



Pengembangan Simulator Robot SCARA pada Platform PyBullet

Tugas Akhir

Oleh:

Amrina Rosyada (4222101038)

Pengembangan Simulator Robot SCARA pada Platform PyBullet

Amrina Rosyada
Department of Electrical Engineering, Politeknik Negeri Batam

ABSTRAK

Abstrak – Revolusi Industri 5.0 membawa perubahan signifikan dalam struktur dan pendekatan industri modern, yang ditandai dengan munculnya era Society 5.0, di mana terjadi sinergi antara manusia dan kecerdasan buatan (Artificial Intelligence/AI). Berbeda dengan era sebelumnya yang berfokus pada otomatisasi penuh, Industri 5.0 menekankan pentingnya integrasi teknologi yang selaras dengan nilai-nilai kemanusiaan untuk meningkatkan kualitas hidup dan produktivitas. Dalam konteks ini, teknologi robotika berperan penting dalam mendukung pelaksanaan tugas-tugas repetitif, presisi tinggi, dan berulang secara efisien, serta mengurangi beban kerja manusia dalam lingkungan industri.

Salah satu bentuk implementasi teknologi ini adalah robot SCARA (Selective Compliance Assembly Robot Arm), yang banyak digunakan dalam proses pick and place untuk menyortir atau memindahkan objek berdasarkan atribut tertentu, seperti warna atau bentuk. Salah satu tantangan yang dihadapi dalam sistem ini adalah proses bin picking, yaitu pengambilan objek secara otomatis dari wadah yang tidak terstruktur. Permasalahan ini menuntut desain sistem yang mampu mengenali dan merespons kondisi lingkungan kerja yang dinamis.

Penelitian ini bertujuan untuk mengembangkan simulator robot SCARA menggunakan mekanisme five-bar linkage pada platform PyBullet. Melalui simulasi ini, potensi kesalahan dalam desain mekanik maupun sistem kendali dapat diidentifikasi dan diperbaiki sebelum dilakukan implementasi fisik. Platform PyBullet dipilih karena kemampuannya dalam memvisualisasikan gerakan secara real-time, mendukung format URDF, serta menyediakan antarmuka interaktif yang memudahkan evaluasi dinamika robot. Simulasi ini diharapkan menjadi landasan awal dalam pengembangan sistem robotik yang lebih efisien, teruji, dan siap diintegrasikan dalam lingkungan kerja industri yang sebenarnya.

Kata Kunci: Industri 5.0, Robot SCARA, URDF, Simulasi, PyBullet.

ABSTRACT

The Industrial Revolution 5.0 brings significant changes to the structure and approach of modern industry, marked by the emergence of the Society 5.0 era, where there is synergy between humans and artificial intelligence (AI). Unlike previous eras that focused on full automation, Industry 5.0 emphasizes the importance of integrating technology that aligns with human values to improve quality of life and productivity. In this context, robotics technology plays a crucial role in supporting the efficient execution of repetitive, high-precision, and routine tasks, thereby reducing the workload on humans in industrial environments.

One form of this technology's implementation is the SCARA robot (Selective

Compliance Assembly Robot Arm), widely used in pick-and-place processes to sort or move objects based on specific attributes, such as color or shape. One challenge in this system is bin picking, the automatic retrieval of objects from unstructured containers. This issue requires a system design capable of recognizing and responding to dynamic work environment conditions.

This research aims to develop a SCARA robot simulator using a five-bar linkage mechanism on the PyBullet platform. Through this simulation, potential errors in mechanical design and control systems can be identified and corrected before physical implementation. The PyBullet platform was chosen for its ability to visualize movements in real-time, support the URDF format, and provide an interactive interface that facilitates the evaluation of robot dynamics. This simulation is expected to serve as a foundation for the development of more efficient, tested, and ready-to-integrate robotic systems in real industrial work environments. |

Keywords: Industry 5.0, SCARA robots, URDF, Simulation, PyBullet

Pendahuluan

Revolusi Industri 5.0 hadir sebagai kelanjutan dari Revolusi Industri 4.0, fokus utama pada kolaborasi antara manusia dan teknologi dalam menciptakan solusi yang lebih efisien (Mubarokah and Wibawa, 2022) . Pada Industri 5.0 tidak hanya berfokus pada penggunaan teknologi untuk memaksimalkan produktivitas dan akurasi, tetapi untuk keuntungan bersama antara manusia dan teknologi. Sasaran Industri 5.0 saat ini memiliki fokus untuk membagikan informasi yang cepat antar manusia dan robot, sistem yang cepat beradaptasi dengan perubahan dan memanfaatkan AI (*Artificial Intelligence*) untuk menambah kemampuan manusia (Zafar et al., 2024) .

Di era industri modern, otomatisasi menjadi elemen penting dalam meningkatkan efisiensi, presisi, dan kecepatan produksi. Salah satu teknologi yang banyak diterapkan adalah robotika, yang memungkinkan penggantian proses manual dengan sistem yang terprogram dan terkontrol. Penggunaan robot dalam industri mempermudah pelaksanaan tugas-tugas kompleks seperti perakitan, pengelasan, dan pengemasan dengan akurasi tinggi. Dengan dukungan sensor, aktuator, dan mikrokontroler, sistem gerak robot dapat dirancang untuk menghasilkan pergerakan yang fleksibel dan sinkron, sehingga mampu menyesuaikan diri dengan kebutuhan lini produksi secara dinamis dan efektif. (Nurussa'adah et al., 2020) . Berdasarkan penelitian ini (Satria Wijaya et al., 2024) Robot memiliki keunggulan dalam bekerja secara berulang tanpa henti, sehingga mampu meningkatkan efisiensi dalam proses produksi. Kemampuan ini menjadikan robot, khususnya robot manipulator, sangat ideal untuk diterapkan di industri. Selain mengurangi ketergantungan pada tenaga manusia, robot juga mampu bekerja dengan stabil dan presisi dalam jangka waktu yang panjang, menjadikannya solusi yang efektif dalam otomasi tugas- tugas seperti *pick and place*.

Contoh implementasi sistem *bin picking* dapat ditemukan pada penggunaan robot SCARA (*Selectively Compliant Assembly Robot Arm*) dalam industri manufaktur. Teknologi ini memungkinkan robot untuk mengambil objek dari wadah yang berisi berbagai komponen yang tersusun secara acak. Dengan kemampuan tersebut, robot dapat meningkatkan efisiensi proses perakitan dan mengurangi ketergantungan pada pekerjaan manual, terutama dalam lingkungan produksi yang dinamis dan tidak terstruktur. (Olesen et al., 2020).

Penelitian sebelumnya menunjukkan bahwa simulasi robot lengan 3 DOF (*Degrees of Freedom*) mampu meningkatkan efektivitas kerja serta mengurangi kesalahan dalam pergerakan robot secara signifikan, dengan tingkat akurasi yang cukup tinggi dalam proses pemindahan objek di ruang

kerja (Uchrowi et al., 2019). Berdasarkan temuan tersebut, penelitian ini secara khusus bertujuan untuk mensimulasikan pergerakan sumbu X, Y, dan Z pada robot SCARA dengan memanfaatkan platform PyBullet.

PyBullet merupakan pustaka simulasi fisika *multi-body open-source* yang mendukung pemodelan dan visualisasi dinamika robot secara *real-time*. Pustaka ini memungkinkan pemuatan model robot dalam format URDF (*Universal Robot Description Format*), serta dilengkapi dengan *solver inverse kinematics* bawaan yang digunakan untuk menghitung konfigurasi sudut sendi berdasarkan posisi dan orientasi *end-effector* yang diinginkan (Baumgärtner et al., 2023).

Melalui simulasi ini, diharapkan diperoleh pemahaman awal mengenai dinamika pergerakan robot SCARA pada sumbu X, Y, dan Z. Fokus utama penelitian ini adalah mengevaluasi *response* sistem terhadap perintah gerakan translasi pada masing-masing sumbu, serta menilai kestabilan dan akurasi pergerakan *end-effector* dalam melaksanakan tugas-tugas dasar di ruang kerja tiga dimensi.

Penelitian ini bertujuan untuk merancang dan mengimplementasikan simulator robot SCARA berbasis mekanisme *five-bar linkage* (*lima batang penggerak simetris*) dengan menggunakan platform PyBullet. Simulator yang dikembangkan ditujukan untuk mengevaluasi pergerakan *end-effector* secara visual dan numerik dalam ruang kerja tiga dimensi (X, Y, Z). Selain itu, penelitian ini juga bertujuan untuk mengidentifikasi keterbatasan konfigurasi kinematika serta memastikan bahwa simulasi dapat dimanfaatkan sebagai landasan awal dalam proses evaluasi sebelum dilakukan pembuatan fisik robot.

Metode

Simulasi dalam penelitian ini diimplementasikan menggunakan PyBullet, sebuah pustaka Python berbasis *Bullet Physics Engine* yang dirancang untuk keperluan simulasi fisika, kontrol robotik, serta aplikasi pembelajaran mesin. PyBullet menyediakan berbagai fitur seperti pemuatan model robot dari file URDF, perhitungan dinamika maju (*forward dynamics*), dinamika balik (*inverse dynamics*), perhitungan kinematika, serta deteksi tumbukan dan rendering visual secara *real-time* (Coumans Erwin and Bai Yunfei, 2023).

Model robot SCARA yang telah dirancang dimasukkan ke dalam lingkungan simulasi PyBullet untuk dievaluasi secara interaktif. Struktur mekanik robot ini menggunakan konfigurasi *five-bar linkage*, yang memungkinkan dua derajat kebebasan di bidang horizontal melalui dua lengan

aktif, serta satu derajat kebebasan tambahan berupa translasi pada sumbu vertikal. Fokus utama simulasi adalah mengamati pergerakan *end-effector* pada sumbu X, Y, dan Z serta menganalisis keterkaitan antara sudut sendi dan posisi ruang gerak.

Pengendalian gerakan dilakukan melalui antarmuka visual berbasis *slider* yang memungkinkan pengguna mengatur posisi masing-masing sendi dan tinggi vertikal *end-effector* secara manual. Selain itu, sistem juga mendukung pendekatan berbasis perhitungan kinematika untuk menentukan konfigurasi sendi secara otomatis berdasarkan posisi *end-effector* yang diinginkan. Dengan pendekatan ini, validasi terhadap model kinematika dan dinamika dapat dilakukan secara efisien sebelum berlanjut ke tahap implementasi fisik.

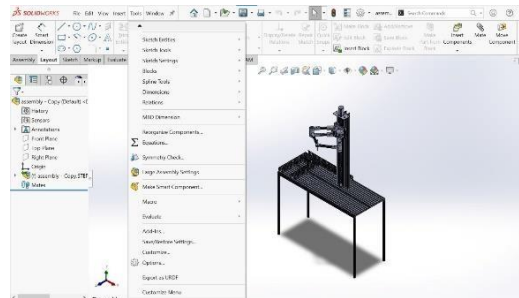


Gambar 1. Desain Robot SCARA

Gambar 1 memperlihatkan desain awal robot SCARA yang dibuat menggunakan perangkat lunak SolidWorks. Model ini menampilkan struktur mekanik lima batang simetris (*five-bar linkage*), lengkap dengan sambungan vertikal dan lengan rotasi sebagai elemen utama. Desain geometri 3D ini selanjutnya dikonversi ke format URDF untuk keperluan simulasi menggunakan plugin URDF Exporter, yang dapat diunduh melalui [plugin SW2URDF](#)

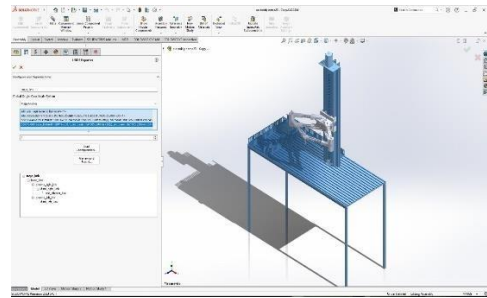
Plugin URDF Exporter mempermudah konversi model CAD (*Computer-Aided Design*) dari SolidWorks ke format URDF untuk simulasi di PyBullet. Proses ini mencakup aktivasi *plugin*, pemilihan *link* (penghubung antar dua bagian) dan *joint* (sendi), serta pengaturan parameter fisik. Hasil ekspor berupa file .urdf dan folder meshes/ yang berisi file geometri .stl. Ilustrasi di bawah ini menunjukkan urutan proses konversi model CAD robot

SCARA menjadi format URDF menggunakan plugin *URDF Exporter*. Langkah-langkahnya adalah sebagai berikut:



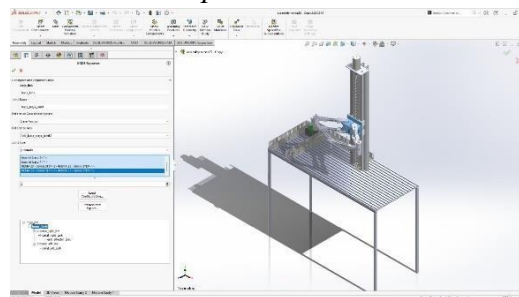
Langkah 1.

Aktifkan *Plugin* > *Tools* > *Add-ins* >
Buka Model > file robot SCARA >
Export to URDF



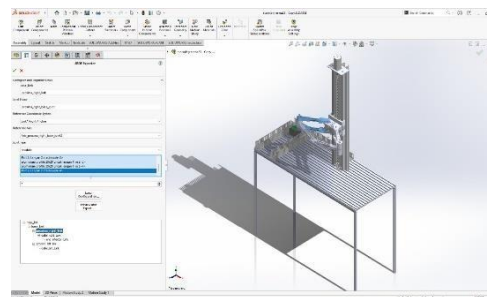
Langkah 2.1

Atur *Link & Joint Meja*



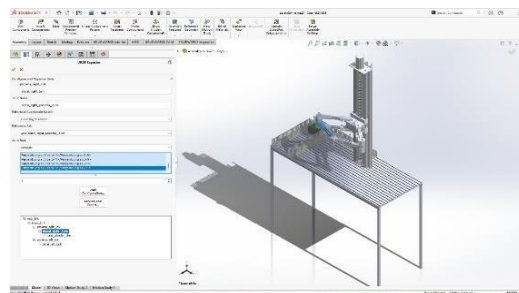
Langkah 2.2

Atur *Link & Joint Base*



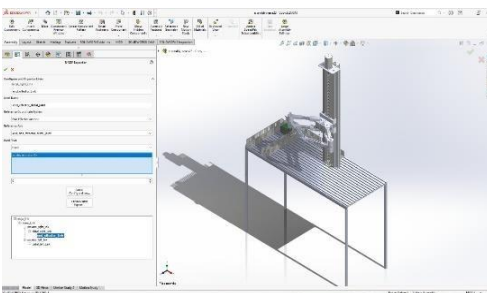
Langkah 2.3

Atur *Link & Joint proxima*



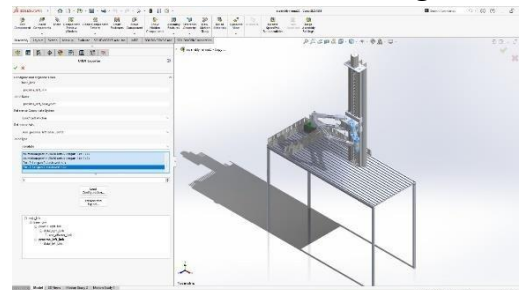
Langkah 2.4

Atur *Link & Joint distal right*



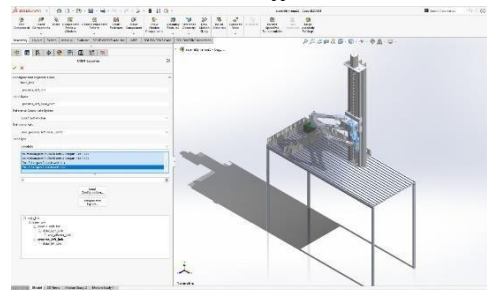
Langkah 2.5

Atur *End-Effector*



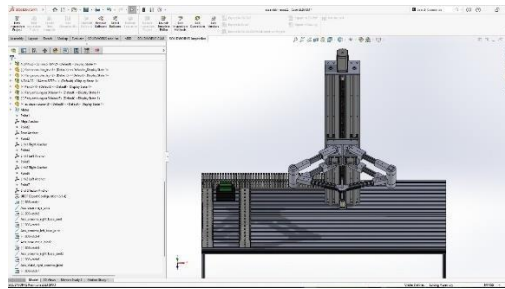
Langkah 2.6

Atur *Link & Joint proxima left*



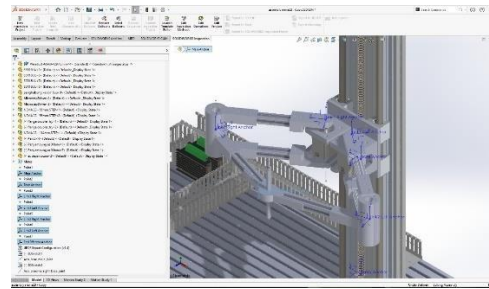
Langkah 2.7

Atur *Link & Joint distal left*



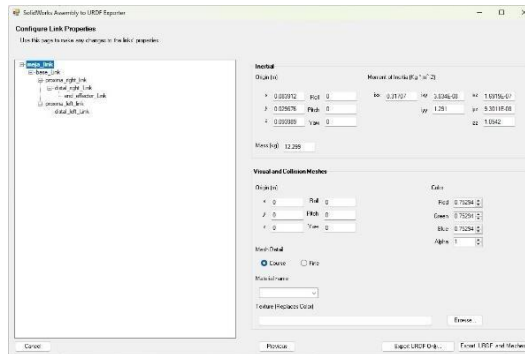
Langkah 3.

Pastikan semua *link* & *joint* sudah ada



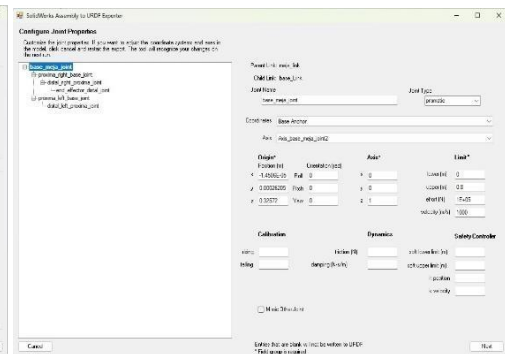
Langkah 4.

Pastikan titik *link* & *joint* benar



Langkah 5.

Konfigurasi *link* properti



Langkah 6.

Konfigurasi *joint* properti

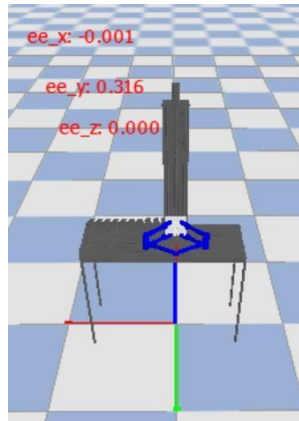
Gambar 2. Langkah-langkah ekspor model dari SolidWorks ke URDF menggunakan *plugin URDF Exporter*.

Gambar 2 menjelaskan bahwa setiap komponen model disimpan dalam bentuk file mesh .stl yang kemudian dirujuk dalam file URDF untuk mewakili struktur fisik robot. Format URDF berbasis XML ini memuat informasi penting seperti *link*, *joint*, *massa*, *inersia*, serta hubungan antar bagian secara struktural.

Struktur akhir model terdiri dari dua *joint revolute*, satu *joint prismatic* untuk mendukung pergerakan pada sumbu X, Y, dan Z, serta beberapa *link* berbentuk geometris dasar. Pendekatan ini memastikan bahwa integrasi model ke dalam simulator PyBullet dapat dilakukan dengan akurat dan efisien, sebagaimana juga diterapkan dalam pengembangan simulasi robot SCARA dan *humanoid* pada penelitian sebelumnya (Fadlillah et al., 2023; Yang et al., 2017).

Dalam simulasi robot SCARA berbasis *five-bar linkage*, motor stepper NEMA 17 digunakan untuk sumbu X dan Y, sementara NEMA 34 untuk sumbu Z. Seluruh pergerakan dikendalikan oleh PLC (*Program Logic Control*) dengan program logika terstruktur. Metode kontrol kecepatan bertingkat digunakan untuk menjaga akurasi posisi dan kestabilan gerak, mengacu pada (Cahyo Wibowo et al., 2023), serta didukung oleh (Ismaranatasia et al., 2021) yang membuktikan efektivitas pengendalian motor stepper berbasis PLC dengan

pengaturan frekuensi secara bertahap pada sistem bergerak multi-sumbu.

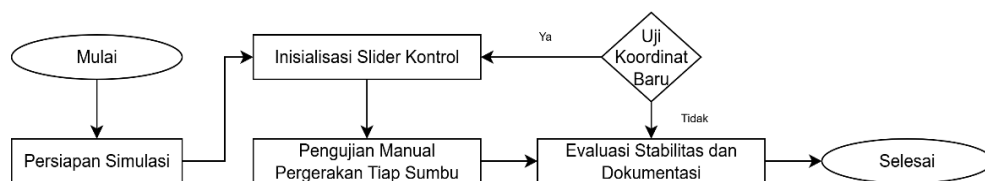


Gambar 3. Visualisasi arah sumbu koordinat dalam simulasi PyBullet

- Sumbu X (Merah): Horizontal mengarah ke kanan–kiri
- Sumbu Y (Hijau): Longitudinal mengarah ke depan–belakang
- Sumbu Z (Biru): Vertikal mengarah atas–bawah

Ketiga sumbu ini mengikuti konvensi tangan kanan (*right-hand rule*) dan menjadi dasar dalam interpretasi posisi *end-effector* di dalam ruang kerja tiga dimensi (Coumans Erwin and Bai Yunfei, 2023)

Validasi dilakukan dengan membandingkan posisi *end-effector* aktual dengan hasil perhitungan berdasarkan parameter geometri robot. Hasil menunjukkan hubungan yang konsisten dan linier antara perubahan sudut sendi dan perpindahan posisi dalam ruang kerja. Ini membuktikan bahwa model bekerja dengan akurat dan stabil sesuai prinsip kinematika maju.



Gambar 4. Proses Simulasi

Gambar 4 yang menggambarkan langkah-langkah pengujian pergerakan robot SCARA pada platform PyBullet. Evaluasi gerak robot SCARA dilakukan dengan menganalisis respons sistem terhadap variasi *input* sudut pada dua aktuator utama. Gerakan pada bidang X dan Y dikendalikan melalui *rotasi* dua lengan utama yang masing-masing terhubung ke sendi pasif, membentuk struktur manipulasi lima batang simetris (*five-bar planar manipulator*). Untuk memperoleh posisi *end-effector* secara akurat, digunakan pendekatan kinematika maju, di mana posisi kartesian (x , y) dihitung

berdasarkan sudut aktuator θ_{a1} dan θ_{a2} . Persamaan kinematika maju diturunkan dari model geometris tertutup dan dinyatakan sebagai berikut:

$$x = l_1 \cos(\theta_{a1}) + l_2 \cos(\theta_{p1}), y = l_1 \sin(\theta_{a1}) + l_2 \sin(\theta_{p1})$$

Sudut pasif θ_{p1} dihitung terlebih dahulu berdasarkan parameter geometri sebagai:

$$m = l_0 + l_1 \cos(\theta_{a2}) - l_1 \cos(\theta_{a1}), \quad n = l_1 \sin(\theta_{a2}) - l_1 \sin(\theta_{a1})$$

$$\theta = \cos^{-1} \left(\frac{m^2 + n^2}{2l_2^2} \right), \varphi = \tan^{-1} \left(\frac{n}{m} \right), \theta_{p1} = \varphi + \emptyset$$

Untuk mempermudah pemahaman alur komputasi posisi *end-effector* berdasarkan masukan sudut, berikut disajikan *pseudocode* algoritma yang digunakan dalam simulasi. Algoritma ini menghitung kinematika maju pada mekanisme *five-bar* untuk memperoleh posisi *end-effector* berdasarkan sudut dua sendi aktif serta parameter panjang lengan.

1. *function five_bar_forward_kinematics*(θ_{a1}, θ_{a2})
2. : $\theta_{a1}, \theta_{a2} \rightarrow$ Active joint angles (in radians)
 $l_0, l_1, l_2 \rightarrow$ Link parameters (constants)
3. $m \leftarrow 2 \cdot l_0 + l_1 \cdot (\cos(\theta_{a2}) - \cos(\theta_{a1}))$
4. $n \leftarrow l_1 \cdot (\sin(\theta_{a2}) - \sin(\theta_{a1}))$
5. $d \leftarrow \sqrt{m^2 + n^2}$
6. if $d \geq 2 \cdot l_2$ then
 return (None, None)
7. $\varphi_1 \leftarrow \arccos(d / (2 \cdot l_2))$
8. $\varphi_2 \leftarrow \text{atan2}(n, m)$
9. $\theta_{p1} \leftarrow \varphi_1 + \varphi_2$
10. $x \leftarrow -l_0 + l_1 \cdot \cos(\theta_{a1}) + l_2 \cdot \cos(\theta_{p1})$
11. $y \leftarrow l_1 \cdot \sin(\theta_{a1}) + l_2 \cdot \sin(\theta_{p1})$
12. return (x, y)

Dalam proses kinematika balik, posisi target dari *end-effector* digunakan untuk menentukan sudut yang harus dicapai oleh kedua aktuator aktif. Berdasarkan posisi (x, y), algoritma menghitung sudut referensi terhadap basis kiri dan kanan mekanisme, kemudian digunakan dalam perhitungan geometri berbasis hukum kosinus untuk mendapatkan sudut aktual. Apabila hasil perhitungan menunjukkan posisi berada di luar jangkauan mekanisme (ditandai dengan nilai kosinus yang tidak valid), maka algoritma

mengembalikan nilai kosong. Hasil akhir berupa dua sudut aktuasi yang dapat digunakan dalam simulasi atau pengendalian sistem.

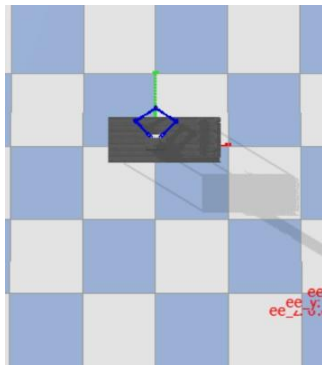
1. *function five_bar_inverse_kinematics(x,y)*
2. *input: x,y → Target end – effector position*
 l₀, l₁, l₂ → Link parameters (constants)
3. $\beta_1 \leftarrow \text{atan2}(y, l_0 + x)$
4. $\beta_2 \leftarrow \text{atan2}(y, l_0 - x)$
5. $\alpha_{1_calc} \leftarrow (l_1^2 + (l_0 + x)^2 + y^2 - l_2^2) / (2 \cdot l_1 \cdot \sqrt{((l_0 + x)^2 + y^2)})$
6. $\alpha_{2_calc} \leftarrow (l_1^2 + (l_0 - x)^2 + y^2 - l_2^2) / (2 \cdot l_1 \cdot \sqrt{((l_0 - x)^2 + y^2)})$
7. *if $|\alpha_{1_calc}| > 1$ or $|\alpha_{2_calc}| > 1$ then*
 return None → Coordinates unreachable
8. $\alpha_1 \leftarrow \arccos(\alpha_{1_calc})$
9. $\alpha_2 \leftarrow \arccos(\alpha_{2_calc})$
10. $\theta_{a1} \leftarrow \beta_1 + \alpha_1$ *→ Left active joint*
11. $\theta_{a2} \leftarrow \pi - \beta_2 - \alpha_2$ *→ Right active joint*
12. *return (θ_{a1}, θ_{a2})*

Metode ini mengacu pada model kinematika dari manipulator simetris lima batang seperti dijelaskan oleh (Le et al., 2013), untuk mengevaluasi posisi *end-effector* secara *real-time* selama simulasi. Validasi dilakukan dengan membandingkan hasil visual simulasi dan perhitungan teoritis berdasarkan panjang lengan dan sudut rotasi. Hasil menunjukkan hubungan linier dan konsisten antara perubahan sudut aktuator dan perpindahan *end-effector*, yang menandakan implementasi kinematika maju berjalan stabil dan akurat secara geometris.

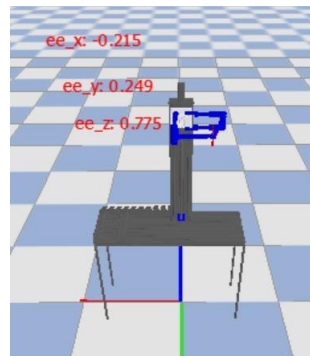
Pernyataan ini diperkuat oleh (Zhao et al., 2021) yang menjelaskan bahwa keterbatasan pencatatan data numerik dalam simulasi merupakan tantangan umum dalam pendekatan *deep reinforcement learning*, terutama karena dapat membatasi validasi presisi saat model diterapkan ke sistem nyata (*sim-to-real transfer*). Pendekatan validasi berbasis visual ini juga sejalan dengan hasil yang diperoleh oleh (Uchrowi et al., 2019) yang memodelkan robot lengan 3 DOF dan menerapkan metode *forward kinematics* untuk menggerakkan *end-effector* dari satu posisi ke posisi lain dengan rata-rata kesalahan di bawah 6%. Hasil ini menunjukkan bahwa validasi berbasis simulasi mampu memberikan gambaran akurat sebelum implementasi fisik dilakukan. |

Hasil dan Pembahasan

Penelitian ini menghasilkan simulasi gerak pada sumbu X, Y, dan Z dari robot SCARA linkage *five-bar mekanisme* menggunakan platform PyBullet. Simulasi ini memvisualisasikan pergerakan *end-effector* secara *real-time*, dengan dua sendi aktif yang masing-masing dapat berputar hingga 90° , serta satu gerakan translasi vertikal. Setiap sumbu dikendalikan melalui tiga *slider* GUI (*Graphical User Interface*) terpisah, sehingga memungkinkan evaluasi hubungan antara input sudut dan posisi spasial berdasarkan kinematika maju secara visual maupun numerik.



Gambar 5. Tampilan Awal Robot SCARA dalam Simulasi



Gambar 6. Tampilan Robot SCARA saat Bergerak dalam Simulasi

Gambar 5 Menampilkan tampilan awal lingkungan simulasi PyBullet saat model robot SCARA pertama kali dimuat. Pada Gambar 6 Perubahan sudut *joint* melalui *slider* menyebabkan perpindahan posisi *end-effector*, yang ditampilkan secara *real-time* untuk memverifikasi hasil gerakan berdasarkan *input*.

Selama simulasi, posisi *end-effector* ditampilkan dalam koordinat X, Y, dan Z, dengan perubahan posisi yang umumnya konsisten terhadap sudut masing-masing sendi, menunjukkan bahwa sistem kinematika maju bekerja dengan baik. Namun, pada kombinasi sudut tertentu, posisi tidak muncul karena konfigurasi berada di luar ruang kerja atau tidak valid secara kinematik. Visualisasi jangkauan memperlihatkan bahwa koordinat X dan Y dipengaruhi oleh dua sendi utama, sementara posisi Z berubah secara linier. Distribusi titik membentuk pola menyerupai kurva sabit, dengan beberapa area yang tidak dapat dijangkau. Hasil ini membantu mengidentifikasi batas ruang kerja dan posisi optimal *end-effector*, serta memperkuat validasi sistem berdasarkan hubungan geometris antar parameter.

No	Joint 1 (°)	Joint 2 (°)	Joint 3 (m)	Posisi End-Effector (X, Y, Z)
1	92	5	0.000	(0.204 m, 0.269 m, 0.000 m)
2	178	88	0.700	(-0.213 m, 0.251 m, 0.700 m)
3	95	85	0.750	(-0.008 m, 0.509 m, 0.750 m)
4	172	12	0.050	(-0.006 m, 0.157 m, 0.050 m)
5	100	25	0.200	(0.153 m, 0.344 m, 0.200 m)
6	160	80	0.700	(-0.169 m, 0.317 m, 0.700 m)
7	110	8	0.100	(0.135 m, 0.249 m, 0.100 m)
8	168	70	0.600	(-0.133 m, 0.261 m, 0.600 m)
9	94	77	0.650	(0.018 m, 0.502 m, 0.650 m)
10	180	15	0.150	(-0.017 m, 0.137 m, 0.150 m)
11	106	36	0.300	(0.110 m, 0.375 m, 0.300 m)
12	150	68	0.600	(-0.106 m, 0.336 m, 0.600 m)
13	99	14	0.200	(0.176 m, 0.300 m, 0.200 m)
14	176	84	0.750	(-0.195 m, 0.254 m, 0.750 m)
15	91	90	0.800	(-0.013 m, 0.514 m, 0.800 m)
16	165	20	0.400	(-0.008 m, 0.156 m, 0.400 m)
17	103	55	0.450	(0.077 m, 0.450 m, 0.450 m)
18	143	66	0.550	(-0.087 m, 0.356 m, 0.550 m)
19	97	32	0.250	(0.152 m, 0.379 m, 0.250 m)
20	170	10	0.050	(-0.000 m, 0.158 m, 0.050 m)

Tabel 1. Posisi End-Effector

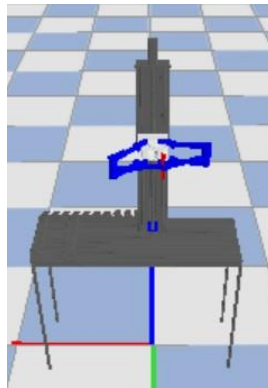
Data hasil simulasi dirangkum dalam Tabel 1 sebagai dasar analisis dan evaluasi lebih lanjut. perubahan sudut pada *Joint 1* dan *Joint 2* berpengaruh signifikan terhadap koordinat X dan Y *end-effector*, sedangkan *Joint 3* memengaruhi posisi pada sumbu Z. Hal ini menunjukkan bahwa implementasi kinematika maju telah berjalan dengan baik sesuai model teoritis.

Namun, pada kombinasi tertentu khususnya saat *Joint 1* berada di rentang 140° – 180° dan *Joint 2* bernilai kecil (0° – 30°) simulasi tidak menghasilkan posisi *end-effector*. Ini menunjukkan bahwa konfigurasi tersebut berada di luar ruang kerja atau tidak valid secara kinematik. Kemungkinan penyebabnya meliputi:

- *End-effector* berada di luar jangkauan,
- Terjadi *self-collision* antar bagian robot,
- Konfigurasi sendi menghasilkan posisi yang tidak mungkin.

Hal ini bukan kesalahan sistem, melainkan batas alami dari struktur robot SCARA. Gambar berikut memperlihatkan kondisi saat posisi *end-effector* tidak

terbaca oleh sistem:



Gambar 7. Posisi *End-Effector* tidak terbaca

Gambar 7 Kondisi robot SCARA saat konfigurasi sudut menyebabkan posisi *end-effector* tidak terbaca dalam simulasi. Terlihat bahwa lengan robot berada dalam posisi ekstrem yang mengakibatkan *end-effector* berada di luar jangkauan ruang kerja atau mengalami konflik fisik antar bagian.

Respons dinamika dari robot terhadap *input slider* menunjukkan kinerja yang halus dan stabil. Tidak ditemukan gejala seperti getaran, osilasi, ataupun keterlambatan respons yang signifikan. Hal ini menunjukkan bahwa sistem kendali motor virtual dalam PyBullet berhasil merepresentasikan perilaku dinamis dasar dari robot SCARA secara realistis. Namun, (Zhao et al., 2021) menekankan pentingnya validasi berbasis metrik numerik dan pengujian terhadap skenario gangguan dunia nyata untuk memastikan bahwa hasil simulasi tidak hanya baik secara visual, tetapi juga tahan terhadap variasi kondisi lingkungan dalam penerapan sebenarnya.

Perbandingan secara kuantitatif terhadap hasil perhitungan teoritis belum dilakukan secara eksplisit dalam penelitian ini. Namun demikian, posisi *end-effector* yang dihasilkan oleh simulasi telah menggunakan rumus kinematika maju berdasarkan parameter panjang lengan dan sudut aktif. Dengan demikian, dapat disimpulkan bahwa hasil simulasi secara umum sesuai dengan nilai yang diperkirakan oleh model matematis.

Apabila dilakukan pencocokan lebih lanjut terhadap koordinat hasil simulasi dan perhitungan manual, besar kemungkinan akan ditemukan kesesuaian dalam nilai posisi *end-effector*, terutama karena perhitungan tersebut telah disesuaikan secara geometri dengan struktur *five-bar linkage*.

Penggunaan PyBullet sebagai platform simulasi dalam penelitian ini menawarkan sejumlah keunggulan, antara lain antarmuka *slider* GUI yang

mempermudah pengaturan gerakan secara manual, visualisasi posisi *end-effector* secara *real-time*, serta performa simulasi yang cepat dan stabil meskipun dijalankan pada perangkat keras dengan spesifikasi terbatas. Fitur-fitur ini memudahkan pengguna untuk melakukan pengujian gerakan robot secara interaktif, mirip dengan konsep simulator berbasis *OpenGL* yang dikembangkan oleh Balza et al. (Achmad et al., 2008) yang memungkinkan kontrol visual tiga dimensi dan pengaturan sudut sendi baik secara manual maupun melalui skenario otomatis.

Di samping itu, PyBullet juga mendukung perluasan fungsionalitas melalui pengembangan pustaka eksternal seperti *PyBullet Industrial*, yang memungkinkan integrasi antara simulasi robot dan proses manufaktur secara simultan. Kemampuan ini mencakup simulasi gaya pemrosesan, perubahan alat (*tool changing*), serta pemrograman lintasan berbasis *G-code*, yang menjadikan PyBullet lebih adaptif terhadap kebutuhan industri (Baumgärtner et al., 2023)

Meski demikian, pendekatan ini tetap memiliki keterbatasan, seperti tidak tersedianya pencatatan data numerik secara otomatis, evaluasi posisi yang masih bersifat visual, serta belum adanya sistem analisis kesalahan seperti RMSE (*Root Mean Square Error*). Selain itu, simulasi yang dilakukan dalam lingkungan virtual sering kali tidak merepresentasikan kondisi fisik nyata secara utuh, khususnya dalam skenario *closed-loop control* yang memerlukan respons waktu nyata dan akurasi tinggi. Ketidaksesuaian ini dikenal sebagai *sim-to-real gap*, yaitu perbedaan antara dinamika simulasi dan dunia nyata yang dapat menurunkan efektivitas model saat diterapkan ke sistem fisik (Zhao et al., 2021). Lebih lanjut, pustaka *PyBullet Industrial* sendiri masih mengandalkan parameterisasi manual dalam membangun skenario, sehingga belum sepenuhnya efisien untuk skala simulasi kompleks atau otonom (Baumgärtner et al., 2023).

Kesimpulan

Penelitian ini berhasil mengembangkan simulasi gerak robot SCARA berbasis struktur *five-bar linkage* pada platform PyBullet, yang mampu memvisualisasikan pergerakan *end-effector* pada tiga sumbu (X, Y, Z) secara *real-time*. Hasil simulasi menunjukkan bahwa perubahan sudut pada dua *joint rotasi* secara langsung memengaruhi posisi *end-effector* di bidang X dan Y, sedangkan translasi pada *joint* vertikal mengubah posisi pada sumbu Z. Validasi visual menunjukkan hubungan linier dan konsisten antara *input* sudut dan posisi spasial, yang menandakan bahwa model kinematika maju telah diimplementasikan dengan baik (Le et al., 2013; Uchrowi et al., 2019)

Namun demikian, pada kombinasi sudut tertentu khususnya saat *Joint*

1 berada di rentang 140° – 180° dan *Joint* 2 memiliki nilai kecil (0° – 30°) posisi *end-effector* tidak terbaca dalam simulasi. Hal ini menunjukkan bahwa konfigurasi tersebut berada di luar ruang kerja atau tidak valid secara kinematik karena kemungkinan *self-collision* atau keterbatasan jangkauan struktural lengan robot.

Platform PyBullet terbukti efektif dalam menyediakan lingkungan simulasi interaktif dengan kontrol manual berbasis GUI *slider*, visualisasi posisi secara *real-time*, serta performa yang responsif dan stabil, bahkan pada perangkat keras terbatas. Fitur-fitur ini mendukung proses evaluasi desain dan pengujian algoritma kinematika secara efisien dan intuitif. Namun, keterbatasan seperti tidak tersedianya pencatatan data numerik secara otomatis dan belum adanya sistem evaluasi berbasis metrik kesalahan seperti RMSE menjadi catatan penting untuk pengembangan ke depan.

Untuk mengatasi keterbatasan tersebut, direkomendasikan pengembangan fitur pencatatan data otomatis, integrasi kontrol berbasis serta menganalisis keterkaitan antara sudut sendi dan posisi ruang gerak kinematics penuh, serta adopsi pendekatan *sim-to-real transfer* seperti *domain randomization* dan *domain adaptation* sebagaimana dijelaskan oleh (Zhao et al., 2021). Dengan demikian, simulator ini dapat lebih mendekati kondisi implementasi nyata dan meningkatkan keandalan model saat diterapkan pada sistem fisik. Secara keseluruhan, simulator ini memiliki potensi besar sebagai landasan awal dalam proses desain, validasi, dan pembelajaran robot SCARA secara virtual.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Politeknik Negeri Batam yang telah mendanai penelitian ini melalui skema *Project Based Learning* (PBL) pada Program Studi Diploma 4 Teknik Robotika.

Kontribusi Penulis

Penulis bertugas mengembangkan program simulasi menggunakan PyBullet, mengonversi desain robot SCARA dari SolidWorks ke format URDF, serta melakukan pengujian simulasi untuk memperoleh data posisi *end-effector* berdasarkan variasi konfigurasi sudut pada masing-masing joint.

Daftar Pustaka

- Achmad, B., Sunaryo, M., Arif, A., 2008. SIMULATOR LENGAN ROBOT ENAM DERAJAT KEBEBASAN MENGGUNAKAN OPENGL.
- Baumgärtner, J., Hansjosten, M., Schönhofen, D., Fleischer, Prof.Dr.-Ing.J., 2023. PyBullet Industrial: A process-aware robot simulation. *J Open Source Softw* 8, 5174.
- Cahyo Wibowo, B., Dahlan, M., Setiawan, R.R., 2023. STEPPER MOTOR CONTROL ANALYSIS USING ACCELERATION-DECELERATION METHOD BASED ON PLC (PROGRAMMABLE LOGIC CONTROL), *Media Elektrik Journal*.
- Coumans Erwin, Bai Yunfei, 2023. *PyBullet Quickstart Guide*.
- Fadlillah, R., Rusdinar, A., Siregar, S., 2023. Implementasi Inverse Kinematics Untuk Pola Jalan Robot Penari Humanoid.
- Ismaranatasia, W., Setiawan, B., Subiyantoro, 2021. Kendali Motor Stepper untuk Pergerakan Sumbu X, Y, Z pada 3D.
- Le, T.D., Kang, H.J., Doan, Q.V., 2013. A method for optimal kinematic design of five-bar planar parallel manipulators, in: 2013 International Conference on Control, Automation and Information Sciences, ICCAIS 2013. IEEE Computer Society, pp. 7–11.
- Mubarokah, M.G., Wibawa, A.P., 2022. Kolaborasi Artificial Intelligence dalam Peningkatan Kebutuhan Bisnis di Era Society 5.0 2, 576–580.
- Nurussa'adah, Mudjirahardjo, P., Kurnia, S.A., Auliya, N., 2020. Robot movement system design based on the recorded movement of a cloning robot. *International Journal of Mechanical Engineering and Robotics Research* 9, 93– 98.
- Olesen, A.S., Gergaly, B.B., Ryberg, E.A., Thomsen, M.R., Chrysostomou, D., 2020. A collaborative robot cell for random bin-picking based on deep learning policies and a multi-gripper switching strategy, in: *Procedia Manufacturing*. Elsevier B.V., pp. 3–10.
- Satria Wijaya, R., Pratama, A., Fatekha, R.A., Soebhakti, H., Prayoga, S., 2024. Visual-Based Pick and Place on 6 DoF Robot Manipulator, *JOURNAL OF APPLIED ELECTRICAL ENGINEERING*.
- Uchrowi, A., Lasmadi, L., Soekarno, S., 2019. Pemodelan dan Simulasi Robot Lengan 3 DOF Menggunakan V-REP. *AVITEC* 1.

Yang, Y., Zhuang, Z., Xu, R., 2017. Construction of SCARA robot simulation platform based on ROS. *International Core Journal of Engineering* 3.

Zafar, M.H., Langås, E.F., Sanfilippo, F., 2024. Exploring the synergies between collaborative robotics, digital twins, augmentation, and industry 5.0 for smart manufacturing: A state-of-the-art review. *Robot Comput Integr Manuf.*

Zhao, W., Queralta, J.P., Westerlund, T., 2021. Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: a Survey.