

## SIMULASI GERAK ROBOT SOCCER BERODA MENGUNAKAN KINEMATIKA OMNIDIREKTIONAL RODA EMPAT

<sup>1</sup> Widi Saputro Purnomo, <sup>2</sup> Aryuanto Soetedjo  
<sup>1,2,3</sup> Teknik Elektro S1 ITN Malang, Malang, Indonesia  
<sup>1</sup>sapuwidi50@gmail.com, <sup>2</sup> aryuanto@lecturer.itn.ac.id

**Abstrak**— Kontes Robot Indonesia (KRI) adalah ajang kompetisi rancang bangun dan rekayasa dalam bidang robotika. KRI diselenggarakan oleh Pusat Prestasi Nasional, Kementerian Pendidikan dan Kebudayaan Republik Indonesia. Kontes Robot Indonesia ini dapat diikuti oleh oleh tim mahasiswa pada Perguruan Tinggi yang tercatat di Pangkalan Data Pendidikan Tinggi. Kontes Robot Sepak Bola Indonesia (KRSBI) merupakan metaforfosis dari Kontes Robot Cerdas Indonesia (KRCI) Robo Soccer Humanoid League (RSHL) yang terakhir kontes tahun 2012.

Untuk mendukung ajang perlombaan ini dilakukan pengembangan robot soccer beroda. Penelitian ini ditujukan untuk mengimplementasikan kinematika omni directional roda empat pada robot soccer beroda. Pada robot soccer beroda ITN Malang sebelumnya tidak menggunakan metode apapun, hanya saja pada tersebut menggunakan trial error, yaitu mencoba menebak arahnya dengan cara mengatur PWM kecepatan motor di setiap roda robot. Hal ini menyebabkan robot menjadi tidak dapat bergerak ke segala arah. Dengan mengimplementasikan kinematika omnidirectional dapat mengoptimalkan gerak robot beroda. Kelebihan dari robot four omnidirectional adalah dapat bergerak kesegala arah tanpa harus mengubah arah hadapnya. Arah gerak pada robot ini bergantung pada perbandingan kecepatan pada tiap roda yang didapat dari perhitungan kinematika robot.

**Kata Kunci:** *Omnidirectional, Kinematika, KRSBI, Robot, Mobile robot*

### I. PENDAHULUAN

Robot adalah seperangkat alat mekanik yang bisa melakukan kegiatan fisik dengan menggunakan program atau kecerdasan buatan yang telah di buat oleh manusia, ataupun dengan di awasi atau di control oleh manusia. Dari International Organization for Standardization (ISO 8373) mendefinisikan robot sebagai sebuah manipulator yang terkendali, multifungsi, dan mampu diprogram untuk bergerak dalam tiga aksis atau lebih, yang tetap berada di tempat atau bergerak untuk digunakan dalam aplikasi [otomasi](#) industri. Setelah mengalami perkembangan,

robot juga di gunakan untuk sarana pembelajaran. Untuk menunjukkan seberapa jauh pembelajaran tentang robot diadakan sebuah perlombaan robot di dunia. Trinity College Fire Fighting Home Robot Contest salah satu perlombaan robot internasional yang di ikuti beberapa negara seperti Indonesia. Ada pula yang di dalam negeri seperti Kontes Robot Indonesia. (Wikipedia, 2020)

Dalam perlombaan, pergerakan robot sangat menentukan kemenangan, pada umumnya robot memiliki pergerakan yang terbatas. Ada banyak bentuk roda salah satunya roda omnidirectional yang membantu atau membuat robot bisa bergerak kea rah yang di inginkan atau tak terbatas pergerakannya. Roda omnidirectional adalah salah satu bentuk robot yang memudahkan pergerakan robot dengan roda cakram kecil (disebut roller) di sekitar lingkaran yang tegak lurus terhadap arah belok. Efeknya adalah bahwa roda dapat digerakkan dengan kekuatan penuh, tetapi juga akan meluncur ke samping dengan sangat mudah. (Rafiuddin Syam, 2012)

Raifudin Syam (2012) menyatakan roda ini mempunyai kelebihan dari segi bentuknya yang simple, roda ini berbentuk seperti *disk* yang terpasang *roller* pada sisi luar. Robert L (2002) Model dinamis disajikan untuk mobile robot beroda *omnidirectional*, karena itu roda *omnidirectional* sering digunakan untuk robot yang membutuhkan pergerakan bebas. Tetapi roda ini sering tergelincir, seperti yang dijelaskan Ansu Man Singh (2016) Masalah lintasan tracking dari robot roda *omnidirectional*. Denta hariyahya (2016) menyatakan Di Indonesia sendiri roda ini sering dipakai salah satu divisi KRI (Kontes Robot Indonesia). Ada 3 beberapa *platform* roda yang dapat digunakan, diantaranya *platform* 3 roda yang membentuk huruf Y (segitiga), *platform* 4 roda yang membentuk huruf X (segi empat). Dalam pemanfaatannya terdapat juga *platform* 6 roda (segi enam), namun *platform* ini jarang digunakan karena boros sumber daya dan sulitnya konfigurasi. *Platform* 6 roda biasanya hanya dipakai untuk robot yang memiliki berat

yang besar, sehingga memerlukan pondasi yang kuat. (Hariyahya, 2016)

Untuk mendukung ajang perlombaan ini dilakukan pengembangan robot soccer beroda. Penelitian ini ditujukan untuk mengimplementasikan kinematika omni directional roda empat pada robot soccer beroda. Pada robot soccer beroda ITN Malang sebelumnya tidak menggunakan metode apapun, hanya saja pada tersebut menggunakan trial error, yaitu mencoba menebak arahnya dengan cara mengatur PWM kecepatan motor di setiap roda robot. Hal ini menyebabkan robot menjadi tidak dapat bergerak ke segala arah. Dengan mengimplementasikan kinematika omnidirectional dapat mengoptimalkan gerak robot beroda. Kelebihan dari robot four omnidirectional adalah dapat bergerak kesegala arah tanpa harus mengubah arah hadapnya. Arah gerak pada robot ini bergantung pada perbandingan kecepatan pada tiap roda yang didapat dari perhitungan kinematika robot.

Berdasarkan latar belakang tersebut, maka permasalahan yang muncul pada penelitian tersebut yaitu :

1. Bagaimana menggerakkan robot soccer beroda secara multiarah?
2. Bagaimana menghitung sudut menggunakan kinematika omnidirectional pada robot soccer beroda?
3. Seberapa efisien pergerakan robot yang menggunakan rumus kinematika?

Mensimulasikan gerak robot yang menggunakan kinematika omnidirectional roda empat di software CoppeliaSim Edu, dan menghitung sudut dengan menggunakan kinematika omnidirectional. Yang dimana bisa menggerakkan robot soccer secara multiarah. Sehingga bisa membandingkan dengan sistem trial error.

## II. KAJIAN PUSTAKA

### A. Robot

Robot berasal dari kata “*robot*” yang dalam bahasa Ceko yang berarti budak, pekerja atau kuli. Pertama kali kata “*robot*” diperkenalkan oleh Karel Capek dalam sebuah pentas sandiwaranya pada tahun 1921 yang berjudul RUR (Rossum’s Universal Robot). Pentas ini mengisahkan mesin yang menyerupai manusia yang dapat bekerja tanpa lelah yang kemudian memberontak dan menguasai manusia. Istilah “*robot*” ini kemudian mulai terkenal dan digunakan untuk menggantikan istilah yang dikenal saat itu yaitu *automation*. Dari berbagai literatur robot dapat didefinisikan sebagai sebuah alat mekanik yang dapat diprogram berdasarkan informasi dari lingkungan (melalui sensor) sehingga dapat melaksanakan beberapa tugas tertentu baik secara otomatis ataupun tidak sesuai program yang di inputkan berdasarkan logika. (Anggoro, Beni and Munadi, Dr. Eng. ST 2013)

Robot adalah seperangkat alat mekanik yang bisa melakukan kegiatan fisik dengan menggunakan program atau kecerdasan buatan yang telah di buat oleh manusia, ataupun dengan di awasi atau di control oleh manusia. Dari International Organization for Standardization (ISO 8373)

mendefinisikan robot sebagai sebuah manipulator yang terkendali, multifungsi, dan mampu diprogram untuk bergerak dalam tiga aksis atau lebih, yang tetap berada di tempat atau bergerak untuk digunakan dalam aplikasi otomasi industri. Setelah mengalami perkembangan, robot juga di gunakan untuk sarana pembelajaran. (Wikipedia, 2013)

### B. Mobile Robot

Mobile robot merupakan salah satu kategori robot, yang menunjukkan adanya fungsi untuk berpindah tempat. Pada awal perkembangannya, mobile robot menggunakan roda (wheel) untuk dapat berpindah tempat, namun saat ini telah dikembangkan robot yang dapat berpindah menggunakan kaki (leg), sehingga lebih dapat melewati area yang tidak dapat dilewati dengan roda. Mobile robot juga dapat diberikan berikan kecerdasan buatan, sehingga melalui informasi yang didapat dari sensor, maka robot dapat berjalan mengikuti garis dan menghindari halangan. Penelitian tentang robot cerdas pengikut garis yang juga dikembangkan oleh Juang-Huei Su et.al. (2010) yang ditujukan untuk tingkat pengenalan (pemula) dan hasil yang didapat adalah bahwa dengan menggunakan algoritma mean square errors, robot dapat melaju dengan kecepatan 1,3 m/detik. (Iqbal and Solekhan 2014)

Mobile robot ini banyak yang menyukai sehingga tidak sedikit yang mulai mempelajari robot. Hal ini dikarenakan mobile robot tidak memerlukan tenaga yang berat untuk membuatnya. Untuk membuat sebuah mobile robot minimal diperlukan pengetahuan tentang mikrokontroler dan sensor sensor elektronik. Base robot mobile bias terbuat dari yang paling mudah seperti triplek/plywood, akrilik sampai menggunakan logam aluminium. Mobile robot dapat di buat untuk mengikuti garis (line follower) atau mengikuti dinding (wall follower) ataupun mengikuti cahaya. Ada beberapa orang juga yang mengembangkan mobile robot dengan beberapa fungsi, diantaranya robot line follower, maze solving dan beberapa bentuk lain yang lebih unik lagi. Beberapa anak SMA pun juga tertarik dengan mobile robot sehingga di beberapa sekolah ada ekstrakurikuler robotic. Dengan adanya beberapa sekelompok anak yang mempelajarinya, ada perlombaan yang di adakan untuk menguji sejauh mana anak anak mampu untuk mengoperasikan mobile robot. (Wikipedia, 2012)



Gambar 1. Contoh Robot Mobile

### C. Omni Wheels

Roda omni telah digunakan bertahun-tahun dalam dunia robot industri dan logistik. Sumber utama pengguna terbanyak roda omni adalah perusahaan yang memproduksi untuk sistem konveyor, seperti untuk menangani paket/barang. Roda omni banyak juga digunakan untuk robot omni. Sebuah robot omni dapat berjalan lurus dari titik A ke titik B juga berputar agar dapat tiba di tujuannya. Roda omni juga digunakan untuk kursi roda, kendaraan servis di bandara dan lain-lain. Tidak banyak orang tahu bahwa roda omni directional pertama dipatenkan oleh J. Grobowiecki di Amerika pada tahun 1919. (Syam and Erlangga 2012)

Roda omni atau roda poli, mirip dengan roda Mecanum, adalah roda dengan cakram kecil (disebut roller) di sekitar lingkaran yang tegak lurus terhadap arah belok. Efeknya adalah bahwa roda dapat digerakkan dengan kekuatan penuh, tetapi juga akan meluncur ke samping dengan sangat mudah. Roda ini sering digunakan dalam sistem penggerak holonomis. (Wikipedia, 2020)

*Omnidirectional wheel* atau roda *omnidirectional* ini dapat berjalan ke segala arah dengan platform 4 roda (segi empat) atau 3 roda (segitiga). Di Indonesia sendiri roda ini sering dipakai salah satu divisi KRI (Kontes Robot Indonesia). Ada 3 beberapa platform roda yang dapat digunakan, diantaranya platform 3 roda yang membentuk huruf Y (segitiga), platform 4 roda yang membentuk huruf X (segi empat). Dalam pemanfaatannya terdapat juga platform 6 roda (segi enam), namun platform ini jarang digunakan karena boros sumber daya dan sulitnya konfigurasi. Platform 6 roda biasanya hanya dipakai untuk robot yang memiliki berat yang besar, sehingga memerlukan pondasi yang kuat. (Hariyahya 2016)



Gambar 2. Omni Wheels

### D. CoppeliaSim Edu

Simulator robot CoppeliaSim, dengan lingkungan pengembangan terintegrasi, didasarkan pada arsitektur kontrol terdistribusi: setiap objek / model dapat dikontrol secara individual melalui skrip tertanam, plugin, ROS atau BlueZero node, klien API jarak jauh, atau solusi kustom. Ini membuat CoppeliaSim sangat fleksibel dan ideal untuk aplikasi multi-robot. Pengontrol dapat ditulis dalam C / C ++, Python, Java, Lua, Matlab atau Octave.

CoppeliaSim digunakan untuk pengembangan algoritma cepat, simulasi otomatisasi pabrik, prototyping dan verifikasi

cepat, pendidikan terkait robotika, pemantauan jarak jauh, pemeriksaan keselamatan, sebagai kembar digital, dan banyak lagi. Anda dapat menemukan gambaran umum fitur di sini.

Robot, robotika, simulator, simulasi, kinematika, dinamika, perencanaan jalur, penghitungan jarak minimum, deteksi tabrakan, sensor penglihatan, pemrosesan gambar, sensor jarak, pengeluaran cat. (Freese, n.d.)



Gambar 1. Logo CoppeliaSim Edu

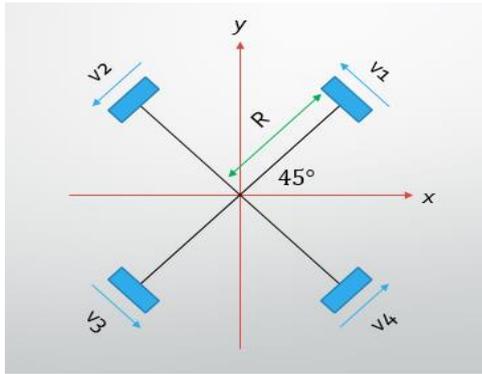
### E. Kinematika

*Invers kinematik* mengacu pada penggunaan persamaan kinematika robot untuk menentukan parameter bersama yang memberikan posisi yang diinginkan pada posisi akhir atau efektor. Spesifikasi pergerakan robot sehingga posisi akhir atau efektor mencapai titik yang diinginkan adalah dikenal sebagai perencanaan gerak. *Invers kinematik* mengubah rencana gerak menjadi nilai yang harus diberikan bagi aktuator atau penggerak dalam pergerakan robot.

Dalam pergerakannya, robot dimodelkan dalam bentuk persamaan kinematika. Persamaan ini menentukan konfigurasi robot dalam hal parameter untuk setiap aktuator. *Forward kinematika* menggunakan parameter untuk menghitung konfigurasi robot, dan *invers kinematika* membalikkan perhitungan ini untuk menentukan parameter bersama dalam mencapai konfigurasi yang diinginkan.

*Invers kinematik* merupakan suatu metode analisa untuk melakukan transformasi dari ruang Cartesian ke ruang sendi. Dari persamaan kinematik, dapat diperoleh hubungan antar konsep geometri ruang sendi pada robot dengan konsep koordinat yang biasa digunakan untuk menentukan kedudukan suatu objek. Dengan model kinematik, programmer dapat menentukan konfigurasi referensi masukan yang harus diumpangkan kepada masing-masing aktuator agar robot dapat melakukan gerakan secara simultan untuk mencapai posisi yang dikehendaki. (Kusuma, Shinta, and Dedy 2015)

Robot soccer beroda akan menerima arah sudut gerak dari kamera (yang tidak di bahas dalam penelitian ini). Sudut yang diterima tadi akan di hitung menggunakan kinematika roda empat yang nantinya akan menghasilkan kecepatan pada setiap motor  $v_1, v_2, v_3, v_4$ .



Gambar 4. Ilustrasi robot soccer beroda 4

Setiap motor akan menghasilkan gaya (F) dengan arah seperti berikut. Setiap R pada robot menghasilkan sudut yang sama terhadap sumbu x. Pengaturan kecepatan (v) masing-masing motor menentukan sudut pergerakan robot. Arah putar motor sesuai dengan arah gaya (F). Dari formasi ini, maka dihasilkan operasi matrix seperti berikut.

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} -\sin(45) & \cos(45) & R \\ -\sin(45) & -\cos(45) & R \\ \sin(45) & -\cos(45) & R \\ \sin(45) & \cos(45) & R \end{bmatrix} \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \\ \omega \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} -0.707 & 0.707 & R \\ -0.707 & -0.707 & R \\ 0.707 & -0.707 & R \\ 0.707 & 0.707 & R \end{bmatrix} \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \\ 0 \end{bmatrix}$$

Dari setiap slide sebelumnya, dapat disimpulkan perhitungan kecepatan masing – masing motor adalah sebagai berikut.

$$\begin{aligned} v_1 &= (\cos(\theta) \cdot -0.707) + (\sin(\theta) \cdot 0.707) \\ v_2 &= (\cos(\theta) \cdot -0.707) + (\sin(\theta) \cdot -0.707) \\ v_3 &= (\cos(\theta) \cdot 0.707) + (\sin(\theta) \cdot -0.707) \\ v_4 &= (\cos(\theta) \cdot 0.707) + (\sin(\theta) \cdot 0.707) \end{aligned}$$

$\theta$  merupakan sudut gerak robot yang diinginkan  
 $\omega$  merupakan gerak angular. Karena kita tidak memerlukannya, maka nilainya adalah 0.

R merupakan jarak antara roda dengan titik tengah robot

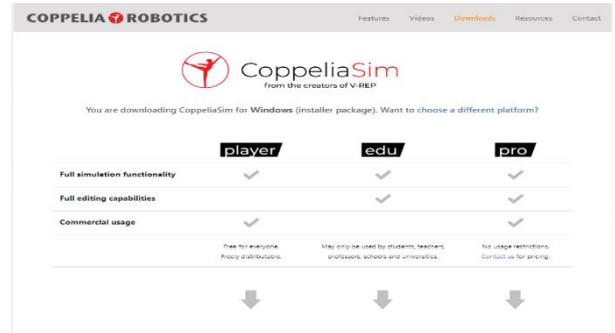
- $v_1$  merupakan kecepatan roda robot 1
- $v_2$  merupakan kecepatan roda robot 2
- $v_3$  merupakan kecepatan roda robot 3
- $v_4$  merupakan kecepatan roda robot 4

### III. METODOLOGI PENELITIAN

#### A. Perancangan Simulasi

Pada perancangan simulasi ini, ada langkah yang akan dilakukan, mendownload dan menginstall software CoppeliaSim Edu. Berikut adalah langkah langkahnya :

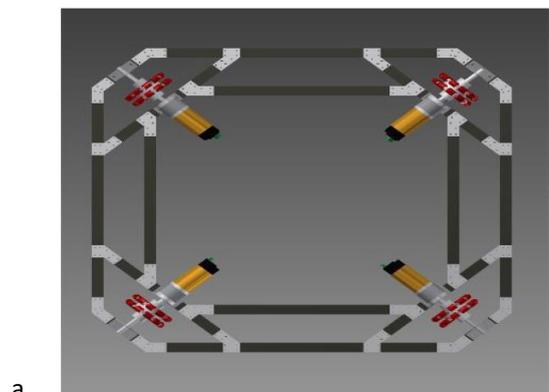
1. Mendownload software CoppeliaSim Edu di web <https://www.coppeliarobotics.com/downloads>. Seperti yang terlihat pada gambar 3.2 adalah tampilan dari link download CoppeliaSim Edu.



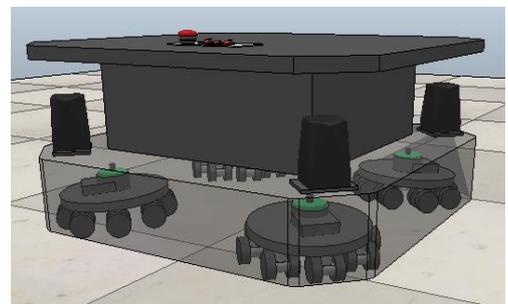
Gambar 5. Tampilan link download CoppeliaSim Edu

#### B. Rancangan Robot Simulasi

Pada simulasi ini desain robot yang digunakan berbeda dengan yang aslinya bisa terlihat pada gambar 6.



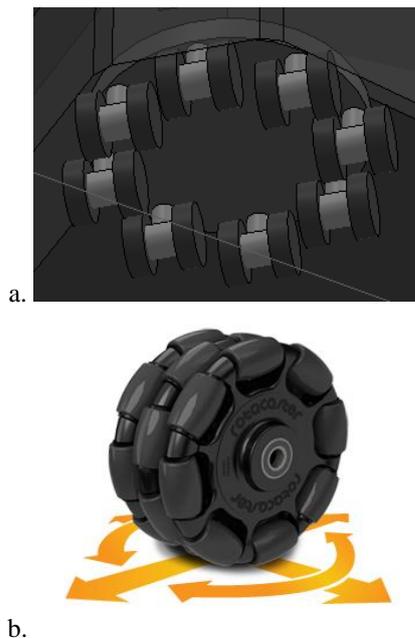
a.



b.

Gambar 6. (a) adalah ilustrasi robot asli (b) adalah ilustrasi robot yang digunakan simulasi

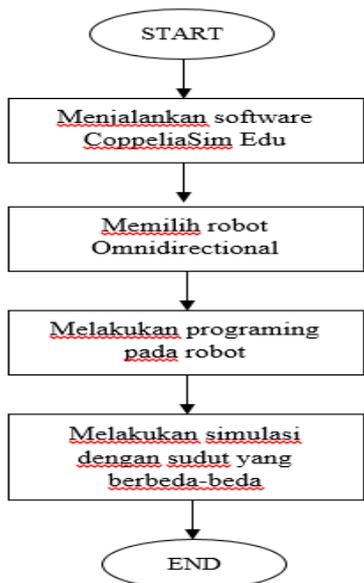
Roda yang digunakan juga berbeda dengan yang aslinya, dengan jenis roda yang berbeda sedikit memengaruhi hasil pada simulasi ini. Perbedaan jenis roda bisa terlihat pada gambar 7.



Gambar 7. (a) adalah roda yang di gunakan simulasi (b) adalah roda yang asli

### C. Cara Kerja Simulasi

Perancangan simulasi robot soccer beroda ini dapat dijalankan dengan menginstall software CoppeliaSim Edu. Berikut flowchart dalam menjalankan simulasi :

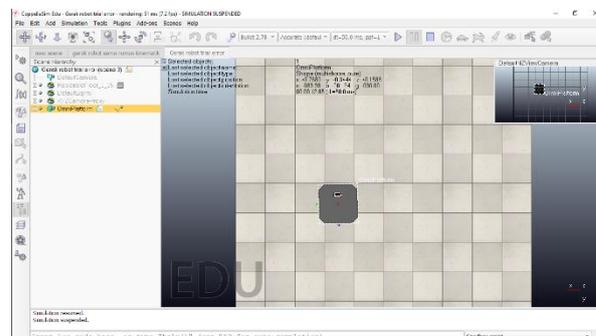


Gambar 8. Flowchart Simulasi Robot

## IV. SIMULASI DAN ANALISA

### A. Pengujian Simulasi

Pengujian simulasi robot ini bertujuan untuk mengetahui bagaimana kinerja dari perancangan sistem yang telah di buat dan dapat mengetahui keakuratan sudut yang diberikan maka bisa dilihat dari koordinat x dan y terakhir robot bergerak. Dan variabel b menunjukkan orientasi robot.



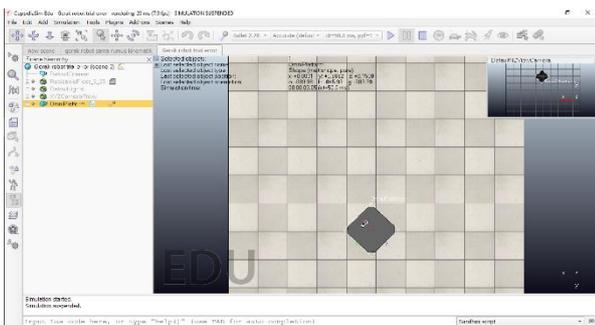
Gambar 9. Tampilan simulasi di software CoppeliaSim Edu

Apakah sistem yang dibuat bisa menggerakkan robot kearah yang di tentukan. Hasil dari pengujian simulasi sistem ini bisa di lihat pada agambar dibawah ini:

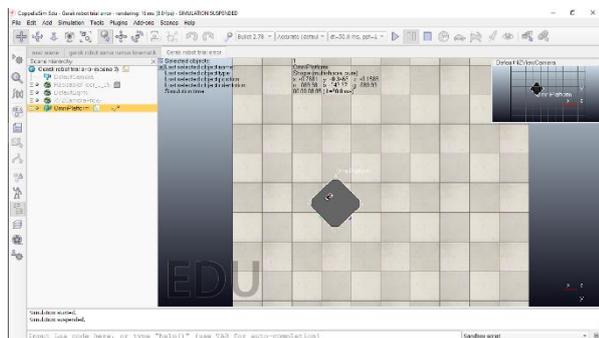
Pada gambar 9 menjelaskan tampilan dari simulasi di software CoppeliaSim Edu. Gambar 9 menunjukkan ada robot dengan nama 'OmniPlatform' yang akan melakukan simulasi dari sistem kinematika dan trial error. Pada saat robot tersebut bergerak ada perubahan posisi yang di tampilkan pada sumbu x, y, dan z. Sumbu tersebut yang akan menunjukkan koordinat pergerakan pada robot. Untuk menunjukkan orientasi robot ialah sumbu a, b, dan g. Ketika sumbu a, b, dan g mengalami perubahan maka akan menunjukkan perubahan orientasi robot contoh ketika sumbu b mengalami perubahan menunjukkan arah hadap robot. Tanda '-' dan '+' menunjukkan arah sesuai arah gerak jarum jam, jika + menunjukkan arah gerak searah jarum jam, jika - menunjukkan arah gerak berlawanan arah jarum jam. Icon kertas di sebelah tulisan OmniPlatform menunjukkan icon untuk memasukkan program pada robot simulasi.

### A. Simulasi menggunakan Trial Error

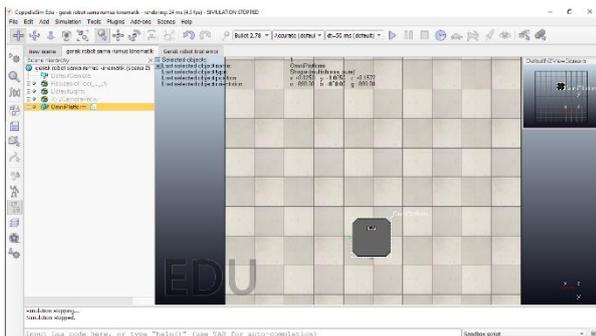
Pada simulasi menggunakan trial error robot akan di program dulu sesuai arah yang akan di inginkan, kemudian simulasi robot dijalankan. Pertama tama robot mengalami perubahan arah hadap robot dulu selanjutnya akan melaju sesuai program yang di inginkan, kemudian robot akan mengalami perubahan arah hadapnya ke posisi awal. Perubahan arah hadap robot pada trial error terpaut oleh waktu yang di tentukan pada saat programing.



Gambar 10. Tampilan robot mengalami perubahan arah hadap



Gambar 11. Tampilan robot berjalan sesuai program



Gambar 12. Tampilan robot mengalami perubahan arah hadap lagi

Pada simulasi menggunakan trial error dilakukan beberapa kali dengan sudut yang berbeda beda, berikut adalah tabel yang menunjukkan hasil dari beberapa simulasi di sudut yang berbeda:

Tabel 1. Data Hasil Perubahan Koordinat Robot Menggunakan Trial Error

$(\theta)$	Posisi Awal Robot			Posisi Setelah Simulasi			Waktu yang dibutuhkan
	x	y	b	x	y	b	
$0^\circ$	-0.0500	1.0250	0	-1.1554	0.9900	-2.71	12 s
$45^\circ$	-0.0500	1.0250	0	-0.8076	0.1435	-1.93	11 s
$90^\circ$	-0.0500	1.0250	0	-0.0067	-0.0064	1.50	10 s
$135^\circ$	-0.0500	1.0250	0	0.7680	0.3346	-1.65	11 s
$180^\circ$	-0.0500	1.0250	0	1.0737	1.1030	-0.98	12 s

### B. Analisa pengujian simulasi trial error

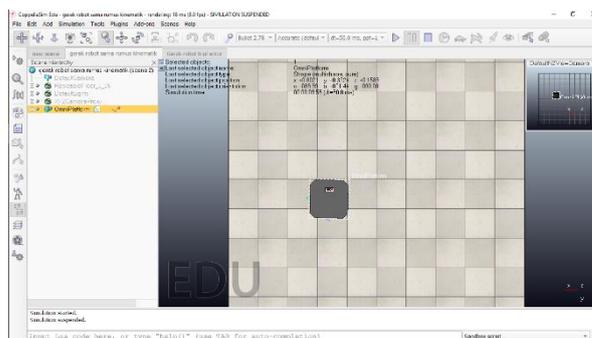
Terlihat pada gambar 10, gambar 11, dan gambar 12 adalah gambar perubahan gerak robot yang menggunakan trial error. Pada gambar 10 menunjukkan robot berubah arah hadapnya menuju sudut yang di inginkan. Kemudian pada gambar 11 robot mengalami perpindahan gerak sampai pada titik koordinat yang ditentukan. Gambar 12 menunjukkan robot mengalami perubahan arah hadap ke posisi awal mula.

Tabel 1 menunjukkan data hasil percobaan pada simulasi trial error. Pada simulasi ini ada perbedaan waktu tempuh pada setiap perubahan sudut yang di inginkan. Perbedaan waktu ini di pengaruhi pada saat robot mengalami perubahan arah hadap di awal simulasi menuju sudut yang di inginkan dan di akhir ketika robot telah berada di koordinat yang di tentukan.

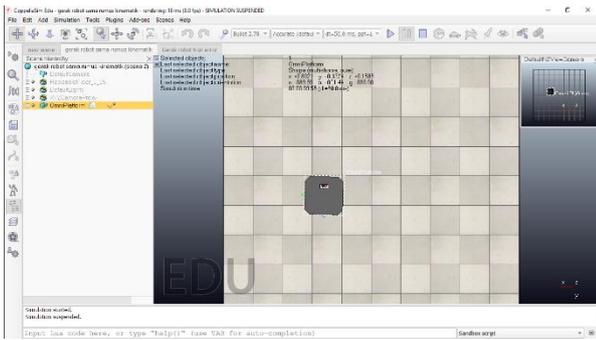
Pada simulasi menggunakan trial error, pemrograman dilakukan berulang ulang untuk mendapatkan sudut dan koordinat yang diinginkan.

### C. Simulasi menggunakan kinematika

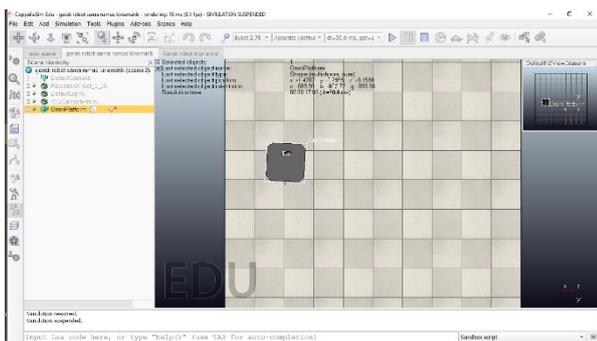
Pada simulasi menggunakan kinematika robot akan di program untuk menuju ke sudut yang telah di tentukan, setelah itu simulasinya di jalankan robot akan mengalami pergerakan tanpa mengubah arah hadap robotnya sampai simulasi di hentikan.



Gambar 13. Tampilan robot aw



Gambar 14. Tampilan robot berjalan sesuai program



Gambar 15. Tampilan robot terus berjalan sampai simulasi di hentikan

Pada simulasi menggunakan kinematika dilakukan beberapa kali dengan sudut yang berbeda beda, berikut adalah tabel yang menunjukkan hasil dari beberapa simulasi di sudut yang berbeda:

Tabel 1. Data Hasil Perubahan Koordinat Robot Menggunakan Kinematika

D. Analisa simulasi menggunakan kinematika

Terlihat pada gambar 13, gambar 14, dan gambar 15 adalah gambar perubahan gerak pada robot yang menggunakan kinematika. Pada gambar 13 menunjukan posisi awal robot. Kemudian pada gambar 14 bergerak menuju sudut yang di tentukan. Gambar 15 menunjukan robot telah berhenti pada koordinat yang telah di tentukan.

Tabel 2 menunjukkan data hasil percobaan pada simulasi menggunakan kinematika. Pada simulasi ini tidak ada perbedaan waktu tempuh pada setiap perubahan sudut yang di inginkan. Perbedaan waktu ini hanya terjadi pada saat robot mengalami perubahan arah hadap di awal simulasi menuju sudut yang di inginkan dan di akhir ketika robot telah berada di koordinat yang di tentukan. Sedangkan pada simulasi menggunakan kinematika robot tidak mengalami perubahan hadapnya, sehingga tidak terjadi perbedaan waktu tempuh robot menuju koordinat yang ditentukan. Contoh robot berada di titik koordinat x -0.0500 dan y 1.0250 menuju titik koordinat x -0.8074 dan y 0.2748. Dan mengalami perubahan arah hadap b 2.20.

Pada simulasi menggunakan kinematika, pemrograman dilakukan hanya dengan memasukkan sudut yang ingin dituju dengan begitu robot langsung berjalan menuju sudut dan koordinat yang di tentukan. Dari hasil perhitungan kinematika di atas juga bisa dilihat ada  $v_1, v_2, v_3, dan v_4$  adalah kecepatan pada setiap motor robot. Untuk V ini merupakan level kecepatan yang diinginkan pada jalannya robot. Sudut yang telah di tentukan akan dihitung seperti diatas kemudian akan dikalikan dengan level kecepatan yang diinginkan sehingga mendapatkan besar kecepatan pada setiap motor.

V. KESIMPULAN

Setelah dilakukan perancangan, pengujian, analisa simulasi, dan pengambilan data hasil, maka dapat disimpulkan beberapa hal yang dapat digunakan untuk perbaikan dan pengembangan selanjutnya, yaitu :

1. Dengan simulasi ini bisa di lihat bahwa menggunakan kinematic pergerakan pada robot lebih cepet bergerak dari pada robot yang menggunakan trial error. Bisa dilihat perbedaan tabel 1 dan tabel 2. Waktu yang dibutuhkan untuk mencapai sudut  $0^\circ$  dengan menggunakan metode trial error adalah 12 detik sedangkan waktu yang dibutuhkan untuk mencapai sudut  $0^\circ$  dengan menggunakan metode kinematika adalah 10 detik.
2. Pada simulasi yang dilakukan mengetahui pergerakan robot dengan menggunakan kinematika memanfaatkan fungsi roda omnidirectional.
3. Simulasi ini juga menunjukkan bahwa pergerakan robot yang menggunakan metode trial error tidak seefisien robot yang menggunakan kinematika. Terlihat pada gambar 4.2 robot yang menggunakan trial error memutar

$(\theta)$	Posisi Awal Robot			Posisi Setelah Simulasi			Waktu yang dibutuhkan
	x	y	b	x	y	b	
$0^\circ$	-0.0500	1.0250	0	-1.1598	0.9913	1.77	10 s
$45^\circ$	-0.0500	1.0250	0	-0.8074	0.2748	2.20	10 s
$90^\circ$	-0.0500	1.0250	0	-0.0067	-0.0064	1.50	10 s
$135^\circ$	-0.0500	1.0250	0	0.7716	0.2916	1.50	10 s
$180^\circ$	-0.0500	1.0250	0	1.0613	1.0849	1.62	10 s

- arah hadapnya kesudut yang di tentukan terlebih dahulu.
4. Pada simulasi ini ada perubahan orientasi pada robot, hal ini di sebabkan keterbatasan komponen pada software tersebut. Yang mana perubahan tersebut memengaruhi keakuratan berjalannya robot.
  5. Robot menggunakan trial error harus mengulangi beberapa pemrograman untuk mencapai sudut yang diinginkan.

#### Saran

Setelah dilakukan perancangan , pengujian, analisa simulasi, dan pengambilan data hasil, maka ada beberapa saran yang dapat digunakan untuk perbaikan dan pengembangan selanjutnya, yaitu :

Pada penelitian selanjutnya bisa langsung diterapkan pada robot beroda ITN malang. Robot yang terdapat pada simulasi ini terbatas karena desain bawaan dari software CoppeliaSim Edu untuk mengetahui keakuratan pada robot sesungguhnya bisa di lakukan penerapan metode kinematika omnidirectional pada robot beroda ITN Malang.

## VI. DAFTAR PUSTAKA

- [1] Anggoro, Beni and Munadi, Dr. Eng. ST, MT. 2013. "Desain Pemodelan Kinematik Dan Dinamik Humanoid Robot." 101(3): 505–7.
- [2] Hariyahya, Denta. 2016. "Prototype Kursi Roda Dengan Penggerak Roda Omnidirectional Berbasis Arduino." : 1–24.
- [3] Iqbal, Mohammad, and Solekhan Solekhan. 2014. "Perancangan Wheel Mobile Robot Sebagai Modul Praktikum Mikrokontoler." *Simetris : Jurnal Teknik Mesin, Elektro dan Ilmu Komputer* 4(1): 59.
- [4] Kusuma, Johan Wijaya, P Shinta, and H Dedy. 2015. "Pergerakan Kaki Pada Robot Hexapod." *Robotics*: 10.
- [5] Syam, Rafiuddin, and Widhi Erlangga. 2012. "Rancang Bangun Omni Wheels Robot Dengan Roda Penggerak Independent." 3(1): 213–20.
- [6] Widodo, Mutijarsa, 2017, Rancang Bangun Sistem Lokomosi, Penggiring, dan Penendang pada Robot Sepak Bola, Institut Teknologi Bandung
- [7] F. Muir, P- Neuijtan, 1986, Kinematic Modeling of Wheeled Mobile Robots, Carnegie-Mellon University
- [8] P.Oliviera, J. Sousa, dkk, Modeling and Assessing of Omni-directional Robots with Three and Four Wheels, Universidade do porto