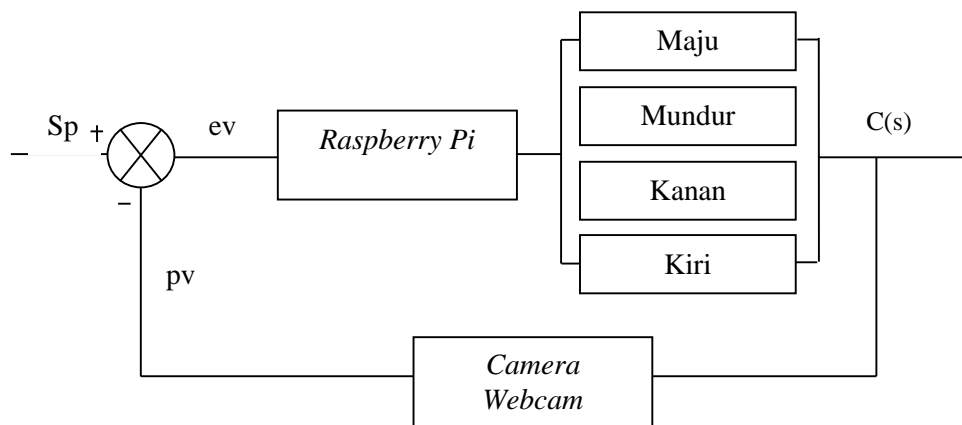
Gambar 1 menjelaskan sistem arsitektur perilaku dan interaksi perilaku robot dengan area lingkungan simulasi robot. Sesuai dengan perilaku aksi yang dikendalikan dengan algoritma *Q-learning*. Pada tiap kali melangkah, robot akan mengamati keadaan saat ini melalui sensor kamera (*Webcam*) dan menimbulkan sebuah aksi dari serangkaian perilaku aksi yang berhubungan dengan keadaan[4]. Aksi perilaku gerak robot ini didesain di dalam software pemrograman berbahasa *python* yang tersedia pada perangkat Raspberry Pi. Kode program yang telah dibuat akan di *upload* (ditanamkan) ke dalam Raspberry Pi[5].

Kode program yang telah dibuat dan ditanamkan kepada pengendali Raspberry Pi, terdapat beberapa pin-pin driver motor yang diinisialisasikan sebagai logika aksi gerak robot. ENA dan ENB berfungsi sebagai pengaktifan *driver* motor dan *input* PWM pada motor. IN1, IN2 IN3 dan IN4 pada *driver motor* berfungsi sebagai *input* logika untuk putaran motor. OUT1, OUT2, OUT3, dan OUT4 berfungsi sebagai *output* untuk arah gerak putaran motor[6].

### Pengendali Perilaku

Pengendali yang digunakan dalam penelitian ini yaitu menggunakan aksi umpan balik dalam sistem pengotrolannya untuk memperkecil suatu sistem. Pengendalian pada penelitian ini diproses oleh pengendali Raspberry Pi dengan pengendalian dari perilaku *learning* (*Q-learning*).

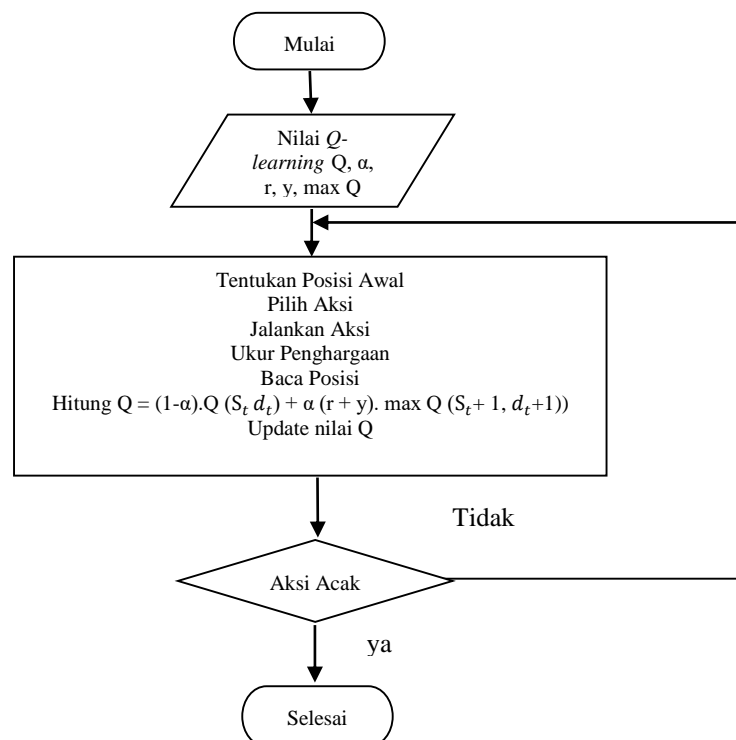
Perakitan pengendali perilaku yang dilakukan pada penelitian ini yaitu berbentuk kode program yang dibuat dengan menggunakan *software python*. Proses pergerakan robot *soccer* dalam mencari titik target diawali ketika sensor kamera (*webcam*) mulai membaca dimana posisi awal keberadaan robot. Sensor kamera (*webcam*) kemudian mengirimkan sinyal masukan kepada Raspberry Pi. Raspberry Pi akan menerima dan mengolah atas sinyal masukan yang dikirim oleh kamera. Selanjutnya, Raspberry Pi akan mengeluarkan hasil sinyal yang telah diolah berupa perilaku aksi robot[7]. Demikian seterusnya hingga robot mencapai titik tujuan atau target.



Gambar 2. Blok Diagram Pengendali Perilaku

### Pembelajaran learning (Q-learning)

Perakitan sistem pembelajaran *learning (Q-learning)* yang diterapkan pada robot *soccer* dalam penelitian ini dapat dilihat seperti Gambar 3.



Gambar 3. Flowchart Q-learning

Pembelajaran *learning (Q-learning)* ini dibuat dalam bentuk program menggunakan *software python* yang ditanamkan ke dalam pengendali Raspberry Pi. Gambar 3 di atas menjelaskan mengenai sistem kerja dari algoritma *Q-learning* yang diterapkan pada robot *soccer* beroda. Algoritma *Q-learning* yang diimplementasikan kedalam robot *soccer* berfungsi untuk memudahkan proses pembelajaran dalam menganalisa perhitungan disaat robot melakukan penelusuran untuk mencapai titik target.

### Pengujian Sistem

Pengujian yang dilakukan untuk melihat kinerja robot *soccer* beroda yaitu meliputi :

### Tanggapan Keadaan Mantap

Tanggapan keadaan mantap adalah keadaan dimana ketika tanggapan suatu grafik mulai menyentuh titik target (*setpoint*) yang telah ditentukan. Pada penelitian ini grafik keadaan mantap didapat ketika robot *soccer* berhasil mencapai titik target. Dalam Tanggapan keadaan mantap terdapat 2 hal yang perlu diperhatikan yaitu error keadaan mantap dan besar nilai error keadaan mantap [8].

### Tanggapan Waktu

Tanggapan waktu adalah respon tanggapan dari keadaan dimana ketika robot melakukan aksi berjalan secara acak menelusuri area untuk menemukan titik target [9]. Berikut merupakan data-data dari tanggapan waktu yang akan diambil :

1. *Time Peak* ( $t_p$ ) adalah waktu ketika respon dari grafik mulai mencapai titik puncak pertama kali.
2. *Time Rise* ( $t_r$ ) adalah waktu ketika keberadaan suatu respon mulai masuk 5 % sd 95 % dari respon nilai tanggapan.
3. *Time Settling* ( $t_s$ ) adalah waktu respon ketika mencapai nilai akhir dan tetap berada pada nilai tersebut dalam range presentase tertentu dari nilai akhir 5% atau 2%).
4. *Delay Time* ( $t_d$ ) adalah waktu untuk mencapai setengah dari titik puncak tanggapan pertama kali.

### Pengujian Q

Pengujian Q yang dilakukan dalam penelitian ini adalah proses pengambilan nilai Q dari data-data yang dilakukan dalam beberapa percobaan berdasarkan iterasi dan langkah robot ketika mencapai titik target (tujuan).

### Analisa Data

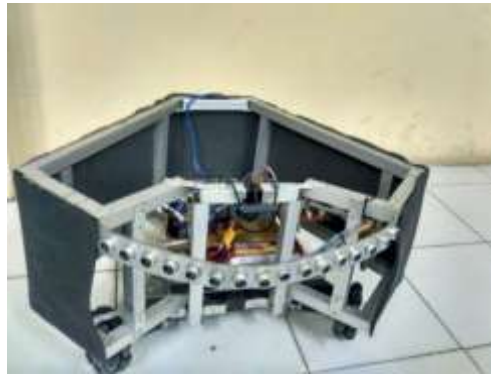
Analisa yang dilakukan dalam penelitian ini bertujuan untuk menganalisis data-data yang sudah diperoleh. Apakah data-data tersebut sudah berhasil atas penggunaan metode algoritma *Q-learning* yang telah diterapkan pada penelitian ini. Analisa yang perlu dilakukan dalam penelitian ini yaitu :

1. Analisa Arsitektur Perilaku  
Analisa Arsitektur Perilaku dalam penelitian ini dilakukan untuk mengetahui apakah langkah gerak robot *soccer* sudah sesuai dengan perancangan sistem arsitektur perilaku yang telah dibuat.
2. Analisa Pengendali Perilaku  
Analisa Pengendali Perilaku dalam penelitian ini dilakukan untuk mengetahui apakah pergerakan robot *soccer* ketika dalam proses penelusuran untuk mencapai titik target sudah sesuai dengan metode algoritma *Q-learning* yang telah dibuat dalam software *python*.
3. Analisa Keberhasilan *Q-learning*  
Analisa Keberhasilan *Q-learning* dalam penelitian ini dilakukan untuk mengetahui apakah metode algoritma *Q-learning* yang telah diimplementasikan pada robot *soccer* sudah sesuai dengan kode pemrograman yang ditanamkan pada Raspberry Pi. Dengan memperoleh langkah terbaik, iterasi sedikit, dan tidak adanya pengulangan langkah (bolak-balik) dari kotak sekarang ke kotak sebelumnya.
4. Analisa Tanggapan Waktu  
Tanggapan waktu adalah respon terhadap masukan saat robot melakukan aksi secara acak hingga mencapai titik target. Adapun data yang dibutuhkan adalah *time peak* ( $t_p$ ), *time rise* ( $t_r$ ), *time settling* ( $t_s$ ), dan *delay time*.
5. Analisa Tanggapan Keadaan Mantap  
Keadaan mantap (*steady state*) dianalisa berdasarkan aspek kesalahan keadaan mantap (*error steady state*). Untuk memperoleh gambaran aspek *error steady state*, maka sistem diujikan dengan menggunakan sinyal uji.

## HASIL DAN PEMBAHASAN

### Sistem Kendali

Objek yang dilakukan pada penelitian ini yaitu metode algoritma *Q-learning* dengan pengendali Raspberry Pi yang diimplementasikan pada robot *soccer* dan kursi bergerak otonom; yang bertugas untuk mencari titik terget dari titik awalnya. Rangkaian penggerak yang digunakan pada robot *soccer*, terdiri dari baterai Lipo 12 volt, driver motor L298N, 2 Motor DC, dan 2 roda Omni Directional [10].



**Gambar 4. Robot Soccer Beroda**



**Gambar 5. Kursi Bergerak Otonom**

Pada pembelajaran *learning (Q-learning)* yang diterapkan pada penelitian ini. Robot *soccer* akan diletakkan di dalam suatu area persegi panjang yang di dalamnya terdiri dari 6 kotak warna dengan jumlah ukuran sebesar  $124 \times 116$  cm. Kemudian robot akan berjalan menelusuri kotak warna secara acak sesuai perilaku aksi yang dipilih oleh robot. Pada penelusuran kotak warna yang hendak dilakukan oleh robot ini, akan dibentuk menjadi beberapa percobaan. Dikatakan satu percobaan, apabila robot dapat menyelesaikan penelusuran dari titik awal hingga menemukan titik target atau keluar kotak. Percobaan ini dilakukan sesuai dengan diagram alir (*flowchart*) *Q-learning* yang telah dibentuk pada Gambar 3. Diagram alir (*flowchart*) *Q-learning* ini di ubah ke dalam bentuk program dengan *software* pemrograman berbahasa *python*.

Program *Q-learning* yang terdapat pada Raspberry Pi dapat dijelaskan bahwa, gerak robot *soccer* dari titik awal untuk mencapai titik terget pada penelitian ini yaitu dengan melakukan aksi secara acak. Aksi 1 sebagai gerak maju, aksi 2 sebagai gerak mundur, aksi 3 sebagai gerak belok kanan, dan aksi 4 sebagai belok kiri. Status keberadaan robot ketika dalam proses penelusuran mencari titik target, dapat dilihat pada tabel percobaan pada bagian kotak sekarang. Kotak 0 sebagai titik awal (warna ungu), Kotak 4 sebagai kotak target (warna biru), kotak 1 sebagai (warna hijau), kotak 2 sebagai (warna hitam), kotak 3 sebagai (warna kuning), dan kotak 5 sebagai (warna merah). Nilai penghargaan awal yang dimiliki oleh robot yaitu 100. Selama robot dalam proses penelusuran mencari titik target, nilai penghargaan yang dimiliki robot akan berkurang 1 dan berlaku juga, untuk robot yang berpindah tempat dari kotak sebelumnya ke kotak sekarang jika menjauhi titik target. Namun, bila robot berjalan mendekati titik target nilai penghargaan robot akan bertambah 10.

Arsitektur perilaku kursi otonom adalah sebagai berikut:

1. Aksi Maju
2. Aksi Mundur
3. Aksi Belok Kanan
4. Aksi Belok Kiri

Untuk mengetahui kemampuan kursi dalam bergerak otonom dalam memilih aksi secara acak maka dibutuhkan area yang tepat, dimana area ini telah ditentukan yaitu berbentuk kotak berwarna yang berukuran  $126 \times 116$  dapat dilihat pada Gambar 6 area pergerakan kursi terdiri dari 6 warna untuk membedakan kotak satu dengan kotak yang lain dan salah satu dari warna dipilih menjadi target tujuan kursi. Adapun area pergerakan kursi dapat dilihat pada Gambar 6 berikut.



Gambar 6. Kotak Warna Pergerakan Kursi.

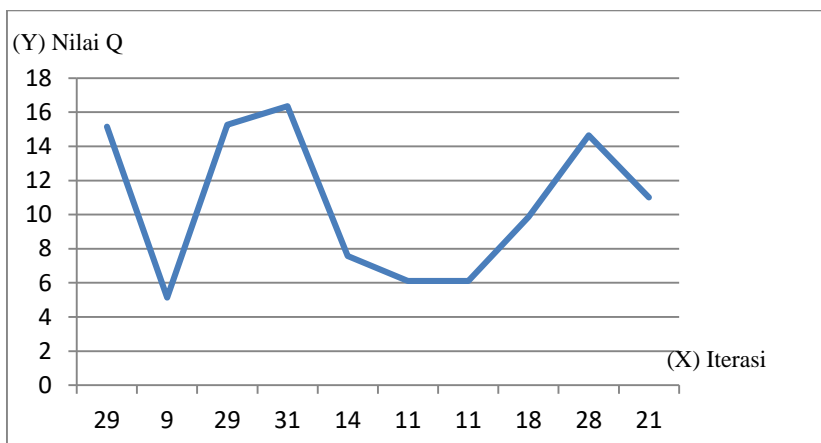
Kotak awal dan kotak target dapat dirubah kapan saja sesuai keinginan. Pergerakan kursi secara keseluruhan dikendalikan oleh program algoritma *Q-Learning* menggunakan *software python* yang ditanamkan pada perangkat *raspberry pi*.

### Hasil Pengujian

Percobaan yang dilakukan dalam penelitian robot *soccer* menggunakan metode algoritma *Q-learning* yang bertugas untuk menemukan titik target (kotak target) dari keberadaan titik awal (kotak awal) yaitu sebanyak sepuluh kali percobaan. Pengujian dilakukan dengan kotak awal ungu dan kotak target biru.

### Pembahasan

Berdasarkan data hasil pengujian yang telah dilakukan dari 10 percobaan pada robot *soccer* menggunakan metode *Q-learning*, maka diperoleh nilai dari masing-masing percobaan seperti Tabel 1.



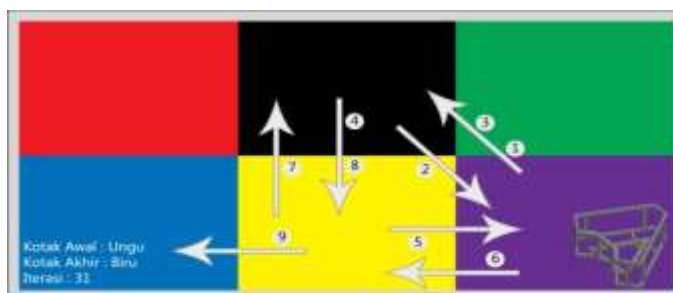
Gambar 7. Hasil Grafik Nilai Q Terhadap Iterasi Pada Tiap-Tiap Percobaan

Tabel 1. Hasil Percobaan Keseluruhan

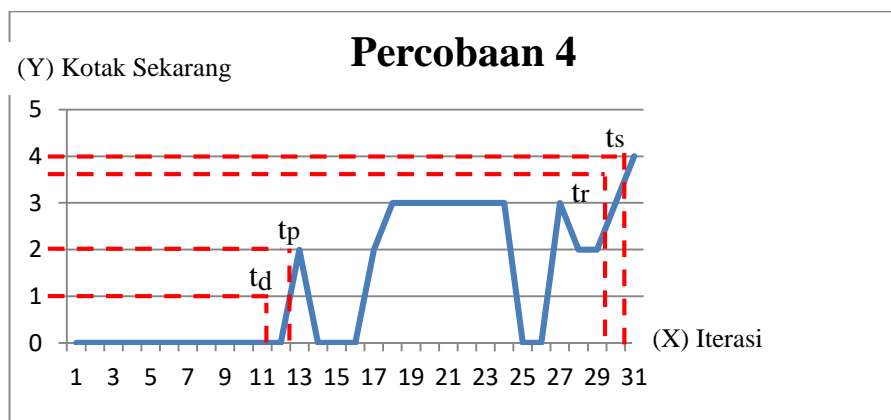
Percobaan	Nilai Q	Iterasi
1.	15.15	29
2.	5.13	9
3.	15.26	29
4.	16.35	31
5.	7.57	14
6.	6.11	11
7.	6.11	11
8.	9.87	18
9.	14.65	28
10.	11.01	21

Dari Tabel 1 hasil percobaan keseluruhan di atas, maka dapat ditarik 3 versi *Q-learning* untuk penelusuran arah jejak robot *soccer* dalam mencapai titik target berdasarkan iterasi, dan langkah pergerakan yang baik (tidak bolak-balik). Tiga versi *Q-learning* yang didapat adalah sebagai berikut :

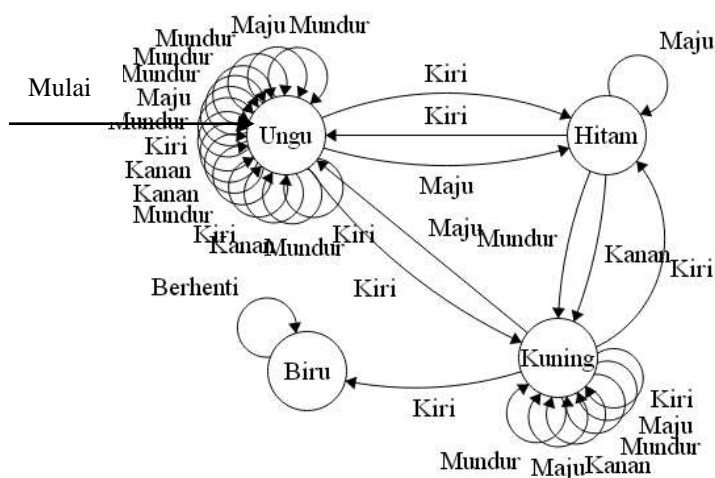
1. Urutan langkah maksimal versi *Q-learning* untuk penelusuran arah jejak robot *soccer* di titik awal kotak warna ungu dengan titik target kotak warna biru terletak pada hasil percobaan keempat. Urutan kotak yang dilewati ketika proses *learning* yaitu, kotak warna ungu-hitam-ungu-hitam-kuning-ungu-kuning-hitam-kuning-biru. Dari urutan langkah ini diperoleh bahwa waktu nilai Q awal hingga akhir adalah :
- a. Ungu : 6.38
  - b. Hitam : 6.98
  - c. Ungu : 8.45
  - d. Hitam : 9.05
  - e. Kuning : 12.59
  - f. Ungu : 13.57
  - g. Kuning : 14.17
  - h. Hitam : 15.15
  - i. Kuning : 15.17
  - j. Biru : 16.35



Gambar 8. Alur Gerak Robot Percobaan Keempat



Gambar 9. Grafik Respon Tanggapan Percobaan keempat

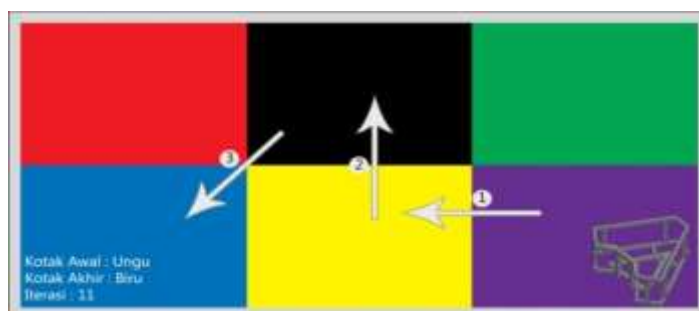


Gambar 10. Hasil Tingkah Laku Robot Versi Maksimal

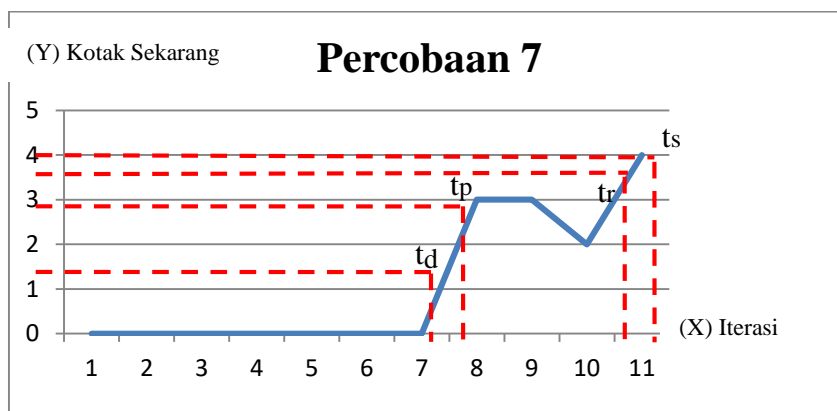
Jika diperoleh urutan nilai Q yaitu 6.32 sampai dengan 16.35 dalam proses mencapai titik target. Maka, langkah ini cukup logis untuk diambil sebagai langkah maksimal versi *Q-learning* dalam proses mencapai titik

taget. Karena, pada proses ini robot diperoleh langkah yang banyak yaitu 9 langkah (9 kali pindah kotak) untuk mencapai titik target dengan jumlah iterasi 31. Adapun nilai dari hasil grafik respon tanggapan percobaan keempat seperti Gambar 9 di bawah ini, didapatkan nilai tanggapan waktunya yaitu *time peak* ( $t_p$ ) = iterasi 13, *time rise* ( $t_r$ ) = iterasi 30, *time settling* ( $t_s$ ) = iterasi 31, *time delay* = iterasi 12, dan nilai *error steady state* = 0.

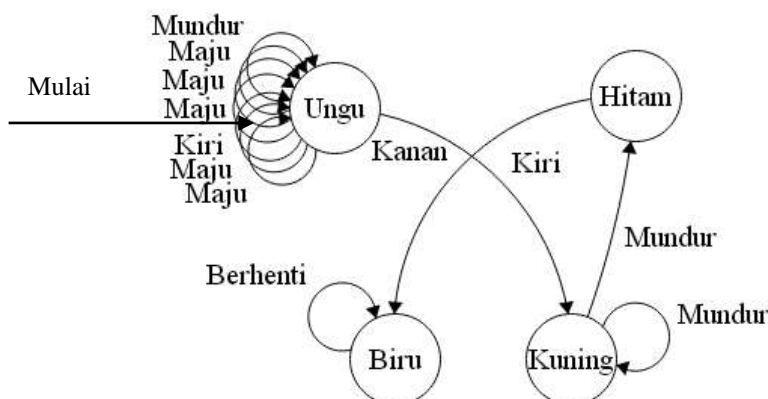
2. Urutan langkah sedang versi *Q-learning* untuk penelusuran arah jejak robot *soccer* di titik awal kotak warna ungu dengan titik target kotak warna biru terletak pada hasil percobaan ketujuh. Urutan kotak yang dilewati ketika proses *learning* yaitu, kotak warna ungu-kuning-hitam-biru. Dari urutan langkah ini diperoleh bahwa waktu nilai *Q* awal hingga akhir adalah :
  - a. Ungu : 3.92
  - b. Kuning : 5.02
  - c. Hitam : 5.51
  - d. Biru : 6.11



Gambar 11. Alur Gerak Robot Percobaan Ketujuh



Gambar 12. Grafik Respon Tanggapan Percobaan Ketujuh



Gambar 13. Hasil Tingkah Laku Robot Versi Sedang

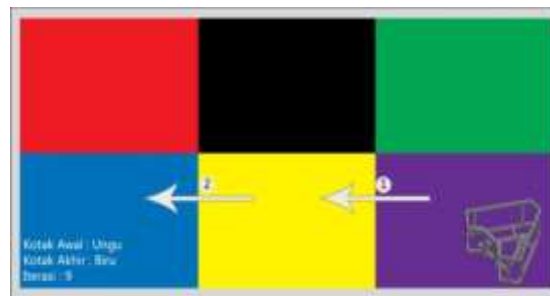


Jika diperoleh urutan nilai Q yaitu 3.92 sampai dengan 6.11 dalam proses mencapai titik target. Maka, langkah ini cukup logis untuk diambil sebagai langkah sedang versi *Q-learning* dalam proses mencapai titik target. Karena, pada proses ini robot diperoleh langkah yang sedikit yaitu 4 langkah (4 kali pindah kotak) untuk mencapai titik target dengan jumlah iterasi 18. Serta tidak adanya proses bolak balik gerak robot dari kotak sekarang ke kotak sebelumnya. Dan nilai tanggapan waktunya *time peak* ( $t_p$ ) = iterasi 8, *time rise* ( $t_r$ ) = iterasi 10, *time settling* ( $t_s$ ) = iterasi 11, *time delay* = iterasi 7, dan nilai *error steady state* = 0.

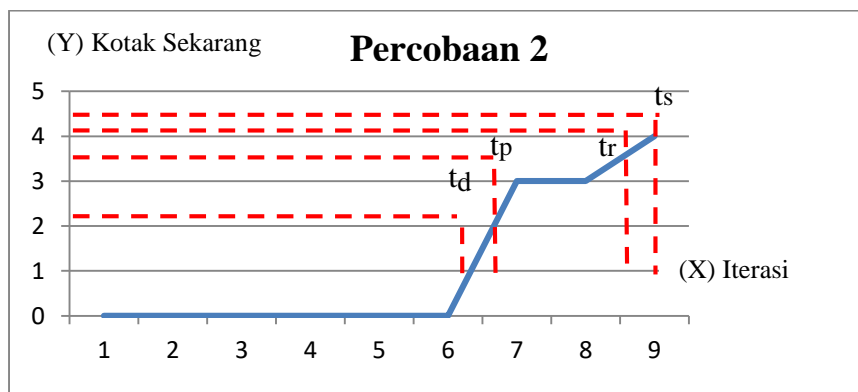
3. Urutan langkah minimal versi *Q-learning* untuk penelusuran arah jejak robot *soccer* di titik awal kotak warna ungu dengan titik target kotak warna biru terletak pada percobaan kedua. Urutan kotak yang dilewati ketika proses *learning* yaitu, kotak warna ungu-kuning-biru. Dari urutan langkah ini diperoleh bahwa waktu nilai Q awal hingga akhir adalah :

- a. Ungu : 3.44
- b. Kuning : 4.53
- c. Biru : 5.13

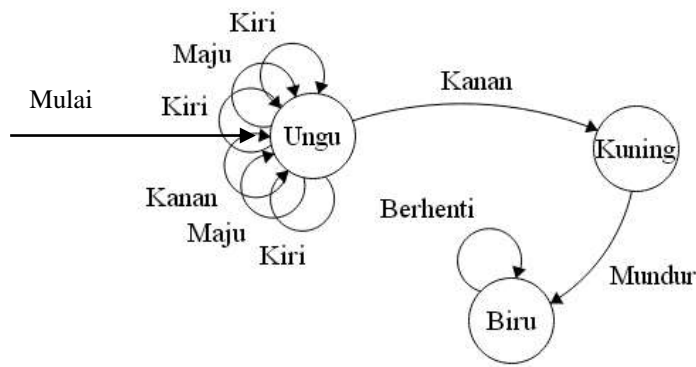
Jika diperoleh urutan nilai Q yaitu 3.44 sampai dengan 5.13 dalam proses mencapai titik target. Maka, langkah ini cukup logis untuk diambil sebagai langkah minimal versi *Q-learning* dalam proses mencapai titik target. Karena, pada proses ini robot diperoleh langkah yang sedikit yaitu 2 langkah (2 kali pindah kotak) untuk mencapai titik target dengan jumlah iterasi sembilan. Serta tidak adanya proses bolak balik gerak robot dari kotak sekarang ke kotak sebelumnya. Dan nilai tanggapan waktunya *time peak* ( $t_p$ ) = iterasi 12, *time rise* ( $t_r$ ) = iterasi 8, *time settling* ( $t_s$ ) = iterasi 9, *time delay* = iterasi 6, dan nilai *error steady state* = 0.



Gambar 14. Alur Gerak Robot Percobaan Kedua



Gambar 15. Grafik Respon Tanggapan Percobaan Kedua



Gambar 16. Hasil Tingkah Laku Robot Versi Minimal

Pada kursi otonom, diambil 3 versi percobaan dari 10 kali percobaan Q-learning yaitu Hasil *Q-Learning* versi terburuk, Hasil *Q-Learning* sedang dan Hasil *Q-Learning* terbaik.

1. Hasil *Q-Learning* Versi Terburuk

Urutan langkah terburuk versi *q-learning* untuk penentu arah jejak kursi dari titik awal kotak warna hijau dan titik target kotak warna biru terdapat pada percobaan pertama. Urutan nilai Q yang dilewati ketika pembelajaran learning pada percobaan terburuk dapat dilihat pada Gambar 17 berikut.



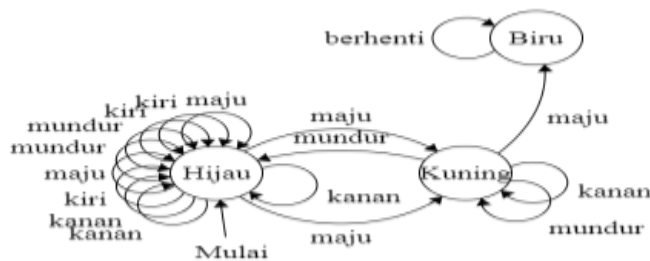
Gambar 17. Hasil Percobaan Versi Terburuk.

- 1) Warna hijau
- 2) Warna kuning
- 3) Warna hijau
- 4) Warna kuning
- 5) Warna biru

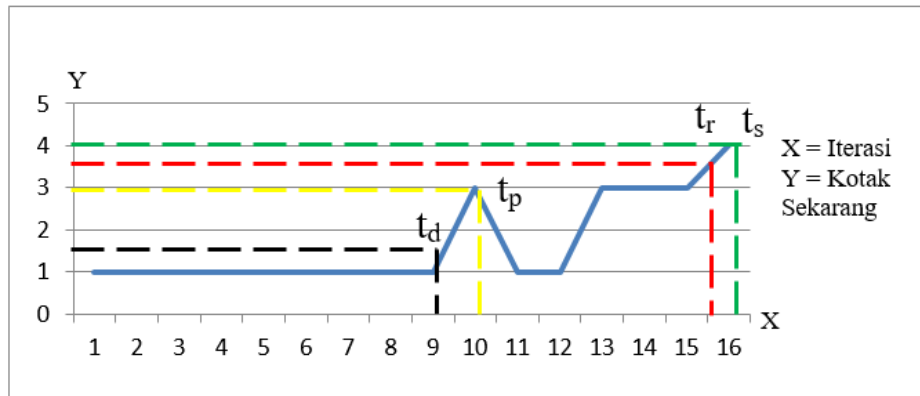
Urutan langkah ini diperoleh dari urutan nilai Q terkecil hingga terbesar, yaitu sebagai berikut.

- 1) Hijau = 4.91
- 2) Kuning = 5.51
- 3) Hijau = 6.49
- 4) Kuning = 8.07
- 5) Biru = 8.67

Jika dilihat dari urutan langkah dengan nilai Q = 4.91 s/d 8.67, maka langkah ini dianggap cukup logis karena berurutan untuk sampai ke titik target. Namun, percobaan pertama dikategorikan terburuk diantara 10 percobaan karena iterasinya lama dengan langkah cukup banyak dan bolak-balik dalam melewati 1 kotak. Adapun hasil perilaku robot kursi dapat dilihat pada Gambar 18 dibawah ini.



Gambar 18. Hasil Perilaku Robot Kursi Versi Terburuk.



Gambar 19. Grafik Respon Percobaan Q-Learning Versi Terburuk

Berdasarkan pada Gambar 19, grafik respon percobaan Q learning dapat dijelaskan sebagai berikut:

a) Analisa respon tanggapan waktu:

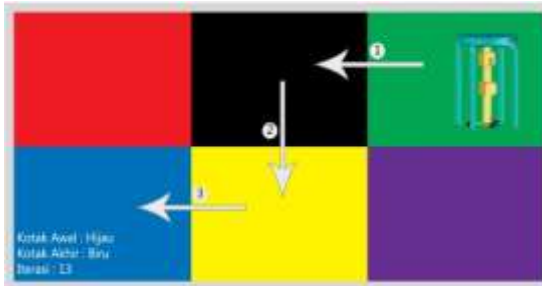
- 1) Waktu Naik ( $t_r$ ) adalah waktu ketika keberadaan suatu respon mulai masuk 10% s/d 90% dari nilai titik targetnya. Grafik respon *Q-Learning* maka waktu untuk gelombang bergerak dari 10% s/d 90% dari nilai titik target pada robot kursi melakukan pergerakan saat nilai iterasi 16. Maka nilai waktu naik pada robot kursi adalah 16 iterasi.
- 2) Waktu Puncak ( $t_p$ ) adalah waktu ketika respon telah mencapai nilai puncak lewatan yang pertama kali. Grafik respon *Q-Learning* waktu puncak pada robot kursi terjadi saat nilai iterasi = 10.
- 3) Waktu Tetap ( $t_s$ ) adalah waktu ketika mencapai nilai akhir dari tanggapan dan tetap berada pada nilai tersebut dalam rentang persentase tertentu dari nilai akhir (5% atau 2%). Grafik respon *Q-Learning* maka waktu untuk tanggapan mencapai dan bertahan pada 2% dari nilai *steady state* (keadaan mantap) pada robot kursi dapat ditempuh saat nilai iterasi 16, jadi nilai waktu tetap robot kursi adalah 16 iterasi.
- 4) Waktu Tunda ( $t_d$ ) adalah waktu untuk mencapai setengah harga dari nilai akhir dari tanggapan untuk pertama kali. Grafik respon *Q-Learning* waktu tunda terjadi saat iterasi ke 9.

b) Respon Keadaan Mantap

Keadaan mantap adalah keadaan yang dimulai pada saat tanggapan mulai pertama kali mendekati nilai akhir hingga waktu yang tak terhingga. Pada percobaan pertama kursi berhasil mencapai titik target dengan nilai  $error = 0$  pada iterasi 16. Tanggapan dari proses iterasi 1 s/d iterasi 16 nilai  $error$  pada kursi adalah mendekati 0 (nol).

2. Hasil *Q-Learning* Versi Sedang

Urutan langkah sedang versi *q-learning* untuk penentu arah jejak kursi dari titik awal kotak warna hijau dan titik target kotak warna biru terdapat pada percobaan kesembilan. Urutan nilai Q yang dilewati ketika pembelajaran learning pada percobaan versi sedang dapat dilihat pada Gambar 20 berikut.



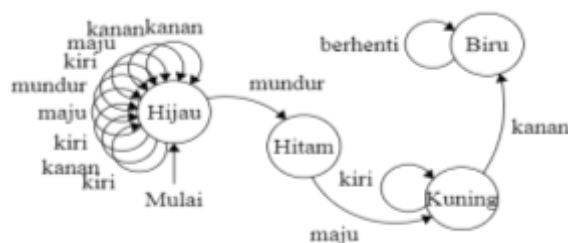
Gambar 20. Hasil Percobaan Versi Sedang.

- 1) Warna hijau
- 2) Warna hitam
- 3) Warna kuning
- 4) Warna biru

Urutan langkah ini diperoleh dari urutan nilai Q terkecil hingga terbesar, yaitu sebagai berikut:

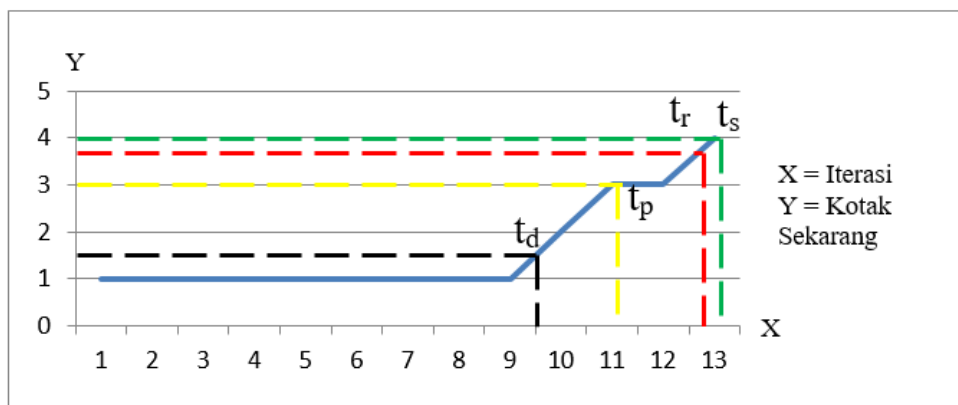
- 1) Hijau = 4.91
- 2) Hitam = 5.51
- 3) Kuning = 6.6
- 4) Biru = 7.2

Jika dilihat dari urutan langkah dengan nilai  $Q = 4.91$  s/d  $7.2$ , maka langkah ini dianggap cukup logis karena berurutan untuk sampai ketitik target. Adapun Namun, percobaan kesembilan dikategorikan sedang diantara 10 percobaan karena iterasinya cukup lama dengan langkah sedang. Adapun hasil perilaku robot kursi dapat dilihat pada Gambar 21 dibawah ini.



**Gambar 21 Hasil Perilaku Robot Kursi Versi Sedang.**

Adapun grafik hasil percobaan robot kursi versi sedang dapat dilihat pada Gambar 22.



**Gambar 22 Grafik Respon Percobaan Q-Learning Versi Sedang**

Berdasarkan Gambar 22, grafik respon percobaan Q learning dapat dijelaskan sebagai berikut:

a) Analisa respon tanggapan waktu:

- 1) Waktu Naik ( $t_r$ ) adalah waktu ketika keberadaan suatu respon mulai masuk 10% s/d 90% dari nilai titik targetnya. Grafik respon *Q-Learning* maka waktu untuk gelombang bergerak dari 10% s/d 90% dari nilai titik target pada robot kursi melakukan pergerakan saat nilai iterasi 13. Maka nilai waktu naik pada robot kursi adalah 13 iterasi.
- 2) Waktu Puncak ( $t_p$ ) adalah waktu ketika respon telah mencapai nilai puncak lewatan yang pertama kali. Grafik respon *Q-Learning* waktu puncak pada robot kursi terjadi saat nilai iterasi = 10.
- 3) Waktu Tetap ( $t_s$ ) adalah waktu ketika mencapai nilai akhir dari tanggapan dan tetap berada pada nilai tersebut dalam rentang persentase tertentu dari nilai akhir (5% atau 2%). Grafik respon *Q-Learning* maka waktu untuk tanggapan mencapai dan bertahan pada 2% dari nilai *steady state* (keadaan mantap) pada robot kursi dapat ditempuh saat nilai iterasi 16, jadi nilai waktu tetap robot kursi adalah 13 iterasi.
- 4) Waktu Tunda ( $t_d$ ) adalah waktu untuk mencapai setengah harga dari nilai akhir dari tanggapan untuk pertama kali. Grafik respon *Q-Learning* waktu tunda terjadi saat iterasi ke 9.

b) Respon Keadaan Mantap

Keadaan mantap adalah keadaan yang dimulai pada saat tanggapan mulai pertama kali mendekati nilai akhir hingga waktu yang tak terhingga. Pada percobaan pertama kursi berhasil mencapai titik target dengan nilai  $error = 0$  pada iterasi 13. Tanggapan dari proses iterasi 1 s/d iterasi 13 nilai  $error$  pada kursi adalah mendekati 0 (nol).

3. Hasil *Q-Learning* Versi Terbaik

Urutan langkah terbaik versi *q-learning* untuk penentu arah jejak kursi dari titik awal kotak warna hijau dan titik target kotak warna biru terdapat pada percobaan kelima. Urutan nilai Q yang dilewati ketika pembelajaran learning pada percobaan versi sedang dapat dilihat pada Gambar 23 berikut.



Gambar 23. Hasil Percobaan Versi Terbaik

- 1) Warna hijau
- 2) Warna kuning
- 3) Warna biru

Urutan langkah ini diperoleh dari urutan nilai Q terkecil hingga terbesar, yaitu sebagai berikut:

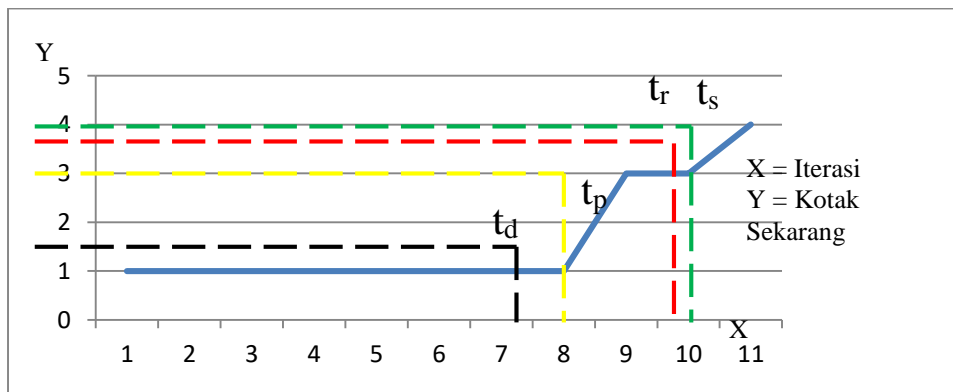
- 1) Hijau = 3.44
- 2) Kuning = 4.53
- 3) Biru = 5.13

Jika dilihat dari urutan langkah dengan nilai  $Q = 3.44$  s/d  $5.13$ , maka langkah ini logis karena berurutan untuk sampai ketitik target. Percobaan kelima adalah percobaan terbaik diantara 10 percobaan karena iterasi dan langkahnya sedikit. Adapun hasil perilaku robot kursi dapat dilihat pada Gambar 24 dibawah ini.



Gambar 24. Hasil Perilaku Robot Kursi Versi Terbaik

Adapun grafik hasil percobaan robot kursi versi sedang dapat dilihat pada Gambar 25 berikut.



Gambar 25. Grafik Respon Percobaan *Q-Learning*

Berdasarkan Gambar 25, grafik respon percobaan *Q-learning* dapat dijelaskan sebagai berikut:

- a) Analisa respon tanggapan waktu:
  - 1) Waktu Naik ( $t_r$ ) adalah waktu ketika keberadaan suatu respon mulai masuk 10% s/d 90% dari nilai titik targetnya. Grafik respon *Q-Learning* maka waktu untuk gelombang bergerak dari 10% s/d 90% dari nilai titik target pada robot kursi melakukan pergerakan saat nilai iterasi 9. Maka nilai waktu naik pada robot kursi adalah 9 iterasi.
  - 2) Waktu Puncak ( $t_p$ ) adalah waktu ketika respon telah mencapai nilai puncak lewatan yang pertama kali. Grafik respon *Q-Learning* waktu puncak pada robot kursi terjadi saat nilai iterasi = 7.
  - 3) Waktu Tetap ( $t_s$ ) adalah waktu ketika mencapai nilai akhir dari tanggapan dan tetap berada pada nilai tersebut dalam rentang persentase tertentu dari nilai akhir (5% atau 2%). Grafik respon *Q-Learning* maka waktu untuk tanggapan mencapai dan bertahan pada 2% dari nilai *steady state* (keadaan mantap) pada robot kursi dapat ditempuh saat nilai iterasi 9, jadi nilai waktu tetap robot kursi adalah 9 iterasi.
  - 4) Waktu Tunda ( $t_d$ ) adalah waktu untuk mencapai setengah harga dari nilai akhir dari tanggapan untuk pertama kali. Grafik respon *Q-Learning* waktu tunda terjadi saat iterasi ke 6.
- b) Respon Keadaan Mantap  
Keadaan mantap adalah keadaan yang dimulai pada saat tanggapan mulai pertama kali mendekati nilai akhir hingga waktu yang tak terhingga. Pada percobaan pertama kursi berhasil mencapai titik target dengan nilai *error* = 0 pada iterasi 9. Tanggapan dari proses iterasi 1 s/d iterasi 9 nilai *error* pada kursi adalah mendekati 0 (nol).

## PENUTUP

### Kesimpulan

Dari hasil penelitian yang telah dilakukan, maka dapat diperoleh beberapa kesimpulan, diantaranya:

1. Langkah minimal dalam mengimplementasikan algoritma *Q-learning* pada robot *soccer*, untuk penelusuran area dalam mencapai titik target dengan titik awal ungu dan titik target biru adalah langkah ungu-kuning-biru yang mendapatkan nilai Q 5.13, jumlah iterasi 9, nilai tanggapan waktunya yaitu *time peak* ( $t_p$ ) = iterasi 12, *time rise* ( $t_r$ ) = iterasi 8, *time settling* ( $t_s$ ) = iterasi 9, *time delay* = iterasi 6, dan nilai *error steady state* = 0.
2. Kursi bergerak otonom menggunakan metode algoritma *q-learning* mendapatkan nilai Q = 5.13 dan respon tanggapan waktu sebagai berikut: *time rise* ( $t_r$ ) = 9 iterasi, *time peak* ( $t_p$ ) = 7 iterasi, *time settling* ( $t_s$ ) = 9 iterasi, *delay time* ( $t_d$ ) = 6 iterasi dan nilai *error steady state* adalah nol (0) pada iterasi ke 9.
3. Berdasarkan data dari hasil penelitian, robot *soccer* memiliki perilaku kembali ke tempat semula (*home position*) lebih baik dibandingkan robot kursi bergerak otonom.

### Saran

Untuk kelanjutan pengembangan penelitian yang akan datang, beberapa saran yang dapat diberikan pada penelitian ini adalah:

1. Perlu peningkatan aksi gerak robot setelah nilai Q diketahui, sehingga mampu memperkecil pengulangan kondisi keberadaan robot di kotak yang sama.
2. Perbaikan pada aksi robot perlu dilakukan setelah nilai Q dihasilkan, sehingga dapat mengurangi *loop* kesalahan robot.

## REFERENSI

- [1] Kantori, M.T. dkk., 2017. Mobile Robot Pemain Bola : Aplikasi Pada KRSBI 2017. Sungailiat, Bangka Belitung : Politeknik Manufaktur Negeri Bangka Belitung.
- [2] Gitakarma, M.S. dan. G. Nurhayata. 2013. *Perancangan Behavior-Based Kendali Robot Dengan Algoritma Fuzzy Q-Learning (FQL) pada Sistem Navigasi Robot Otonom Beroda dalam Medan yang Tidak Terstruktur*. Singaraja: Universitas Pendidikan Ganesha.
- [3] Arifin, S, A.T. Hermawan. Dan. Y. Kristian. 2016. *Pencarian Rute Line Follower Mobile Robot Pada Maze Dengan Metode Q-Learning*. Surabaya: STMIK Asia Malang, STT Surabaya.

- [4] Andre, Ali, Zulfikar. 2016. *Sistem Security Webcam Dengan Menggunakan Microsoft Visual Basic (6.0)*. Panam-Pekanbaru : Jurnal Teknologi dan Sistem Informasi UNIVRAB.
- [5] Dinata Anda. 2017. *Physical Computing dengan Raspberry Pi*. Jakarta : PT Elex Media Komputindo.
- [6] Lestari Dyah dan Andrik Rizki Ari Wijaya. 2014. *Kontrol Arah Dan Kecepatan Motor DC Menggunakan Android*. Malang : Teknik Elektro Universitas Negeri Malang.
- [7] Kurniawan, Dwi Ely., dan Syafarudin Fani. 2017. *Perancangan Sistem Kamera Pengawas Berbasis Perangkat Bergerak Menggunakan Raspberry Pi*. Batam: Politeknik Negri Batam.
- [8] Wati, D.A.R. 2011. *Sistem Kendali Cerdas*. Edisi 1. Graha Ilmu. Yogyakarta: Indonesia.
- [9] Norman S, Nise. 2011. *Control System Engineering*. Sixth Edition. Pomona: California State Polytechnic University.
- [10] Putro, dkk.,. 2018. *Sistem Penggerak Robot Beroda Vacuum Cleaner Berbasis Mini Computer Raspberry Pi*. Manado : Jurusan Teknik Elektro-FT, UNSRAT.

*[halaman ini sengaja dikosongkan]*