

WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI INSTYTUT AUTOMATYKI

PISEMNE OPRACOWANIE PROJEKTU

**Symulacja komputerowa układów robotyki**

**Model i działanie manipulatora POO**

**Wykonali:**

Paweł Richter

Robert Jureczko

**Prowadzący:**

dr hab. inż. Ryszard Beniak, prof. PO

**Spis treści:**

1. Cel projektu
2. Wstęp teoretyczny
3. Model manipulatora
4. Przestrzeń robocza
5. Określenie przeniesienia napędu
6. Proste zadanie kinematyki
7. Cel projektu

Celem projektu jest zaprojektowanie rozwiązań manipulatora trój złączowego składającego się z jednego złącza przesuwnego oraz dwóch złącz obrotowych z stratami w złączu drugim, który jest podwieszany.

1. Wstęp teoretyczny

**Manipulator** – w robotyce, urządzenie służące do manipulowania elementami bez bezpośredniego kontaktu fizycznego operatora, zwane też „mechanicznym ramieniem”, stosowane w przemyśle w dużej mierze w fabrykach samochodów, automatycznych liniach produkcyjnych, fabrykach w których istnieje zagrożenie dla zdrowia ludzi oraz tam gdzie czynności wykonywane są powtarzalne. Dla łatwiejszego opisu ramienia wprowadzone zostały pojęcia: człon automatyki, współrzędne lokalne, współrzędne globalne, kinematyka manipulatora, stopnie swobody oraz notacja Denavita-Hartenberga. Pozwalają one w sformalizowany sposób opisać budowę manipulatora oraz zależności występujące pomiędzy kolejnymi elementami składowymi.

1. Model manipulatora

Wymiary naszego manipulatora to:

Długość toru jezdnego: 1500mm

Szerokość podstawy robota: 619mm

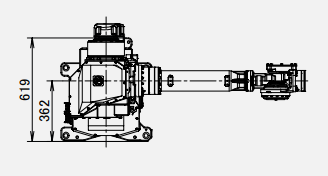
Długość ramienia L1: 870mm

Długość ramienia L2: 1016

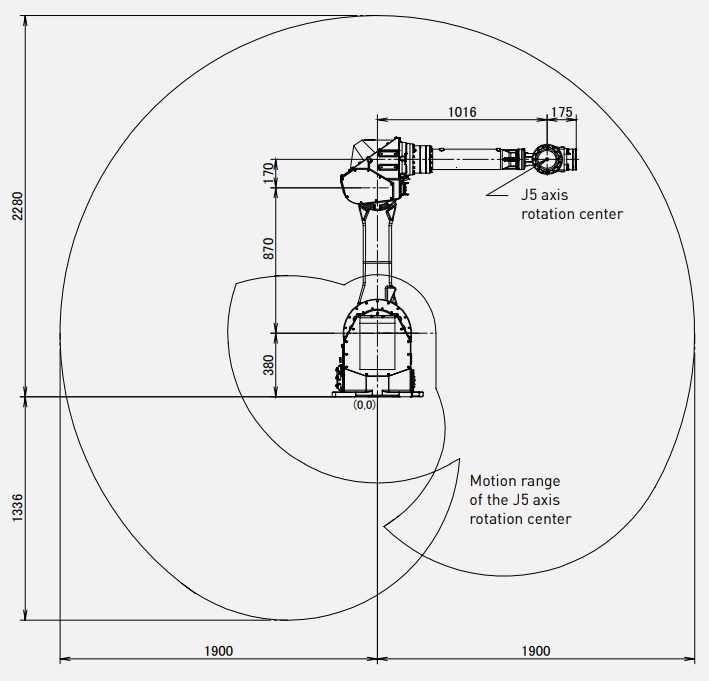
Obraz zawierający żółty

Opis wygenerowany automatycznie

Rys 3.1 Zdjęcie poglądowe robota



Rys 3.2 Wymiary robota perspektywa górna

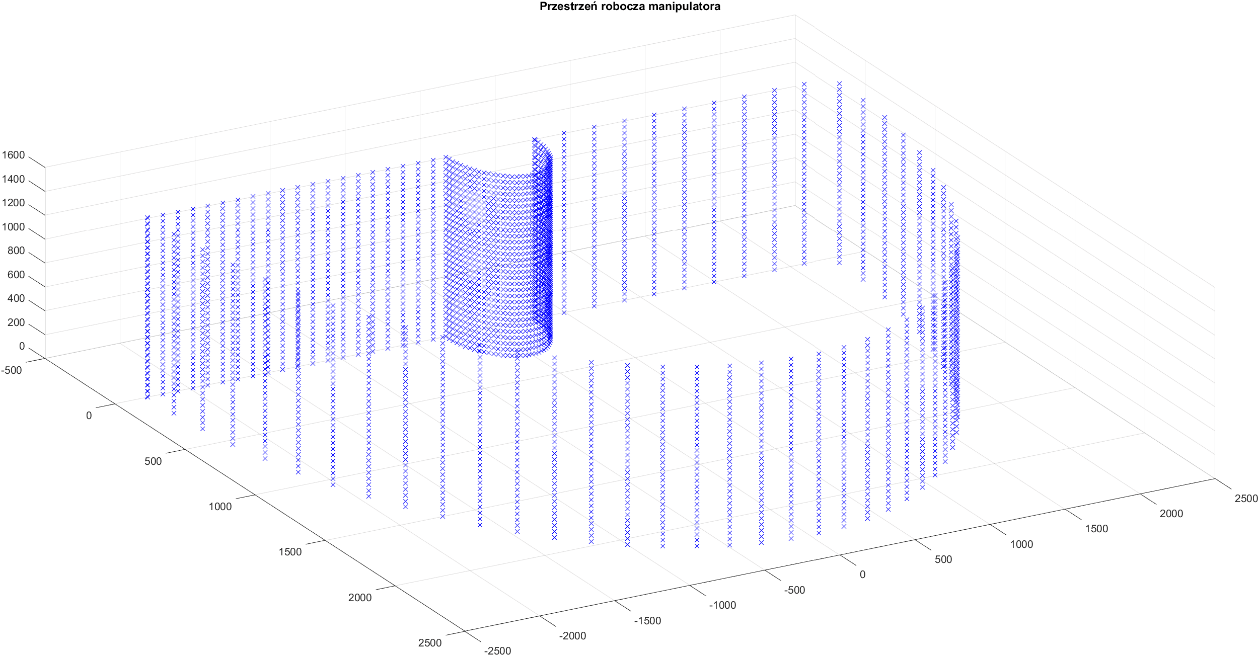


Rys 3.3 Wymiary robota perspektywa boczna

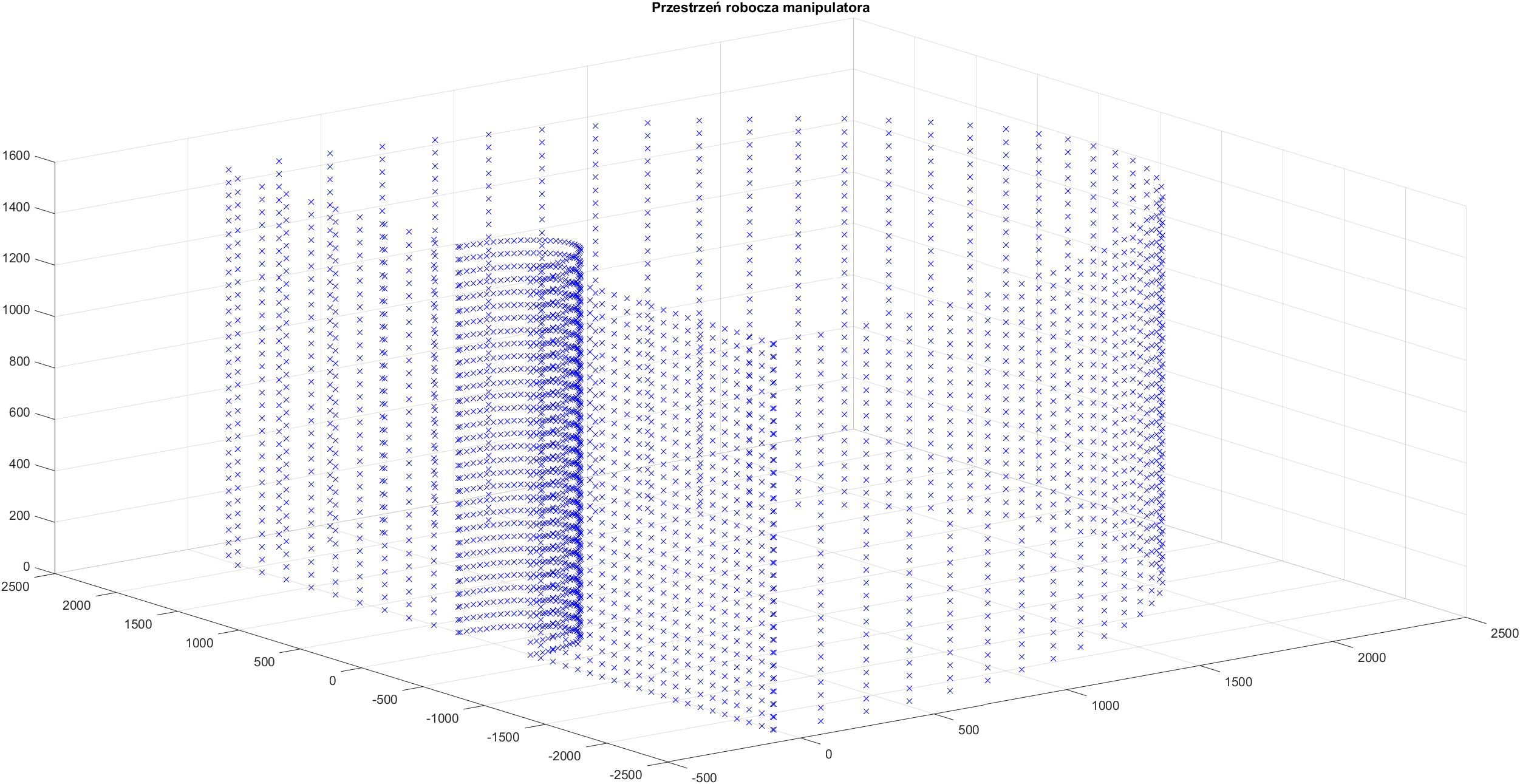
Podczas symulacji w Matlab ustawiliśmy ograniczenia na drugim oraz trzecim złączu na 180 stopni.

1. Przestrzeń robocza

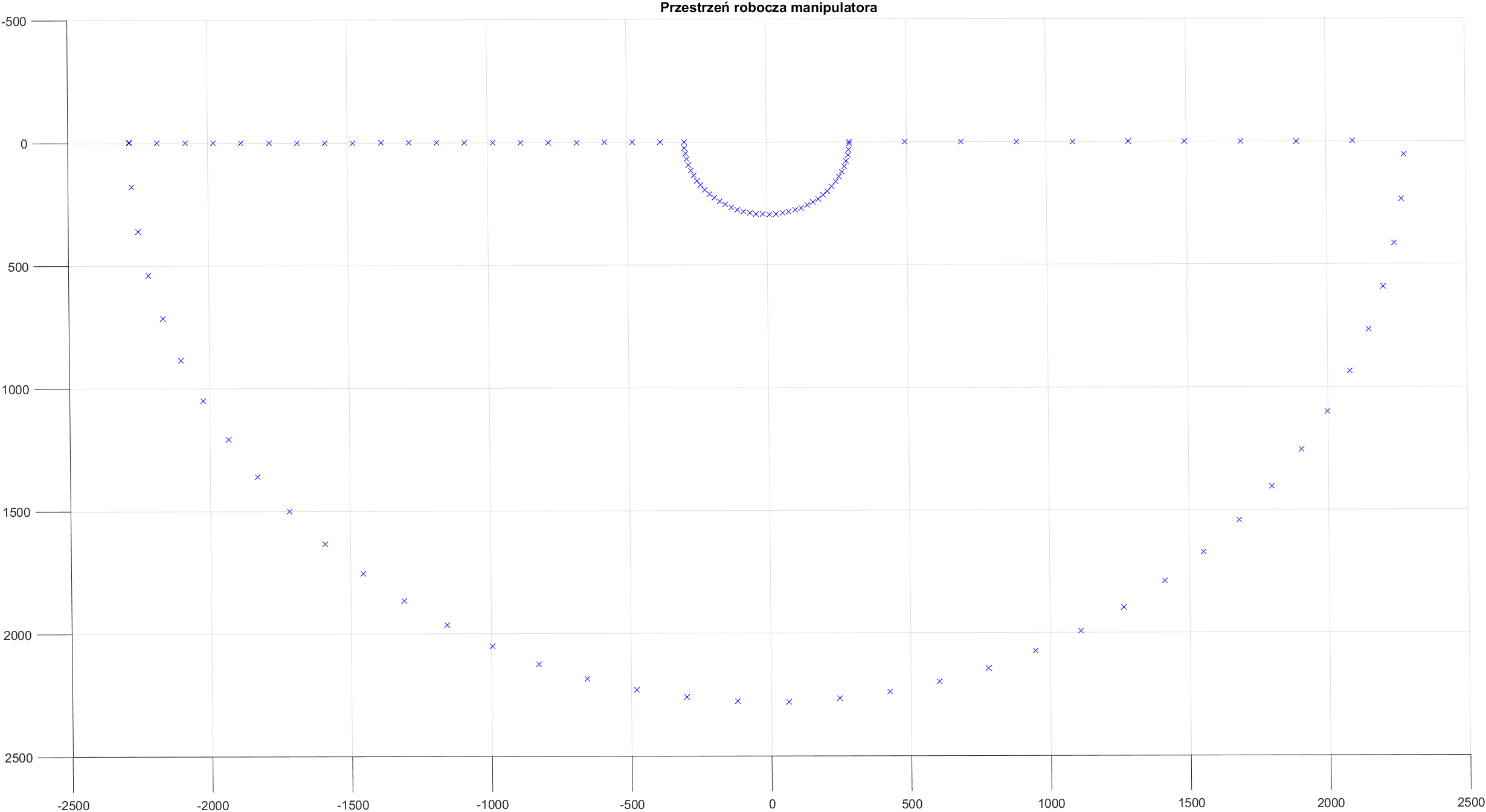
By wyznaczyć przestrzeń roboczą naszego manipulatora POO użyliśmy programu Matlab.



Rys 4.1 Przestrzeń robocza wykonana w programie Matlab od przodu



Rys 4.2 Przestrzeń robocza wykonana w programie Matlab od tyłu



Rys 4.3 Przestrzeń robocza wykonana w programie Matlab od góry

Skrypt programu który pozwolił nam na wyznaczenie przestrzeni roboczej która została przedstawiona powyżej:

P0=[0;0;0;1];

dlugoscRamienia1 = 870;

dlugoscRamienia2 = 1016;

odstepWysokosc = 100; %odstep kropek po wysokosc toru jezdnego

odstepSzerokosc = 50; %odstep kropek po szerokosci/max wychylenie

odstepKat = 0.08; %odstep kropek po kacie

szerokoscRobota = 619; %szerokosc podstawy robota

martwePole = 0.476\*szerokoscRobota; %martwe pole przy podstawie robota, obliczone z tg (rys. dlugosc martwego pola)

torJezdny = 1500; %dlugosc toru jezdnego

maxWyprostowanie = 2280; %dlugosc maksymalnie wyprostowanego robota

obszarRoboczy = maxWyprostowanie - martwePole; %obszar ktory osiaga robot

% macierze

figure(1);

for theta1 = 3.14 : odstepKat : 2\*3.14

for L1 = 0 : odstepSzerokosc : torJezdny

for L2 = martwePole : obszarRoboczy : maxWyprostowanie

% macierze

A12 = [1 0 0 0; 0 1 0 0; 0 0 1 L1; 0 0 0 1];

A11 = [cos(theta1) -sin(theta1) 0 0; sin(theta1) cos(theta1) 0 0; 0 0 1 0; 0 0 0 1];

A1=A11\*A12;

A23 = [1 0 0 0; 0 cos(-pi/2) -sin(-pi/2) 0; 0 sin(-pi/2) cos(-pi/2) 0; 0 0 0 1];

A2=A23;

A3 = [1 0 0 0; 0 1 0 0; 0 0 1 L2; 0 0 0 1];

%

T5=A1\*A2\*A3;

P1=T5\*P0;

x=P1(1,1);

y=P1(2,1);

z=P1(3,1);

plot3(x,y,z,'bx'),grid on, hold on;

end

end

end

x2=0;

for s1= -maxWyprostowanie : odstepWysokosc : -martwePole

%linie proste gora dol

for s2= 0 : odstepSzerokosc : torJezdny

plot3(x2,s1,s2,'bx'),grid on, hold on;

end

end

for s3= martwePole : odstepWysokosc + 100 : maxWyprostowanie

for s4= 0 : odstepSzerokosc : torJezdny

plot3(x2,s3,s4,'bx'),grid on, hold on;

title 'Przestrzeń robocza manipulatora';

end

end

1. Określenie przeniesienia napędu

W zależności od rodzaju energii która wprowadza w ruch nasz manipulator wybraliśmy napędy elektryczne, ponieważ charakteryzują się dużą precyzją, powtarzalnością oraz łatwym sposobem doprowadzenia zasilania.

- Złącze pierwsze

Pierwsze złącze naszego manipulatora jest przesuwne o długości 1500mm

Dzięki zamocowaniu podstawy manipulatora na przesuwnej osi może ten manipulator pracować na większym obszarze.

Do naszego zestawu dobraliśmy napęd firmy DynaGear o przekładni N=12,5 oraz średnicy zębatki napędzającej równej 152mm.

Obraz zawierający niebo, zabawka

Opis wygenerowany automatycznie

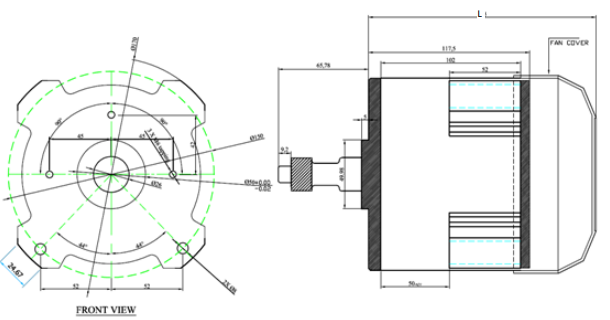
Rys 5.1 Tor jezdny



Rys 5.2 Przekładnia

Jako silnik napędzający dobraliśmy silnik BLDC marki NRControls. Wybrany silnik to silnik bezszczotkowy który charakteryzuje się lepszymi parametrami od silników szczotkowych oraz przez brak szczotek nie wymaga ingerencji co powoduje przestoje w pracy.

Parametry oraz wymiary silnika BLDC 125/2D są podane poniżej:



Rys 5.3 Rzut oraz wymiary silnika

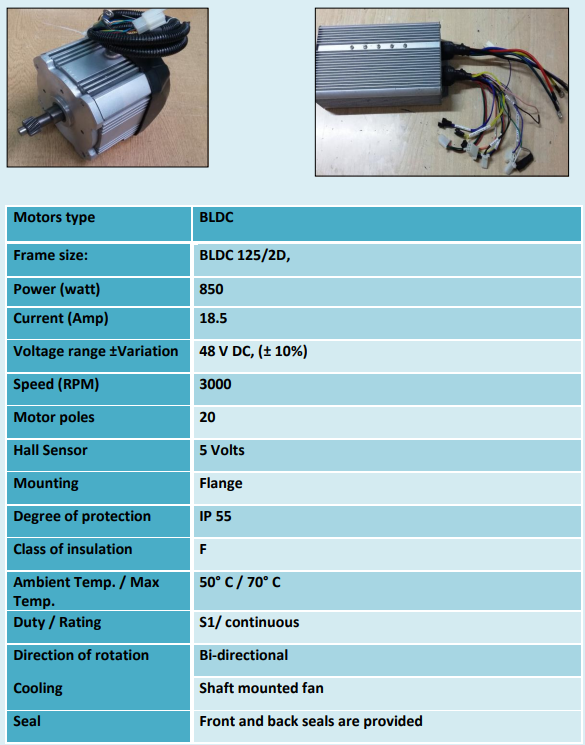
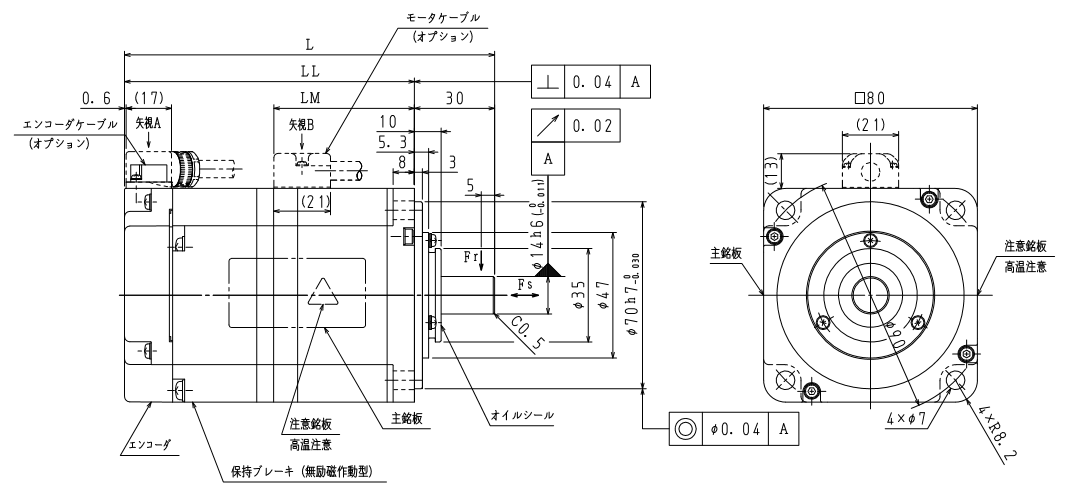


Tabela 5.1 Dane techniczne silnika napędzającego pierwsze złącze

- Złącze obrotowe drugie oraz trzecie

Dla tych złącz wybraliśmy serwonapędy ponieważ posiadają enkodery oraz hamulce

Parametry oraz wymiary serwonapędu Yaskawa SGM7P-02A są podane poniżej:



Rys 5.4 Rzut oraz wymiary silnika

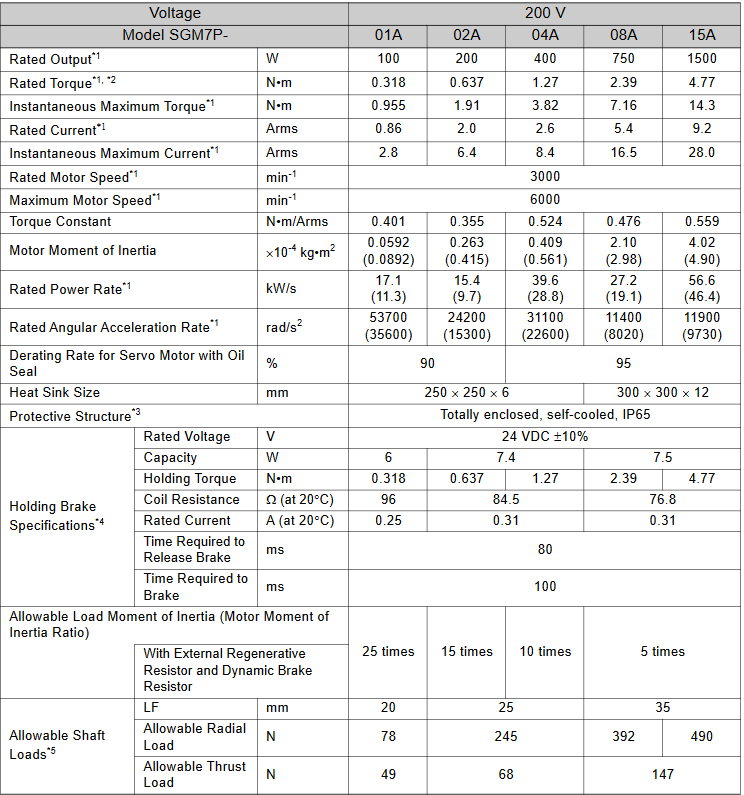
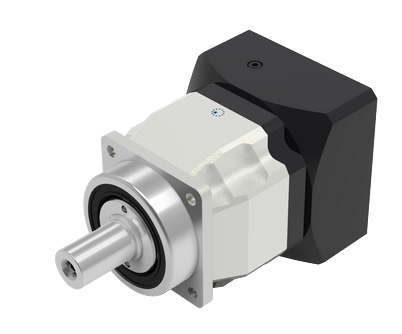


Tabela 5.2 Dane techniczne silnika napędzającego drugie oraz trzecie złącze

Dla drugiego oraz trzeciego złącza zastosowano również przekładnie firmy Apex w celu zwiększenia momentu wyjściowego. Przekładnia z serii AFX o numerze seryjnym AFX060 jest 2 stopniowa ze sprawnością 94%.

Dla drugiego złącza przełożenie wynosi N = 20

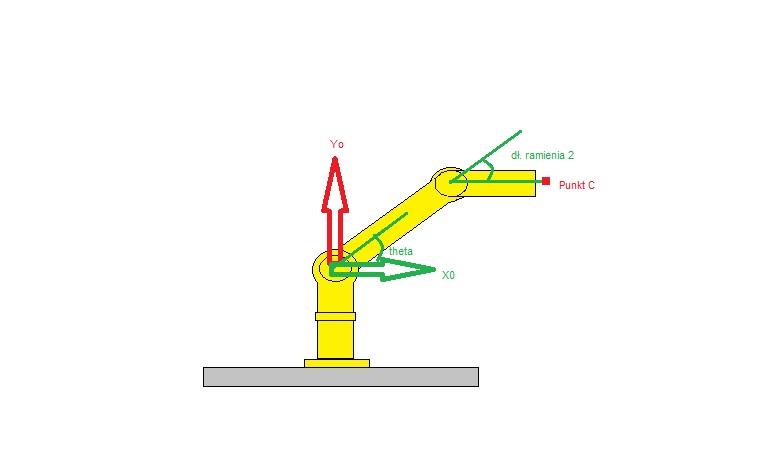
Dla trzeciego złącza przełożenie wynosi N = 70



Rys 5.5 Rysunek przekładni

1. Proste zadanie kinematyki

Dla przedstawionego manipulatora POO można w sposób klasyczny wyznaczyć położenie punktu C. Przyjęto stały układ, nazywany układem bazowym lub odniesienia, względem, którego rozpatruje się wszystkie obiekty łącznie z manipulatorem. Układ ten został zaczepiony w punkcie O0x0y0leżącym w pierwszym przegubie robota. Sposób przyjęcia układu i oznaczenia kątów przedstawiono na rysunku poniżej.



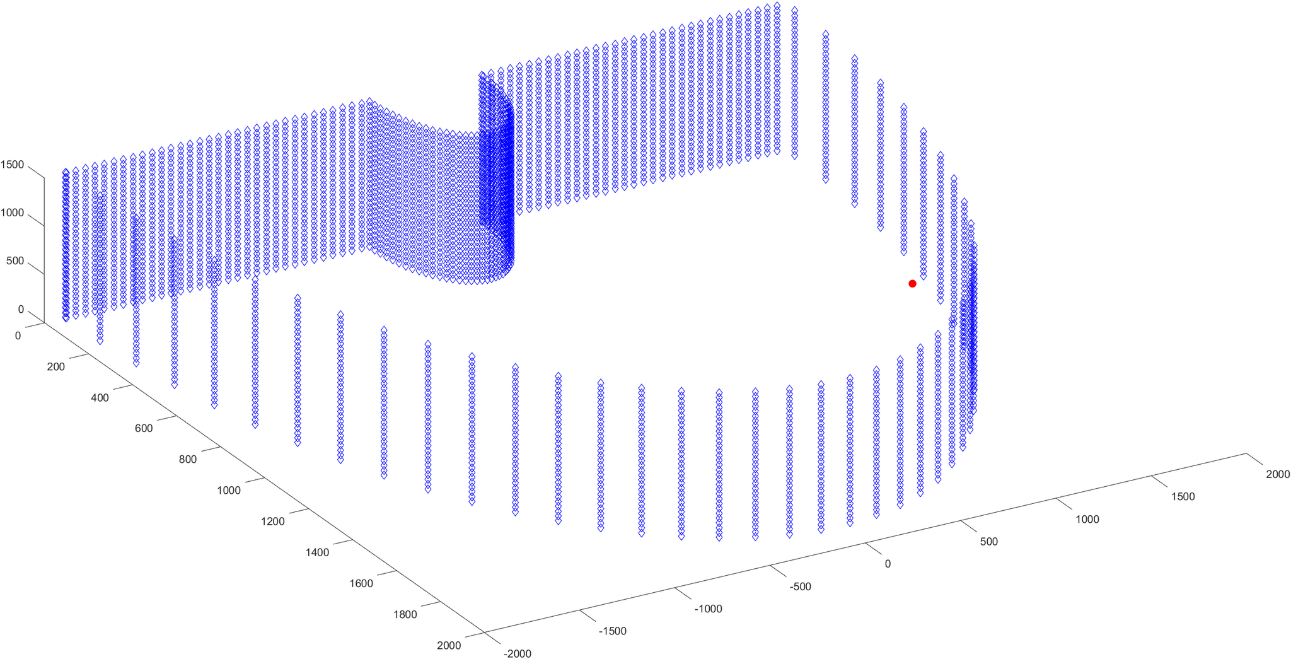
Rys 6.1 Przyjęty układ współrzędnych

Poniżej przedstawiono rysunki wykonane programem Matlab na którym wyznaczono punkt C w przestrzeni robota POO.

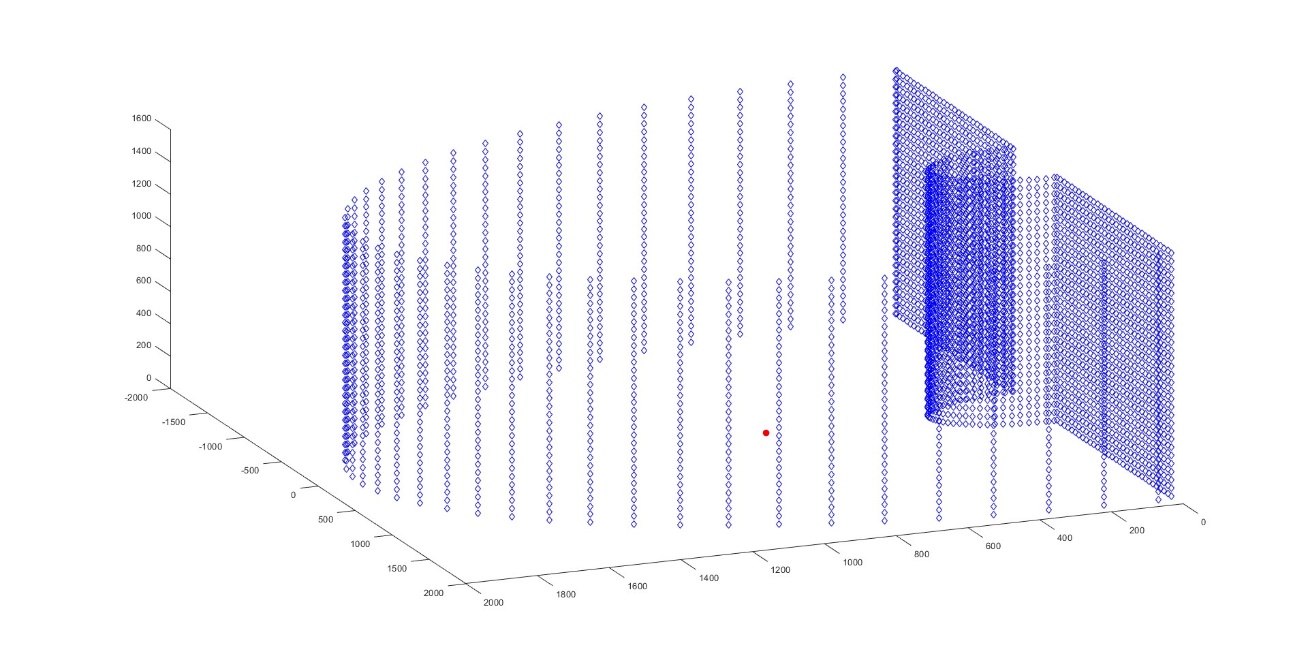
Gdzie:

a1 – długość 1-go ramienia

a2 – długość 2-go ramienia



Rys 6.2 Prosty problem kinematyki zwizualizowany poprzez program Matlab rzut z góry



Rys 6.3 Prosty problem kinematyki zwizualizowany poprzez program Matlab rzut z boku

Skrypt do programu Matlab który wykonuje prosty problem kinematyczny naszego robota POO:

% P0=[0;0;0;1];

% dlugoscRamienia1 = 870;

% dlugoscRamienia2 = 1016;

% odstepWysokosc = 100; %odstep kropek po wysokosc toru jezdnego

% odstepSzerokosc = 50; %odstep kropek po szerokosci/max wychylenie

% odstepKat = 0.08; %odstep kropek po kacie

% szerokoscRobota = 619; %szerokosc podstawy robota

% martwePole = 0.476\*szerokoscRobota; %martwe pole przy podstawie robota, obliczone z tg (rys. dlugosc martwego pola)

% torJezdny = 1500; %dlugosc toru jezdnego

% maxWyprostowanie = 2280; %dlugosc maksymalnie wyprostowanego robota

% obszarRoboczy = maxWyprostowanie - martwePole; %obszar ktory osiaga robot

% % macierze

% figure(1);

% for theta1 = 3.14 : odstepKat : 2\*3.14

% for L1 = 0 : odstepSzerokosc : torJezdny

% for L2 = martwePole : obszarRoboczy : maxWyprostowanie

% % macierze

% A12 = [1 0 0 0; 0 1 0 0; 0 0 1 L1; 0 0 0 1];

% A11 = [cos(theta1) -sin(theta1) 0 0; sin(theta1) cos(theta1) 0 0; 0 0 1 0; 0 0 0 1];

%

% A1=A11\*A12;

% A23 = [1 0 0 0; 0 cos(-pi/2) -sin(-pi/2) 0; 0 sin(-pi/2) cos(-pi/2) 0; 0 0 0 1];

% A2=A23;

%

% A3 = [1 0 0 0; 0 1 0 0; 0 0 1 L2; 0 0 0 1];

% %

% T5=A1\*A2\*A3;

%

% P1=T5\*P0;

% x=P1(1,1);

% y=P1(2,1);

% z=P1(3,1);

% plot3(x,y,z,'bx'),grid on, hold on;

% end

% end

% end

% x2=0;

% for s1= -maxWyprostowanie : odstepWysokosc : -martwePole

% %linie proste gora dol

% for s2= 0 : odstepSzerokosc : torJezdny

% plot3(x2,s1,s2,'bx'),grid on, hold on;

%

% end

% end

%

% for s3= martwePole : odstepWysokosc + 100 : maxWyprostowanie

% for s4= 0 : odstepSzerokosc : torJezdny

% plot3(x2,s3,s4,'bx'),grid on, hold on;

% title 'Przestrzeń robocza manipulatora';

% end

% end

katpierwszy =input('kąt obrotu 1 (od -90 do 90 stopni) = ');

katdrugi =input('kąt obrotu 2 (-150 do 150 stopni) = ');

wysokosc =input('wysokosc max 1500[mm] = ');

l1 = 870;

l2 = 1016;

katpierwszy =katpierwszy\*3.14/180; %radiany

martwePole = 0.476\*619; %martwe pole przy podstawie robota, obliczone z tg (rys. dlugosc martwego pola)

torJezdny = 1500; %dlugosc toru jezdnego

maxWyprostowanie = 1886; %dlugosc maksymalnie wyprostowanego robota

katdrugi=katdrugi\*3.14/180;

% wzór przedstawiony w sprawozdaniu do wyznaczenia punktu

x1=l1\*cos((katpierwszy))+l2\*cos((katpierwszy+katdrugi)); y1=l1\*sin((katpierwszy))+l2\*sin((katpierwszy+katdrugi));

z1=wysokosc;

r1=sqrt((x1\*x1)+(y1\*y1));

if (r1<martwePole || r1>maxWyprostowanie || x1<0)

display('Wartości x i y poza zakresem! Proszę podać inne wartości!');

x1=input('Punkt początkowy wektora odejścia x: ');

y1=input('Punkt początkowy wektora odejścia y: ');

end;

r1=sqrt((x1\*x1)+(y1\*y1));

if (r1<martwePole || r1>maxWyprostowanie || x1<0)

error('bledne wartosci!');

end;

P0=[0;0;0;1];

% macierze(przestrzeń robocza)

figure(1);

for theta1 = 3.14 : 0.08 : 2\*3.14

for L1 = 0 : 50 : 1500

for L2 = 294 : 1592 : 1886

% macierze

A12 = [1 0 0 0; 0 1 0 0; 0 0 1 L1; 0 0 0 1];

A11 = [cos(theta1) -sin(theta1) 0 0; sin(theta1) cos(theta1) 0 0; 0 0 1 0; 0 0 0 1];

A1=A11\*A12;

A23 = [1 0 0 0; 0 cos(-pi/2) -sin(-pi/2) 0; 0 sin(-pi/2) cos(-pi/2) 0; 0 0 0 1];

A2=A23;

A3 = [1 0 0 0; 0 1 0 0; 0 0 1 L2; 0 0 0 1];

%

T5=A1\*A2\*A3;

P1=T5\*P0;

x=P1(1,1);

y=P1(2,1);

z=P1(3,1);

plot3(x,y,z,'bd'),hold on;

end

end

end

x2=0;

for s1= -1886 : 50 : -294

for s2= 0 : 50 : 1500

plot3(x2,s1,s2,'bd'),hold on;

end

end

for s3= 294 : 50 : 1886

for s4= 0 : 50 : 1500

plot3(x2,s3,s4,'bd'),hold on;

end

end

plot3(x1,y1,z1,'.','MarkerSize',25,'MarkerEdge','r'),

hold on;