

# OPTIMASI LINTASAN MODEL LENGAN ROBOT PUMA560 MENGUNAKAN ALGORITMA GENETIK

Temon Wahyu P<sup>1</sup>, Wahyudi<sup>2</sup>, Aris Triwiyatno<sup>2</sup>

**Abstrak** -- Puma560 merupakan salah satu jenis robot yang banyak digunakan dalam dunia industri karena termasuk jenis robot yang mempunyai kemampuan gerak yang cepat dan fleksibilitas yang tinggi. Kemampuan kerja robot ini sangat dipengaruhi oleh lintasan (*path*) lengan-lengannya. Lintasan yang baik dapat dilihat dari nilai *kinematic manipulability* yang besar dan kurva *manipulability* yang halus. Salah satu metode penentuan lintasan yang sering digunakan adalah metode *Joint Interpolated Trajectory*, dimana lintasan ditentukan dengan mencari nilai sudut masing-masing joint dalam fungsi waktu. Akan tetapi metode ini belum mampu menghasilkan nilai *kinematic manipulability* yang besar sehingga lintasan yang dihasilkan kurang optimal.

Algoritma Genetik sebagai salah satu metode optimasi yang berdasarkan evolusi alam digunakan untuk menentukan lintasan model lengan robot Puma560. Pada tugas akhir ini, optimasi lintasan ditentukan dengan melakukan optimasi *kinematic manipulability*. Metode Algoritma Genetik yang digunakan adalah “*Simple Genetic Algorithm*”, dimana pengkodean dilakukan terhadap parameter percepatan awal dan akhir masing-masing joint model lengan robot Puma560.

Metode algoritma genetik mampu memberikan lintasan yang lebih optimal bila dibandingkan dengan metode *Joint Interpolated Trajectory* ditandai dengan nilai *kinematic manipulability* yang lebih besar dan kurva yang lebih halus.

## I. PENDAHULUAN

Peningkatan efisiensi kerja dalam dunia industri menjadi hal penting yang perlu mendapatkan perhatian khusus. Dalam proses-proses tertentu, faktor “kelelahan” bila menggunakan tenaga manusia menjadi salah satu permasalahan proses produksi, yang akan berakibat pada banyak hal diantaranya “keseragaman” suatu produk yang pembuatannya dilakukan secara masal. Alternatif lain guna mengatasi hal tersebut diantaranya dengan pemanfaatan robot yang terintegrasi dengan komputer.

Penggunaan robot dalam suatu proses produksi, belum sepenuhnya dapat mengatasi permasalahan, karena akan muncul permasalahan lain diantaranya adalah pengontrolan robot. Pengontrolan robot erat kaitannya dengan pergerakan lengan-lengannya, yang akan membentuk suatu jalur atau lintasan tertentu. Jalur pergerakan lengan-lengan tersebut dikenal dengan “*path*”. Lintasan yang baik adalah lintasan yang

pergerakannya halus sehingga kebutuhan torsi joint-jointnya kecil, dan mempunyai nilai “*manipulability*” yang tinggi. “*Manipulability*” yang tinggi memungkinkan robot untuk dapat bergerak dengan baik.

Penentuan lintasan lengan robot yang optimal, sangat tergantung pada parameter-parameter robot yang digunakan. Penentuan lintasan tersebut pada umumnya menggunakan interpolasi lintasan joint yang diinginkan dengan fungsi polinomial. Karena kompleksnya perhitungan, sering dihasilkan urutan titik-titik yang kurang optimal, sehingga diperlukan alternatif teknik penentuan lintasan yang lebih baik.

Pada Tugas Akhir ini, teknik optimasi Algoritma Genetik digunakan untuk menentukan urutan titik-titik dalam lintasan lengan robot, sehingga diperoleh suatu lintasan yang lebih tepat dan ditandai dengan nilai *kinematic manipulability* yang lebih besar dan kurva *manipulability* yang dihasilkan lebih halus. Algoritma Genetik merupakan teknik optimasi berbasis evolusi alam yang melalui proses seleksi, pindah silang dan mutasi. Percepatan angular awal dan akhir masing-masing joint manipulator akan dikodekan sebagai kromosom dengan kode biner. Kandidat solusi akan dievaluasi dengan suatu fungsi fitness yang berkaitan dengan nilai fungsi objektif *kinematic manipulability* manipulator. Teknik tersebut di atas disimulasikan dengan bahasa pemrograman MATLAB versi 5.31. Hasil-hasil yang diperoleh dari simulasi akan dibandingkan dengan metode interpolasi lintasan joint (*joint interpolated trajectory*) menggunakan polinomial orde 5.

## II. DASAR TEORI

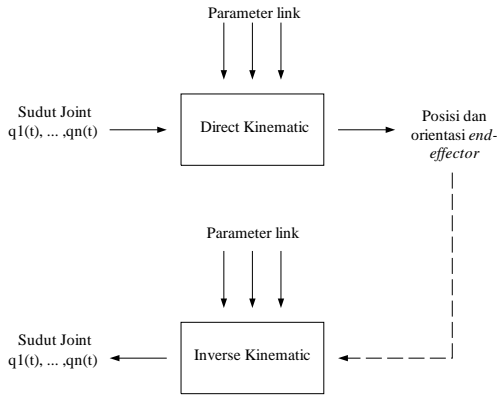
### A. Kinematik Lengan Robot

Manipulator dapat diartikan sebagai rangkaian artikulasi terbuka yang tersusun atas beberapa *link* yang terangkai secara seri baik yang dihubungkan oleh *revolute joint* ataupun *prismatic joint* [3]. Pada sisi yang satu melekat pada badan dasar (*supporting base*) yang berfungsi sebagai penyangga dan sisi yang lain terhubung dengan peralatan tertentu (*end-effector*).

Kinematik lengan robot berhubungan dengan geometri pergerakan lengan robot terhadap sistem koordinat referensi sebagai fungsi waktu, tanpa memperhatikan besarnya gaya atau moment yang menyebabkan pergerakannya. Dikenal 2 permasalahan utama dalam kinematik lengan robot yaitu *direct* (*forward*) *kinematic*, dan *inverse kinematic*.

<sup>1</sup> Mahasiswa Teknik Elektro Undip

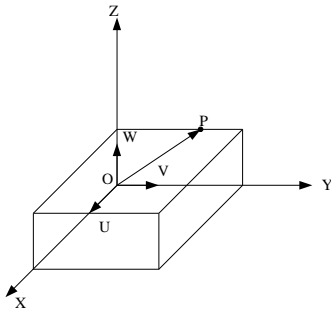
<sup>2</sup> Staf Pengajar Teknik Elektro Undip



Gambar 1. Persoalan *direct* dan *inverse kinematic*

### A.1 Matrik Rotasi

Matrik rotasi 3x3 dapat didefinisikan sebagai matrik transformasi yang menentukan posisi sebuah vektor dalam ruang 3 dimensi dan memetakan koordinatnya ke dalam sistem kordinat terrotasi OUVW terhadap sistem koordinat referensi OXYZ.



Gambar 2. Sistem koordinat referensi (OXYZ) dan bergerak (OUVW)

Kedua sistem koordinat mempunyai titik pusat (*origin*) yang sama. Titik  $\mathbf{p}$  pada Gambar 2 dapat direpresentasikan dengan kedua sistem koordinat tersebut di atas.

$$\mathbf{p}_{uvw} = (p_u, p_v, p_w)^T \text{ dan } \mathbf{p}_{xyz} = (p_x, p_y, p_z)^T \quad \dots\dots (1)$$

dimana  $\mathbf{p}_{uvw}$  dan  $\mathbf{p}_{xyz}$  menunjukkan titik  $\mathbf{p}$  yang sama dalam ruang akan tetapi berbeda sistem koordinat dan superscript T pada vektor dan matrik menunjukkan operasi *transpose*.

Titik  $\mathbf{p}$  pada  $\mathbf{p}_{xyz}$  dapat ditunjukkan dengan sistem koordinat OUVW dengan melakukan operasi rotasi.

$$\mathbf{p}_{xyz} = \mathbf{R}\mathbf{p}_{uvw} \quad \dots\dots(2)$$

dimana R merupakan matrik rotasi ,

$$\mathbf{R} = \begin{bmatrix} i_x \cdot i_u & i_x \cdot j_v & i_x \cdot k_w \\ j_y \cdot i_u & j_y \cdot j_v & j_y \cdot k_w \\ k_z \cdot i_u & k_z \cdot j_v & k_z \cdot k_w \end{bmatrix} \quad \dots\dots(3)$$

Jika sistem koordinat OUVW diputar terhadap sumbu OX sebesar sudut  $\alpha$  untuk mencapai titik baru dalam ruang, maka titik  $\mathbf{p}_{uvw}$  dengan koordinat  $(\mathbf{p}_u, \mathbf{p}_v, \mathbf{p}_w)^T$  relatif terhadap sistem OUVW mempunyai koordinat lain  $(\mathbf{p}_x, \mathbf{p}_y, \mathbf{p}_z)^T$  relatif terhadap sistem referensi OXYZ. Matrik transformasi  $\mathbf{R}_{x,\alpha}$  disebut sebagai matrik rotasi terhadap sumbu OX dengan perputaran sudut  $\alpha$ .  $\mathbf{R}_{x,\alpha}$  dapat diturunkan dari konsep matrik transformasi di atas sehingga,

$$\mathbf{p}_{xyz} = \mathbf{R}_{x,\alpha} \mathbf{p}_{uvw} \quad \dots\dots(4)$$

dengan  $i_x \equiv i_u$  dan

$$\mathbf{R}_{x,\alpha} = \begin{bmatrix} i_x \cdot i_u & i_x \cdot j_v & i_x \cdot k_w \\ j_y \cdot i_u & j_y \cdot j_v & j_y \cdot k_w \\ k_z \cdot i_u & k_z \cdot j_v & k_z \cdot k_w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad \dots\dots(5)$$

Dengan cara yang sama, matrik rotasi 3x3 untuk perputaran terhadap sumbu OY dengan sudut putar  $\varnothing$  dan terhadap sumbu OZ dengan sudut putar  $\theta$  berturut-turut :

$$\mathbf{R}_{y,\varnothing} = \begin{bmatrix} \cos \varnothing & 0 & \sin \varnothing \\ 0 & 1 & 0 \\ -\sin \varnothing & 0 & \cos \varnothing \end{bmatrix} \quad \mathbf{R}_{z,\theta} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \dots\dots(6)$$

Matrik  $\mathbf{R}_{x,\alpha}$ ,  $\mathbf{R}_{y,\varnothing}$  dan  $\mathbf{R}_{z,\theta}$  disebut sebagai matrik rotasi dasar.

### A.2 Matrik Rotasi Gabungan

Matrik rotasi dasar dapat dikalikan secara bersamaan untuk merepresentasikan urutan dari beberapa rotasi terhadap sumbu dasar sistem koordinat OXYZ. Perkalian matrik ini harus dilakukan dengan urutan yang benar. Aturan-aturan dalam penggabungan matrik rotasi tersebut adalah sebagai berikut :

1. Pada awalnya kedua sistem koordinat berhimpit, dan matrik rotasi merupakan matrik identitas 3x3.
2. Jika sistem koordinat rotasi OUVW dirotasikan terhadap sumbu kerangka OXYZ, maka berlaku aturan *premultiply*.
3. Jika sistem koordinat rotasi OUVW dirotasikan terhadap sumbunya sendiri maka berlaku aturan *postmultiply*.

### A.3 Matrik Transformasi dan Koordinat Homogen

Matrik transformasi homogen merupakan matrik 4x4 yang memetakan vektor posisi dalam koordinat homogen dari sistem koordinat yang satu ke dalam sistem kordinat yang lain. Matrik transformasi homogen tersusun atas 4 buah submatrik, ditunjukkan pada persamaan 7.

$$T = \begin{bmatrix} \mathbf{R}_{3 \times 3} & \mathbf{p}_{3 \times 1} \\ \mathbf{f}_{1 \times 3} & 1 \times 1 \end{bmatrix} = \begin{bmatrix} \text{matrik rotasi} & \text{vektor posisi} \\ \text{transformasi perspektif} & \text{skala} \end{bmatrix} \quad \dots\dots (7)$$

Jika vektor posisi  $\mathbf{p}$  dalam ruang 3 dimensi ditunjukkan dengan koordinat homogen  $\mathbf{p}=(p_x, p_y, p_z, 1)^T$ , maka dengan menggunakan konsep matrik transformasi, matrik rotasi 3x3 dapat diperluas menjadi matrik transformasi homogen 4x4  $\mathbf{T}_{rot}$  untuk operasi rotasi murni. Maka persamaan (5) dan (6) dapat direpresentasikan dengan matrik rotasi homogen menjadi :

$$T_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad T_{y,\phi} = \begin{bmatrix} \cos\phi & 0 & \sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \dots\dots (8)$$

$$T_{z,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrik rotasi 4x4 di atas dinamakan matrik rotasi homogen dasar.

Submatrik 3x1 pada sisi kanan atas dari matrik transformasi homogen menyebabkan translasi (pergeseran) sistem koordinat OUVW terhadap sistem koordinat OXYZ.

$$T_{tran} = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \dots\dots (9)$$

Matrik transformasi 4x4 di atas dinamakan matrik translasi homogen dasar.

Matrik transformasi homogen memetakan vektor yang diekspresikan dalam koordinat homogen terhadap sistem koordinat OUVW ke dalam sistem koordinat referensi OXYZ sebagai berikut,

$$\mathbf{p}_{xyz} = \mathbf{T} \mathbf{p}_{uvw} \quad \dots\dots (10)$$

#### A.4 Matrik Transformasi Homogen Gabungan

Matrik rotasi dan translasi homogen dapat dioperasikan bersama-sama untuk membuat suatu matrik transformasi homogen gabungan ( $\mathbf{T}$ ). Seperti halnya matrik lain, sifat perkalian matrik yang tidak komutatif menjadikan operasi penggabungan harus dilakukan dengan urutan yang benar.

Matrik transformasi homogen digunakan untuk menggambarkan hubungan pergerakan ruang antara sistem koordinat referensi OXYZ dan sistem koordinat bergerak (rotasi dan translasi) OUVW.

Jika kedua sistem koordinat diatas diaplikasikan pada setiap lengan robot, misalnya link  $i-1$  dan link  $i$  secara berurutan, maka sistem koordinat link ke  $i-1$  merupakan sistem koordinat referensi, sedangkan sistem koordinat link  $i$  merupakan sistem koordinat bergerak, ketika joint  $i$  aktif. Dengan menggunakan matrik transformasi homogen  $\mathbf{T}$ , maka suatu titik misalkan  $\mathbf{p}_i$  pada link  $i$  jika dilihat dari sistem koordinat link  $i-1$  (OXYZ) menjadi :

$$\mathbf{p}_{i-1} = \mathbf{T} \mathbf{p}_i \quad \dots\dots(11)$$

dimana  $\mathbf{T}$  = matrik transformasi homogen 4x4

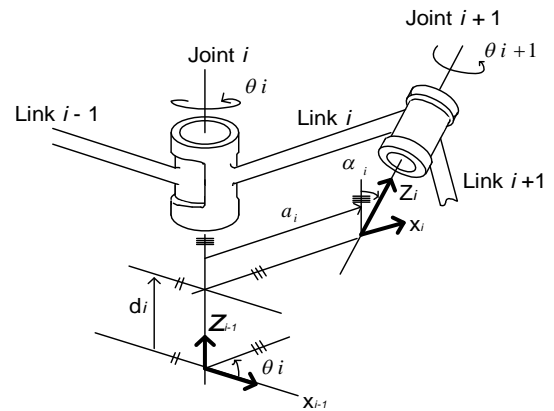
$\mathbf{p}_i$  =vektor posisi 4x1  $(x_i, y_i, z_i, 1)^T$  yang menunjukkan suatu titik pada sistem koordinat link  $i$  dalam bentuk koordinat homogen.

$\mathbf{p}_{i-1}$  = vektor posisi 4x1  $(x_{i-1}, y_{i-1}, z_{i-1}, 1)^T$  yang menunjukkan titik yang sama dengan  $\mathbf{p}_i$  dilihat dari sistem koordinat link  $i-1$ .

#### B. Link dan Joint

Setiap pasangan *link-joint* pada lengan manipulator akan membentuk 1 derajat kebebasan (1 *d.o.f*). Manipulator dengan jumlah derajat kebebasan N, terdapat pasangan joint-link sebanyak N. Penomoran *joint* dan *link* dimulai dari “*supporting base*”, dimana joint 1 merupakan titik penghubung antara link 1 dan “*supporting base*”, demikian seterusnya sampai link terakhir (*tool*).

Sumbu *joint* (untuk *joint* ke  $i$ ) terbentuk dari hubungan 2 buah *link* (Gambar 3). Posisi relatif dari 2 buah link yang terhubung (*link i-1* dan *link i*) diberikan oleh jarak  $d_i$  yang dihitung sepanjang sumbu *joint* antara kedua titik normal. Sudut *joint*  $\theta_i$  antara kedua titik normal dihitung pada bidang normal terhadap sumbu *joint*. Parameter  $d_i$  dan  $\theta_i$  sering disebut sebagai jarak dan sudut antara link-link yang berdekatan. Parameter-parameter tersebut menunjukkan posisi relatif dari link-link yang saling berdekatan.



Gambar 3. Sistem koordinat dan parameter link

Pada umumnya *link i* ( $i= 1, \dots, 6$ ) dihubungkan ke 2 *link* yang lain (misal *link i-1* dan *link i+1*), sehingga kedua sumbu joint terbentuk pada akhir kedua hubungan tersebut. Terdapat 2 parameter lain yang menyusun

konfigurasi suatu lengan manipulator dan bersifat tetap yaitu  $a_i$  dan  $\alpha_i$ . Parameter  $a_i$  merupakan jarak antara sumbu  $Z_{i-1}$  dan  $Z_i$  diukur sepanjang sumbu  $X_i$  negatif, mulai dari pusatnya sampai berpotongan dengan sumbu  $Z_{i-1}$ . Sedangkan  $\alpha_i$  adalah sudut antara sumbu  $Z_{i-1}$  dan  $Z_i$  yang diputar terhadap sumbu  $X_i$ . Kedua parameter  $a_i$  dan  $\alpha_i$  dinamakan sebagai panjang dan sudut putaran dari link  $i$ . Struktur dari link  $i$  ditentukan oleh kedua parameter ini.

Jika keempat parameter diatas ( $d_i, \theta_i, a_i, \alpha_i$ ) telah diketahui, maka konfigurasi kinematik setiap link dari suatu lengan robot dapat ditentukan.

### C. Persamaan Denavit-Hartenberg

Representasi Denavit-Hartenberg (D-H) merupakan salah satu metode yang dapat digunakan untuk menggambarkan hubungan rotasi dan translasi link-link yang saling berdekatan pada suatu lengan manipulator. Metode ini menggunakan matrik transformasi homogen 4x4 yang merepresentasikan sistem koordinat setiap link terhadap sistem koordinat link sebelumnya.

Sistem koordinat kartesius orthonormal ( $x_i, y_i, z_i$ ) dapat dibuat untuk setiap linknya pada sumbu joint, dimana  $i = 1, 2, \dots, n$  ( $n =$  jumlah derajat kebebasan). Karena sumbu rotasi hanya mempunyai 1 derajat kebebasan, maka setiap kerangka koordinat ( $x_i, y_i, z_i$ ) dari lengan robot berhubungan dengan joint  $i+1$  dan terletak pada link  $i$  yang tetap. Dan karena sistem koordinat ke- $i$  terletak pada link  $i$ , maka saat terjadi pergerakan maka sistem koordinat ini akan ikut bergerak bersama-sama dengan link  $i$ . Sehingga kerangka koordinat ke- $n$  bergerak bersama link ke- $n$ . Koordinat basis yang bersifat tetap didefinisikan sebagai kerangka koordinat ke-0 ( $x_0, y_0, z_0$ ) yang merupakan kerangka koordinat awal dari lengan robot.

Setiap kerangka koordinat ditetapkan dan dibuat berdasarkan 3 aturan :

1. Sumbu  $Z_{i-1}$  terletak sepanjang sumbu gerak dari joint ke- $i$ .
2. Posisi sumbu  $X_i$  normal terhadap sumbu  $Z_{i-1}$ , dan dimulai dari titik normalnya.
3. Sumbu  $Y_i$  menyesuaikan kedua sumbu sebelumnya dengan menggunakan aturan tangan kanan.

Representasi D-H dari link-link manipulator tergantung pada 4 parameter geometrik yang terdapat pada setiap link. Keempat parameter tersebut adalah  $a_i, \alpha_i, \theta_i$ , dan  $d_i$ .

Langkah-langkah pembentukan matrik D-H pada suatu manipulator [8] adalah sebagai berikut :

1. Beri nomor setiap link dan joint. Base yang tetap ("stationary base") dijadikan sebagai link ke-0 sedangkan *end-effector* sebagai link ke- $n$ . Di sini link  $i$  bergerak terhadap link  $i-1$ .
2. Buat sistem koordinat link-link untuk setiap joint berdasarkan aturan
  - Sumbu  $Z_{i-1}$  dipilih sepanjang sumbu gerak dari joint  $i$ . Untuk *revolute joint*, link  $i$  berotasi terhadap link  $i-1$  pada sumbu  $Z_{i-1}$  sebesar  $\theta_i$ .

Sedangkan untuk *prismatic joint*, link  $i$  bergeser relatif terhadap link  $i-1$  sepanjang sumbu  $Z_{i-1}$  sebesar  $d_i$ .

- Sumbu  $X_i$  dipilih tegak lurus terhadap sumbu  $Z_{i-1}$ . Jika  $Z_i$  dan  $Z_{i-1}$  tidak berpotongan, maka sumbu  $X_i$  berada sepanjang titik normal dari  $Z_i$  dan  $Z_{i-1}$  dan arahnya ditentukan mulai dari  $Z_{i-1}$  menuju  $Z_i$ .
  - Sumbu  $Y_i$  dipilih untuk melengkapi sistem koordinat di atas sesuai dengan aturan tangan kanan.
3. Penentuan parameter joint dan link ( $\theta_i, d_i, \alpha_i$  dan  $a_i$ )
  4. Tentukan matrik transformasi homogen untuk joint ke- $i$

$$T_{i-1}^i = \begin{bmatrix} C_{i-1}^i & & & d_{i-1}^i \\ & \dots & & \\ & & \dots & \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \dots\dots(12)$$

dimana

$C_{i-1}^i$  : matrik orientasi 3x3 dari link  $i$  terhadap link  $i-1$

$$C_{i-1}^i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i \end{bmatrix} \quad \dots\dots(13)$$

$d_{i-1}^i$  : vektor translasi 3x1, menunjukkan posisi dari pusat kerangka koordinat  $O_{i-1}$

$$d_{i-1}^i = \begin{bmatrix} a_i \cos \theta_i \\ a_i \sin \theta_i \\ d_i \end{bmatrix} \quad \dots\dots(14)$$

### D. Kinematic Manipulability

*Manipulability* digunakan dalam desain baik kinematik maupun dinamik lengan robot. Untuk dapat mencapai posisi yang optimal dalam suatu *workspace* atau ruang gerak robot, diperlukan nilai *manipulability* yang tinggi. *Manipulability* suatu manipulator dapat didefinisikan sebagai

$$\omega = \sqrt{\det((J(q)J^T(q)))} \quad \dots\dots (15)$$

dimana  $q$  : vektor sudut joint manipulator  
 $J(q)$  : matrik jacobian manipulator  
 $J^T(q)$  : transpose matrik jacobian manipulator.

### E. Algoritma Genetik

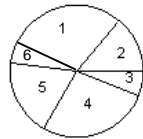
Algoritma genetik merupakan algoritma optimisasi yang diilhami oleh proses biologi yang dapat diterapkan

pada sebagian besar permasalahan. Algoritma genetik adalah suatu metoda optimisasi yang memodelkan mekanisme seleksi alam dan proses genetika untuk menuntun suatu pencarian seperti cara-cara alam dalam menyelesaikan permasalahan adaptasi organisme, untuk mempertahankan kelangsungan hidupnya .

Algoritma genetik mengevaluasi dan memanipulasi string-string individu dengan operator genetika sehingga dihasilkan individu baru yang meningkat kualitasnya untuk generasi selanjutnya. Tiga operator dasar yang sering digunakan adalah reproduksi, pindah silang, dan mutasi.

1. Reproduksi

Reproduksi adalah proses penyalinan string individu sesuai dengan nilai fitnessnya. Setiap individu dalam populasi pada suatu generasi akan diseleksi berdasarkan nilai fitnessnya. Probabilitas terpilihnya satu individu untuk bereproduksi adalah sebesar nilai fitness individu tersebut dibagi dengan jumlah nilai fitness seluruh individu dalam populasi. Banyak salinan satu individu sebanding dengan nilai fitness individu tersebut. Sehingga individu dengan nilai fitness lebih besar akan diproduksi lebih banyak. Salah satu cara mengimplementasikan proses reproduksi adalah dengan piringan rolet.

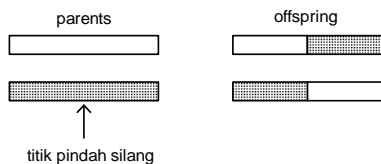


Gambar 4. Piringan rolet

2. Pindah Silang

Pindah silang bertujuan menambah keanekaragaman individu dalam populasi dengan mengawinkan individu-individu dalam populasi ( $P_t$ ) sehingga menghasilkan keturunan berupa individu-individu baru untuk ditempatkan pada populasi ( $P_{t+1}$ ). Individu baru akan memiliki bagian dari struktur kedua orang tuanya.

Gambar 5 menunjukkan proses pindah silang pada satu titik dari sepasang *parent* hingga diperoleh sepasang individu baru hasil pindah silang (*offspring*).



Gambar 5. Proses pindah silang satu titik.

3. Mutasi

Melalui mutasi individu baru dapat diciptakan dengan melakukan modifikasi terhadap satu atau lebih gen pada individu yang sama.. Gambar 6 menunjukkan individu/kromosom yang termutasi.

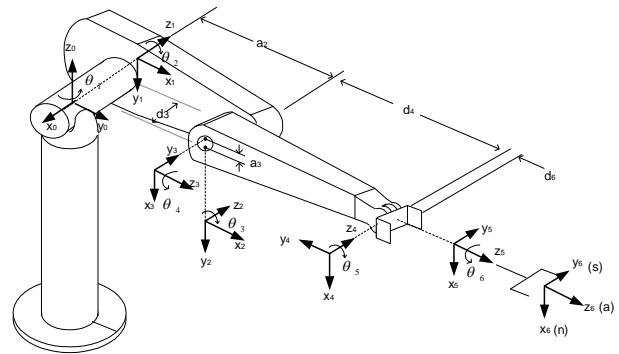


Gambar 6 Kromosom yang mengalami mutasi pada bit ke empat.

III. PERANCANGAN

A. Model Lengan Robot Puma560

Pada tugas akhir ini, plant yang digunakan adalah model lengan robot Puma560 dengan 6 derajat kebebasan. Secara garis besar, bentuk, sistem koordinat dan parameter-parameter yang menyusunnya ditunjukkan pada Gambar 7.



Gambar 7. Sistem koordinat model robot Puma560

TABEL I  
PARAMETER KOORDINAT LINK LINGAN ROBOT PUMA560

Joint $i$	$\alpha_i$	$a_i$	$d_i$	Jangkauan joint
1	-90	0	0	-160 sampai +160
2	0	431.8 mm	0	-225 sampai +45
3	90	-20.32 mm	149.09 mm	-45 sampai +225
4	-90	0	433.07 mm	-110 sampai +170
5	90	0	0	-100 sampai +100
6	0	0	56.25 mm	-266 sampai +266

Untuk menunjukkan rotasi dan translasi antar link-linknya, maka sistem koordinat yang menyusunnya dibuat dengan menggunakan metode Denavit-Hartenberg. Metode ini menggunakan matrik transformasi homogen untuk merepresentasikan setiap sistem koordinat link pada setiap jointnya terhadap sistem koordinat link sebelumnya. Dengan menggunakan aturan-aturan pada metode Denavit-Hartenberg, diperoleh parameter-parameter link dan joint model lengan robot Puma560 yang dapat dilihat pada Tabel I

Matrik transformasi yang digunakan untuk merepresentasikan sistem koordinat ke  $i$  terhadap sistem kordinat ke  $i-1$  adalah

$${}^{i-1}A_i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots (16)$$

Dengan menggunakan matrik pada persamaan (16) di atas maka sistem koordinat antar link-link dapat diperoleh sebagai berikut :

$${}^0A_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^1A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots (17)$$

$${}^2A_3 = \begin{bmatrix} C_3 & 0 & S_3 & a_3 C_3 \\ S_3 & 0 & -C_3 & a_3 S_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^3A_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots (18)$$

$${}^4A_5 = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^5A_6 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & -1 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots (19)$$

$$\mathbf{T}_1 = {}^0A_1 {}^1A_2 {}^2A_3 \\ = \begin{bmatrix} C_1 C_{23} & -S_1 & C_1 S_{23} & a_2 C_1 C_2 + a_3 C_1 C_{23} - d_2 S_1 \\ S_1 C_{23} & C_1 & S_1 S_{23} & a_2 S_1 C_2 + a_3 S_1 C_{23} + d_2 C_1 \\ -S_{23} & 0 & C_{23} & -a_2 S_2 - a_3 S_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots (20)$$

$$\mathbf{T}_2 = {}^3A_4 {}^4A_5 {}^5A_6 \\ = \begin{bmatrix} C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & C_4 S_5 & d_6 C_4 S_5 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & S_4 S_5 & d_6 S_4 S_5 \\ -S_5 C_6 & S_5 S_6 & C_5 & d_6 C_5 + d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots (21)$$

dimana  $C_i \equiv \cos \theta_i$  ;  $S_i \equiv \sin \theta_i$  ;  $C_{ij} \equiv \cos(\theta_i + \theta_j)$  ;  $S_{ij} \equiv \sin(\theta_i + \theta_j)$ .

Untuk memperoleh keseluruhan matrik lengan robot Puma560,

$$\mathbf{T} = \mathbf{T}_1 \mathbf{T}_2 = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5 {}^5A_6 \dots (22)$$

## B. Penentuan Titik Awal dan Titik Akhir

TABEL II  
TITIK AWAL DAN AKHIR DARI LINTASAN END-EFFECTOR

No	Titik awal (x, y,z)	Titik akhir (x, y, z)
1	(0.4521, -0.1500, 0.4318)	(0.6861, -0.1501, 0.3535)
2	(-0.1501, -0.5557, -0.5172)	(0.6861, -0.1501, 0.3535)
3	(0.4521, -0.1501, 0.4318)	(-0.1501, -0.5557, -0.5172)
4	(0.4521, -0.1501, 0.4318)	(0.5605, 0.3205, -0.5842)

Pada tugas akhir ini, titik awal dan titik akhir ditentukan oleh penulis. Pergerakan manipulator dimulai dari titik awal dan harus berhenti pada titik akhir tersebut. Titik-titik tersebut digambarkan dalam koordinat

kartesian 3 dimensi yang merupakan posisi dari *end effector* (x, y, z). Di sini diambil 4 contoh variasi titik awal dan titik akhir sebagaimana pada Tabel II.

## C. Fungsi Objektif dan Fungsi Fitness

Baik tidaknya suatu kandidat solusi dalam algoritma genetik akan ditentukan oleh fungsi objektif atau fungsi fitness. Kandidat solusi dengan fitness terbaik akan keluar sebagai pemenang dan dapat menjadi solusi dari suatu permasalahan baik itu ditandai dengan nilai minimasi maupun maksimasi. Pada tugas akhir ini penulis menggunakan nilai *kinematic manipulability* sebagai fungsi objektif untuk menentukan performansi algoritma genetik dalam mencari suatu lintasan manipulator yang optimal sehingga memberikan kinerja robot yang baik. Semakin besar nilai *kinematic manipulability* maka semakin baik lintasan yang dilalui manipulator.

Fungsi objektif yang digunakan dalam pencarian atau penentuan lintasan manipulator dengan algoritma genetik ditunjukkan oleh persamaan 23.

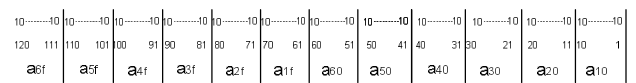
$$f = \max(\sqrt{\det(J(\theta)J^T(\theta))}) \dots (23)$$

dimana  $\theta$  = vektor sudut joint manipulator

$J(\theta)$  = matrik jacobian manipulator

## D. Pengkodean Parameter

Pengkodean parameter merupakan hal pertama yang harus dilakukan dalam proses algoritma genetik. Pada tugas akhir ini, parameter yang akan dikodekan adalah percepatan awal dan percepatan akhir masing-masing joint manipulator yang berjumlah 6 joint. Jadi jumlah total parameter yang dikodekan sebanyak 12 (masing-masing joint 2 parameter, yaitu percepatan awal dan akhir). Masing-masing parameter dikodekan dengan angka biner 0 atau 1 sepanjang 10 bit, sehingga panjang total kromosom untuk setiap individunya adalah 120 bit. Susunan kromosomnya tampak seperti pada Gambar 8.



Gambar 8. Susunan bit-bit kromosom untuk setiap parameter

dimana  $a_{i0}$  dan  $a_{if}$  berturut-turut adalah percepatan awal dan akhir joint ke  $i$  sedangkan  $n$  adalah banyaknya derajat kebebasan, dalam hal ini 6.

Setelah proses pengkodean, maka proses selanjutnya adalah proses pendekodean parameter dimana pada proses ini, nilai bit-bit biner diubah ke dalam nilai integer masing-masing parameter.

$$a_{i0\_decode} = \sum_{k=10(i-1)+1}^{10i} (d_k x 2^{k-(10(i-1)+1)}) \dots (24)$$

$$a_{if\_dekode} = \sum_{k=10(i-1)+61}^{10i+60} (d_k x 2^{k-(10(i-1)+61)}) \quad \dots\dots(25)$$

dimana

- $a_{i0\_dekode}$  = percepatan awal joint ke  $i$  hasil dekode  
 $a_{if\_dekode}$  = percepatan akhir joint ke  $i$  hasil dekode  
 $d_k$  = bilangan biner ke- $k$   
 $i$  = banyaknya joint

Parameter hasil dekode berupa bilangan bulat yang merepresentasikan nilai binernya dalam jangkauan nilai  $0-2^L$  dimana  $L$  adalah panjang bit tiap parameter. Bilangan bulat hasil dekode tersebut harus dipetakan pada interval tertentu  $[U_{min}, U_{max}]$  dengan interval dan ketelitian pemetaan ditunjukkan oleh persamaan (26).

$$\pi = \frac{U_{max} - U_{min}}{2^L - 1} \quad \dots\dots(26)$$

Dengan menggunakan interval tersebut, maka nilai dari masing-masing parameter percepatan adalah :

$$a_{i0} = (a_{i0\_dekode} x \pi) + U_{min} \quad \dots\dots(27)$$

$$a_{if} = (a_{if\_dekode} x \pi) + U_{min} \quad \dots\dots(28)$$

dengan:

- $U_{max}$  = batas atas bilangan riil  
 $U_{min}$  = batas bawah bilangan riil  
 $L$  = panjang kromosom tiap parameter  
 $a_{i0}$  = nilai parameter percepatan awal dalam bilangan riil  
 $a_{if}$  = nilai parameter percepatan akhir dalam bilangan riil

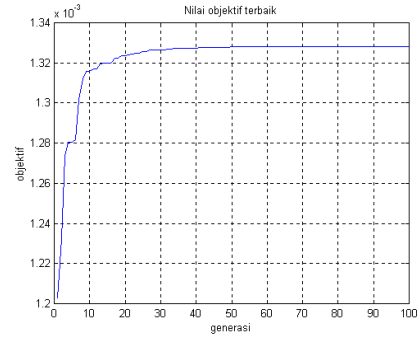
#### E. Parameter Genetik

Sebelum memulai proses genetik, maka perlu terlebih dahulu ditentukan parameter-parameter yang digunakan. Parameter-parameter genetik tersebut adalah jumlah generasi, jumlah populasi, probabilitas pindah silang, probabilitas mutasi, gap antar generasi dan penentuan range optimasi. Pada tugas akhir ini ditentukan jumlah generasi sebanyak 100, jumlah populasi 30, probabilitas pindah silang (pc) 0.9, probabilitas mutasi 0.01, gap antar generasi 0.9 dan besarnya range optimasi pada interval  $[0, 0.01]$ .

### IV. HASIL SIMULASI DAN ANALISA

#### A. Path 1

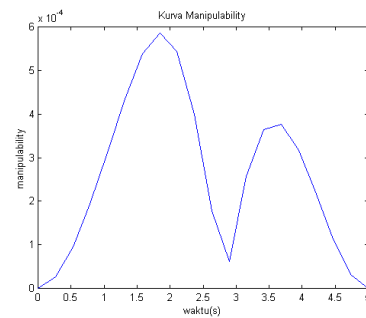
Pada *path 1*, manipulator harus bergerak mulai dari titik awal  $[x, y, z] = [0.4521, -0.1500, 0.4318]$  dan berakhir pada titik  $[x, y, z] = [0.6861, -0.1501, 0.3535]$ . Hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 9.



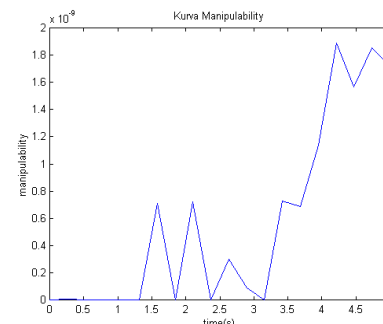
Gambar 9. Performansi algoritma genetik pada *Path 1*

Pada Gambar 9 terlihat bahwa generasi pertama menghasilkan nilai objektif sebesar 0.0012028. Nilai objektif mulai menunjukkan perubahan yang cukup besar pada generasi ke 4 yaitu sebesar 0.0012807. Proses pencarian solusi terbaik dengan algoritma genetik terus berlanjut dan nilai objektif terus bertambah besar sebagaimana tujuan optimasi yaitu untuk memaksimalkan fungsi objektif. Solusi optimal mulai diperoleh pada generasi ke 54 ditandai dengan tidak terjadinya perubahan nilai mulai generasi ke 54 sampai generasi terakhir yaitu generasi ke 100. Nilai objektif terbaik yang diperoleh adalah 0.0013281. Nilai objektif setiap generasi yang lebih baik atau sama bila dibandingkan dengan nilai objektif sebelumnya tetap dipertahankan pada generasi berikutnya. Hal ini menunjukkan seleksi elit berlaku pada proses algoritma genetik.

Nilai *kinematic manipulability* hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 10, sedangkan hasil dengan menggunakan metoda *joint interpolated trajectory* ditunjukkan pada Gambar 11.



Gambar 10. Nilai *kinematic manipulability* *Path 1* dengan metode algoritma genetik

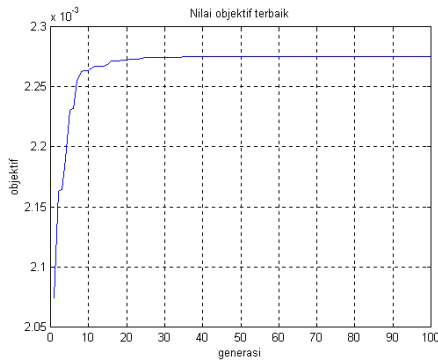


Gambar 11 Nilai *kinematic manipulability* *Path 1* dengan metode *joint interpolated trajectory*

Pada Gambar 10 dan Gambar 11 terlihat bahwa *kinematic manipulability* hasil pencarian dengan metode algoritma genetik lebih besar bila dibandingkan dengan *kinematic manipulability* hasil pencarian dengan metode *joint interpolated trajectory*. Nilai total *kinematic manipulability* diperoleh dengan cara menghitung luas area di dalam kurva *manipulability*, masing-masing pada Gambar 10 dan Gambar 11. Dengan metode algoritma genetik, diperoleh nilai total *kinematic manipulability* sebesar 0.0013281 sedangkan dengan metode *joint interpolated trajectory* sebesar  $2.7735127 \times 10^{-9}$ . Semakin besar nilai *kinematic manipulability* maka semakin baik lintasan yang dilalui manipulator. Demikian juga semakin halus bentuk kurvanya maka semakin halus gerakan lengan manipulatornya.

### B. Path 2

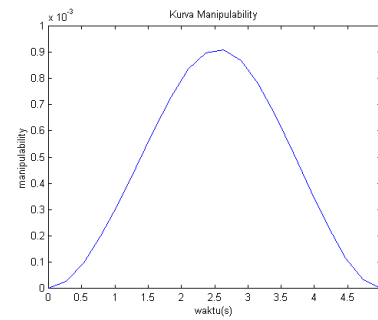
Pada *path 2*, manipulator harus bergerak mulai dari titik awal  $[x, y, z] = [-0.1501, -0.5557, -0.5172]$  dan berakhir pada titik  $[x, y, z] = [0.6861, -0.1501, 0.3535]$ . Hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 12.



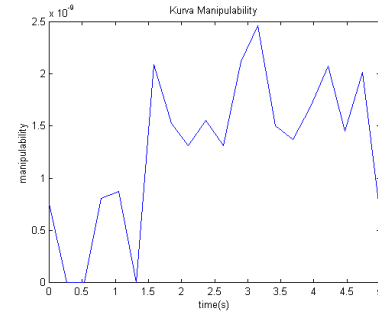
Gambar 12. Performansi algoritma genetik pada *Path 2*

Pada Gambar 12 terlihat bahwa generasi pertama menghasilkan nilai objektif sebesar 0.0020740. Nilai objektif mulai menunjukkan perubahan yang cukup besar pada generasi ke 5 yaitu sebesar 0.0022303. Proses pencarian solusi terbaik dengan algoritma genetik terus berlanjut dan nilai objektif terus bertambah besar sebagaimana tujuan optimasi yaitu untuk memaksimalkan fungsi objektif. Solusi optimal mulai diperoleh pada generasi ke 35 ditandai dengan tidak terjadinya perubahan nilai mulai generasi ke 35 sampai generasi terakhir yaitu generasi ke100. Nilai objektif terbaik yang diperoleh adalah 0.0022749. Nilai objektif setiap generasi yang lebih baik atau sama bila dibandingkan dengan nilai objektif sebelumnya tetap dipertahankan pada generasi berikutnya. Hal ini menunjukkan seleksi elit berlaku pada proses algoritma genetik.

Nilai *kinematic manipulability* hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 13, sedangkan hasil dengan menggunakan metoda *joint interpolated trajectory* ditunjukkan pada Gambar 14



Gambar 13 Nilai *kinematic manipulability* *Path 2* dengan metode algoritma genetik



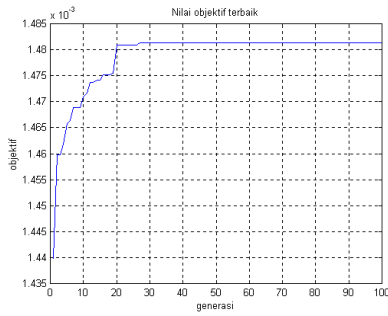
Gambar 14 Nilai *kinematic manipulability* *Path 2* dengan metode *joint interpolated trajectory*

Skala pada Gambar 13 dan Gambar 14 dibuat berbeda karena perbedaan nilai *kinematic manipulability* dari kedua metode yang cukup besar. Pada Gambar 13 dan Gambar 14 terlihat bahwa *kinematic manipulability* hasil pencarian dengan metode algoritma genetik lebih besar bila dibandingkan dengan *kinematic manipulability* hasil pencarian dengan metode *joint interpolated trajectory*. Nilai total *kinematic manipulability* diperoleh dengan cara menghitung luas area di dalam kurva *manipulability*, masing-masing pada Gambar 13 dan Gambar 14. Dengan metode algoritma genetik, diperoleh nilai total *kinematic manipulability* sebesar 0.0022749 sedangkan dengan metode *joint interpolated trajectory* sebesar  $6.5417787 \times 10^{-9}$ . Semakin besar nilai *kinematic manipulability* maka semakin baik lintasan yang dilalui manipulator. Demikian juga semakin halus bentuk kurvanya maka semakin halus gerakan lengan manipulatornya.

### C. Path 3

Pada *path 3*, manipulator harus bergerak mulai dari titik awal  $[x, y, z] = [0.4521, -0.1501, 0.4318]$  dan berakhir pada titik  $[x, y, z] = [-0.1501, -0.5557, -0.5172]$ . Hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 15

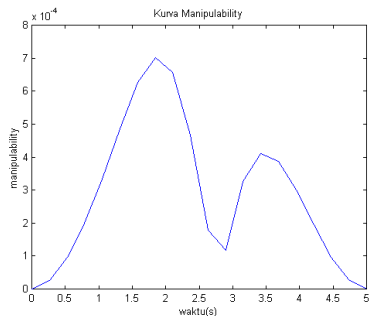




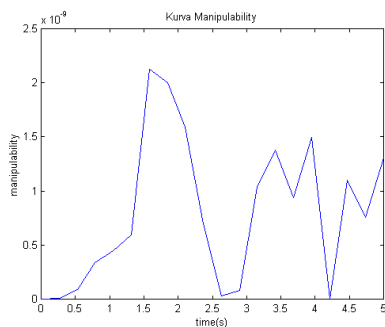
Gambar 15 Performansi algoritma genetik pada Path 3

Pada Gambar 15 terlihat bahwa generasi pertama menghasilkan nilai objektif sebesar 0.0013392. Nilai objektif mulai menunjukkan perubahan yang cukup besar pada generasi ke 3 yaitu sebesar 0.0014432. Proses pencarian solusi terbaik dengan algoritma genetik terus berlanjut dan nilai objektif terus bertambah besar sebagaimana tujuan optimasi yaitu untuk memaksimalkan fungsi objektif. Solusi optimal mulai diperoleh pada generasi ke 28 ditandai dengan tidak terjadinya perubahan nilai mulai generasi ke 28 sampai generasi terakhir yaitu generasi ke 100. Nilai objektif terbaik yang diperoleh adalah 0.0014814. Nilai objektif setiap generasi yang lebih baik atau sama bila dibandingkan dengan nilai objektif sebelumnya tetap dipertahankan pada generasi berikutnya. Hal ini menunjukkan seleksi elit berlaku pada proses algoritma genetik.

Nilai *kinematic manipulability* hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 16, sedangkan hasil dengan menggunakan metoda *joint interpolated trajectory* ditunjukkan pada Gambar 17.



Gambar 16 Nilai *kinematic manipulability* Path 3 dengan metode algoritma genetik

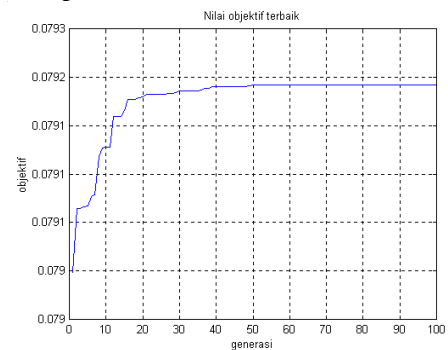


Gambar 17. Nilai *kinematic manipulability* Path 3 dengan metode *joint interpolated trajectory*

Pada Gambar 16 dan Gambar 17 terlihat bahwa *kinematic manipulability* hasil pencarian dengan metode algoritma genetik lebih besar bila dibandingkan dengan *kinematic manipulability* hasil pencarian dengan metode *joint interpolated trajectory*. Nilai total *kinematic manipulability* diperoleh dengan cara menghitung luas area di dalam kurva *manipulability*, masing-masing pada Gambar 16 dan Gambar 17. Dengan metode algoritma genetik, diperoleh nilai total *kinematic manipulability* sebesar 0.0014814 sedangkan dengan metode *joint interpolated trajectory* sebesar  $4.03608991 \times 10^{-9}$ . Semakin besar nilai *kinematic manipulability* maka semakin baik lintasan yang dilalui manipulator. Demikian juga semakin halus bentuk kurvanya maka semakin halus gerakan lengan manipulatornya.

#### D. Path 4

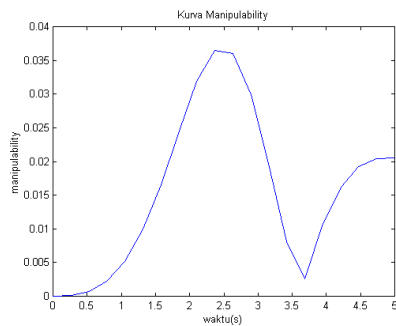
Pada path 4, manipulator harus bergerak mulai dari titik awal  $[x, y, z] = [0.4521, -0.1501, 0.4318]$  dan berakhir pada titik  $[x, y, z] = [0.5605, 0.3205, -0.5842]$ . Hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 18.



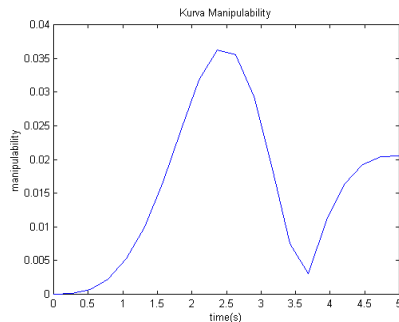
Gambar 18. Performansi algoritma genetik pada Path 4

Pada Gambar 18 terlihat bahwa generasi pertama menghasilkan nilai objektif sebesar 0.0789983. Nilai objektif mulai menunjukkan perubahan yang cukup besar pada generasi ke 7 yaitu sebesar 0.0791183. Proses pencarian solusi terbaik dengan algoritma genetik terus berlanjut dan nilai objektif terus bertambah besar sebagaimana tujuan optimasi yaitu untuk memaksimalkan fungsi objektif. Solusi optimal mulai diperoleh pada generasi ke 54 ditandai dengan tidak terjadinya perubahan nilai mulai generasi ke 54 sampai generasi terakhir yaitu generasi ke 100. Nilai objektif terbaik yang diperoleh adalah 0.0791923. Nilai objektif setiap generasi yang lebih baik atau sama bila dibandingkan dengan nilai objektif sebelumnya tetap dipertahankan pada generasi berikutnya. Hal ini menunjukkan seleksi elit berlaku pada proses algoritma genetik.

Nilai *kinematic manipulability* hasil pencarian dengan menggunakan metoda algoritma genetik ditunjukkan pada Gambar 19, sedangkan hasil dengan menggunakan metoda *joint interpolated trajectory* ditunjukkan pada Gambar 20.



Gambar 19 Nilai *kinematic manipulability* Path 4 dengan metode algoritma genetik



Gambar 20. Nilai *kinematic manipulability* Path 4 dengan metode *joint interpolated trajectory*

Pada Gambar 19 dan Gambar 20 terlihat bahwa *kinematic manipulability* hasil pencarian dengan metode algoritma genetik lebih besar bila dibandingkan dengan *kinematic manipulability* hasil pencarian dengan metode *joint interpolated trajectory*. Nilai total *kinematic manipulability* diperoleh dengan cara menghitung luas area di dalam kurva *manipulability*, masing-masing pada Gambar 19 dan Gambar 20. Dengan metode algoritma genetik, diperoleh nilai total *kinematic manipulability* sebesar 0.0791923 sedangkan dengan metode *joint interpolated trajectory* sebesar 0.0788093. Semakin besar nilai *kinematic manipulability* maka semakin baik lintasan yang dilalui manipulator. Demikian juga semakin halus bentuk kurvanya maka semakin halus gerakan lengan manipulatornya.

## V. KESIMPULAN DAN SARAN

### A. Kesimpulan

Dari hasil simulasi dan analisa terhadap 4 buah variasi titik awal dan titik akhir *path* yang harus dilalui manipulator model robot Puma560, dapat diambil beberapa kesimpulan sebagai berikut :

1. Pada penentuan lintasan manipulator, algoritma genetik dengan jumlah generasi 100 dan jumlah populasi 30 sudah mampu menemukan solusi optimal global untuk masing-masing variasi lintasan.

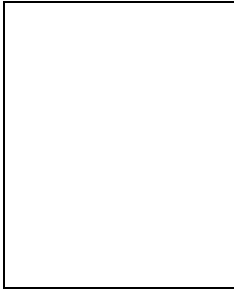
2. Metode Algoritma Genetik menghasilkan nilai *kinematic manipulability* yang lebih besar bila dibandingkan dengan metode *Joint Interpolated Trajectory*.
3. Metode Algoritma Genetik menghasilkan kurva *kinematic manipulability* yang lebih halus bila dibandingkan dengan metode *Joint Interpolated Trajectory*.

### B. Saran

1. Penelitian pada tugas akhir ini, algoritma genetik yang digunakan merupakan *Simple Genetic Algorithm*, pada penelitian berikutnya bisa digunakan algoritma genetik dengan tingkat yang lebih tinggi.
2. Model robot yang digunakan hanya satu jenis yaitu Puma560, untuk itu bisa ditambah dengan berbagai variasi model robot yang lain.

## DAFTAR PUSTAKA

- [1] Chipperfield, Andrew., *Genetic Algorithm Toolbox User's Guide*, Version 1.2, Department of Automatic Control and Systems Engineering, University of Sheffield.
- [2] Davis, L., *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.
- [3] Fu, K.S., Gonzales, R.C., Lee, C.S.G., *Robotics : Control, Sensing, Vision and Intelligence*, McGraw-Hill, New York, 1988.
- [4] Fuller, James L., *Robotics : Introduction, Programming, and Projects.*, Macmillan Publishing Company, New York.
- [5] Goldberg, David E, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc, 1989.
- [6] Groover, Mikell P., Weiss, M., Nagel, R.N., Odrey, N.G., *Industrial Robotics : Technology, Programming and Applications.*, McGraw-Hill Book Company, ew York.
- [7] Klatfer, R.K., Chmielewski, T.A., *Robotics Engineering : An Integrated Approach*, Prentice-Hall International, Inc.
- [8] Koren. Yoram, *Robotics For Engineers*, McGraw-Hill Book Company, New York.
- [9] Man, K.F., K.S. Tang, S.Kwong, W.A. Halang, *Genetic Algorithms for Control and Signal Processing*, Springer-Verlag London, 1997.
- [10] Michelewicz, Zbigniew., *Genetic Algorithms + Data Structures = Evolution Programs*, Third, revised and Extended Edition, Springer-Verlag Berlin Heidelberg New York, 1995.
- [11] Park, F.C., Kim,J.W., *Manipulability of Closed Kinematic Chains*, School of Mechanical, and Aerospace engineering, Seoul National University, Korea.
- [12] Staffetti, E., Bruyninckx,H., De Schutter, J, *On The Invariance of Manipulability Indices*, Departement of Mechanical Engineering, Katholieke Universiteit Leuven.
- [13] ..... , [www.d.umn.edu/lis/ga](http://www.d.umn.edu/lis/ga)



**Temon Wahyu P.** Mahasiswa kelahiran Wonogiri, Januari 1979. Mulai terdaftar sebagai Mahasiswa Teknik Elektro Universitas Diponegoro Semarang pada tahun 1997. Konsentrasi yang ditekuni adalah pada bidang Kontrol .  
e-mail : [moon\\_el97@yahoo.com](mailto:moon_el97@yahoo.com)

Mengetahui

Pembimbing I

Pembimbing II

Wahyudi, ST, MT  
NIP. 132 086 662

Aris Triwiyatno, ST  
NIP 132 230 559