

**Podstawy Robotyki**

**Laboratorium**

**seria I**

**ćwiczenie 2**

**Wyznaczenie równań kinematyki prostej układu manipulacyjnego**

LP1 C1, EADI-2

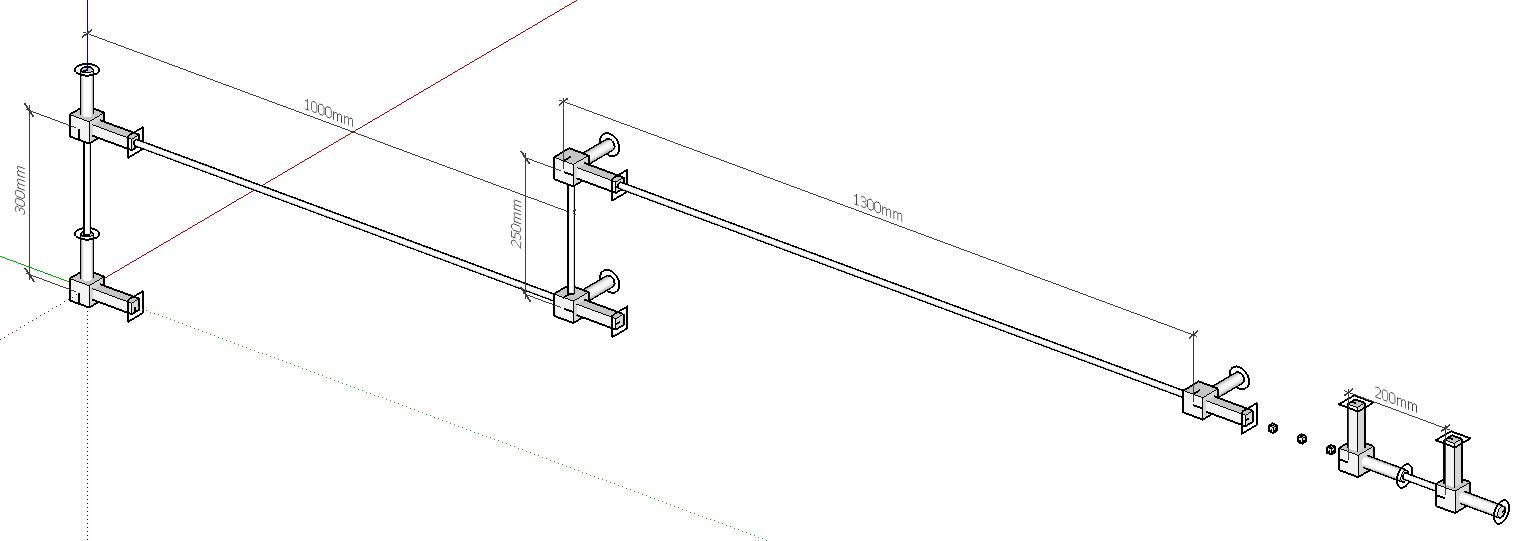
Bazan Arkadiusz

Bełch Jakub

Bizior Mateusz

Rzeszów, X 2015

1. Schemat robota:



os X -zakonczona kwadratem

os Z - zakonczona kolkiem

os Y nie zaznaczona na rysunku, zgodna z regula prawej reki

1. Dane wejsciowe

tabelka DHnum z wartościami konstrukcyjnymi robota:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Θ | d | a | α |
| 1 | Θ1,var | 300 | 0 | 90 |
| 2 | Θ2,var | 0 | 1000 | 0 |
| 3 | Θ3,var | 0 | 250 | 0 |
| 4 | Θ4,var | 0 | 1300 | 0 |
| 5 | Θ5,var | 0 | 0 | 90 |
| 6 | Θ6,var | 200 | 0 | 0 |

tabelka DHsym z wartościami symbolicznymi:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Θ | d | a | α |
| 1 | Θ1,var | d1 | 0 | alfa1 |
| 2 | Θ2,var | 0 | a2 | 0 |
| 3 | Θ3,var | 0 | a3 | 0 |
| 4 | Θ4,var | 0 | a4 | 0 |
| 5 | Θ5,var | 0 | 0 | alfa5 |
| 6 | Θ6,var | d6 | 0 | 0 |

wczytanie danych do matlaba za pomoca skryptu:

%dane wejsciowe z tabelek

%dane symboliczne

clear; %czyszczenie workspace

stopni=180/pi;radianow=pi/180;

syms th1 th2 th3 th4 th5 th6 d1 d6 a2 a3 a4 alfa1 alfa5;

%DHmacierz z danymi symbolicznymi

DHsym=[ th1 d1 sym('0') alfa1;

th2 sym('0') a2 sym('0');

th3 sym('0') a3 sym('0');

th4 sym('0') a4 sym('0');

th5 sym('0') sym('0') alfa5;

th6 d6 sym('0') sym('0') ];

%macierz z qi (jak thi,var to w 1 kolumnie thi, jak di,var to w drugiej kolumnie di

DHvar=[ th1 0;

th2 0;

th3 0;

th4 0;

th5 0;

th6 0 ];

%macierz z danymi robota

DHnum=[ th1, 300, 0, pi/2;

th2, 0, 1000, 0;

th3, 0, 250, 0;

th4, 0, 1300, 0;

th5, 0, 0, pi/2;

th6, 200, 0, 0 ];

fprintf('wczytane dane cwiczenia 2...')

fprintf('\t\t\t[ OK ]\n')

1. Funkcje pomocnicze:

Przed przystąpieniem do zadania aby oszczędzić sobie czas i ułatwić prace napisaliśmy sobie funkcje **AmatrixSym** generujaca macierze A0-A6 na podst. DHmatrixsym:

function A=AmatrixSym(DHmatrixSym)

%wejscie maceirz DH, format wiersza: th d a alfa, wyjscie macierz3d macierzy Ai,

%zmienne pomocnicze

fprintf('symboliczne wyznaczanie macierzy Ai...')

A=sym(zeros(4,4));

n=size(DHmatrixSym,1);

for i =1: n

th=DHmatrixSym(i,1);

d=DHmatrixSym(i,2);

a=DHmatrixSym(i,3);

alfa=DHmatrixSym(i,4);

% glowny kod

A(:,:,i)=[ cos(th), -cos(alfa)\*sin(th), sin(alfa)\*sin(th), a\*cos(th);

sin(th), cos(alfa)\*cos(th), -sin(alfa)\*cos(th), a\*sin(th);

0, sin(alfa), cos(alfa), d;

0, 0, 0, 1];

fprintf('.')

end

fprintf('[ OK ]\n')

end

oraz **TmatrixSym** generujaca macierze T01-T06:

function T=TmatrixSym(AmatrixSym)

% wejscie macierz A z fcji Amatrix, wyjscie macierz3d macierzy Ti,

%

% zmienne pomocnicze

fprintf('symboliczne wyznaczanie macierzy Ti...')

T=sym(zeros(4,4));

T(:,:,1)=AmatrixSym(:,:,1);

%pomijam kontrole ilosci przegubow, zakladam ze wieksza od 2

n=size(AmatrixSym,3);

for i =2: n

T(:,:,i)=T(:,:,i-1)\*AmatrixSym(:,:,i);

T(:,:,i)=simplify(T(:,:,i));

fprintf('.')

end

fprintf('[ OK ]\n')

end

oraz **PmatrixSym** wyodrebniajaca wspolrzedne kartezjanskie wtkrow z macierzy Ti:

function Psym=PmatrixSym(TmatrixSym)

% wejscie macierz3D macierzy T, wyjscie macierz wektorw P,

%

% zmienne pomocnicze

fprintf('symboliczne wyznaczanie wektorow polozenia Pi...')

n=size(TmatrixSym,3);

Psym=sym(zeros(1,3));

%pomijam kontrole ilosci przegubow, zakladam ze wieksza od 1

for i =1: n

for j=1:3

Psym(i,j)=TmatrixSym(j,4,i);

fprintf('.')

end

end

simplify(Psym);

fprintf('[ OK ]\n')

end

analogicznie dla orientacji **BmatrixSym** macierze Bi:

function B=BmatrixSym(TmatrixSym)

% wejscie macierz3D macierzy T, wyjscie macierz wektorw P,

%

% zmienne pomocnicze

fprintf('symboliczne wyznaczanie tensorów? obrotu Bi...')

n=size(TmatrixSym,3);

B=[sym('0'),sym('0'),sym('0');

sym('0'),sym('0'),sym('0');

sym('0'),sym('0'),sym('0')];

%pomijam kontrole ilosci przegubow, zakladam ze wieksza od 1

for i =1: n

for j=1:3

for k=1:3

B(j,k,i)=TmatrixSym(j,k,i);

fprintf('.')

end

end

end

simplify(B);

fprintf('[ OK ]\n')

end

i dla katow Eulera **katyEuleraSym**

function Eul=katyEuleraSym(Tsym)

%wejscie macierz3D macierzy T, wyjscie macierz katow eulera Eul,

%

%zmienne pomocnicze

fprintf('symboliczne wyznaczanie katow Eulera...')

n=size(Tsym,3);

Eul=sym(zeros(1,3));

%pomijam kontrole ilosci przegubow, zakladam ze wieksza od 1

for i =1: n

%format katow => [fi; th; psi]

ax=Tsym(1,3,i); ay=Tsym(2,3,i); az=Tsym(3,3,i);

nx=Tsym(1,1,i); ny=Tsym(2,1,i);

ox=Tsym(1,2,i); oy=Tsym(2,2,i);

fi=atan2(ay,az);

th=atan2(cos(fi)\*ax+sin(fi)\*ay,az);

psi=atan2(-sin(fi)\*nx+cos(fi)\*ny, -sin(fi)\*ox+cos(fi)\*oy);

%fi=simplify(fi);

Eul(i,1)=fi; fprintf('.')

%th=simplify(th);

Eul(i,2)=th; fprintf('.')

%psi=simplify(psi);

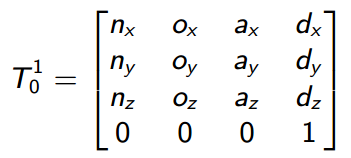
Eul(i,3)=psi; fprintf('.')

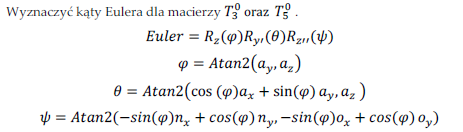
end

fprintf('[ OK ]\n')

end

wg tabelki:

****

****

ostatecznie przygotowana funkcja kinDirXXXSym przybrala postac:

function [Asym,Tsym,Psym,Bsym,Eulsym]=kinDirXXXSym(DHsym)

% funkcja Kuby Miszcza:] (2015)

%DHsym=>macierz DH symboliczna

%symboliczne tworzenie A i T i P(macierzy wketorow)

Asym=AmatrixSym(DHsym); %symboliczne wyznaczanie Ai

Tsym=TmatrixSym(Asym); %symboliczne wyznaczanie Ti

Psym=PmatrixSym(Tsym); %symboliczne wyznaczanie wketorow polozenia Pi

Bsym=BmatrixSym(Tsym); %symboliczne macierz katow Bi

Eulsym=katyEuleraSym(Tsym); %symboliczne katy eulera Ei

end

Za jej pomoca latwo i szybko możemy sobie wyznaczyc polozenie dla zadanych katów.

1. Przebieg ćwiczenia.
   1. macierze Ai

>> Asym

Asym(:,:,1) =

[ cos(th1), -cos(alfa1)\*sin(th1), sin(alfa1)\*sin(th1), 0]

[ sin(th1), cos(alfa1)\*cos(th1), -sin(alfa1)\*cos(th1), 0]

[ 0, sin(alfa1), cos(alfa1), d1]

[ 0, 0, 0, 1]

Asym(:,:,2) =

[ cos(th2), -sin(th2), 0, a2\*cos(th2)]

[ sin(th2), cos(th2), 0, a2\*sin(th2)]

[ 0, 0, 1, 0]

[ 0, 0, 0, 1]

Asym(:,:,3) =

[ cos(th3), -sin(th3), 0, a3\*cos(th3)]

[ sin(th3), cos(th3), 0, a3\*sin(th3)]

[ 0, 0, 1, 0]

[ 0, 0, 0, 1]

Asym(:,:,4) =

[ cos(th4), -sin(th4), 0, a4\*cos(th4)]

[ sin(th4), cos(th4), 0, a4\*sin(th4)]

[ 0, 0, 1, 0]

[ 0, 0, 0, 1]

Asym(:,:,5) =

[ cos(th5), -cos(alfa5)\*sin(th5), sin(alfa5)\*sin(th5), 0]

[ sin(th5), cos(alfa5)\*cos(th5), -sin(alfa5)\*cos(th5), 0]

[ 0, sin(alfa5), cos(alfa5), 0]

[ 0, 0, 0, 1]

Asym(:,:,6) =

[ cos(th6), -sin(th6), 0, 0]

[ sin(th6), cos(th6), 0, 0]

[ 0, 0, 1, d6]

[ 0, 0, 0, 1]

T03

>> Tsym(:,:,3)

[ cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)), - cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) - sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)), sin(alfa1)\*sin(th1), a2\*cos(th1)\*cos(th2) - a3\*sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + a3\*cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - a2\*cos(alfa1)\*sin(th1)\*sin(th2)]

[ cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)), - cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) - sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)), -sin(alfa1)\*cos(th1), a2\*cos(th2)\*sin(th1) - a3\*sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + a3\*cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) + a2\*cos(alfa1)\*cos(th1)\*sin(th2)]

[ sin(th2 + th3)\*sin(alfa1), cos(th2 + th3)\*sin(alfa1), cos(alfa1), d1 + a2\*sin(alfa1)\*sin(th2) + a3\*sin(alfa1)\*cos(th2)\*sin(th3) + a3\*sin(alfa1)\*cos(th3)\*sin(th2)]

[ 0, 0, 0, 1]

**T05**

>> Tsym(:,:,5)

[ cos(th5)\*(cos(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1))) - sin(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)))) - sin(th5)\*(cos(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2))) + sin(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)))), sin(alfa1)\*sin(alfa5)\*sin(th1) - cos(alfa5)\*sin(th5)\*(cos(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1))) - sin(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)))) - cos(alfa5)\*cos(th5)\*(cos(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2))) + sin(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)))), sin(alfa5)\*cos(th5)\*(cos(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2))) + sin(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)))) + sin(alfa5)\*sin(th5)\*(cos(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1))) - sin(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)))) + cos(alfa5)\*sin(alfa1)\*sin(th1), a4\*cos(th4)\*(cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1))) - a3\*sin(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) - a4\*sin(th4)\*(cos(th3)\*(cos(th1)\*sin(th2) + cos(alfa1)\*cos(th2)\*sin(th1)) + sin(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2))) + a2\*cos(th1)\*cos(th2) + a3\*cos(th3)\*(cos(th1)\*cos(th2) - cos(alfa1)\*sin(th1)\*sin(th2)) - a2\*cos(alfa1)\*sin(th1)\*sin(th2)]

[ cos(th5)\*(cos(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2))) - sin(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)))) - sin(th5)\*(cos(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2))) + sin(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)))), - cos(alfa5)\*cos(th5)\*(cos(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2))) + sin(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)))) - cos(alfa5)\*sin(th5)\*(cos(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2))) - sin(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)))) - sin(alfa1)\*sin(alfa5)\*cos(th1), sin(alfa5)\*cos(th5)\*(cos(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2))) + sin(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)))) + sin(alfa5)\*sin(th5)\*(cos(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2))) - sin(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)))) - cos(alfa5)\*sin(alfa1)\*cos(th1), a4\*cos(th4)\*(cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) - sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2))) - a3\*sin(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) - a4\*sin(th4)\*(cos(th3)\*(sin(th1)\*sin(th2) - cos(alfa1)\*cos(th1)\*cos(th2)) + sin(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2))) + a2\*cos(th2)\*sin(th1) + a3\*cos(th3)\*(cos(th2)\*sin(th1) + cos(alfa1)\*cos(th1)\*sin(th2)) + a2\*cos(alfa1)\*cos(th1)\*sin(th2)]

[ sin(th2 + th3 + th4 + th5)\*sin(alfa1), cos(alfa1)\*sin(alfa5) - sin(th2 + th3 + th4)\*cos(alfa5)\*sin(alfa1)\*sin(th5) + cos(th2 + th3 + th4)\*cos(alfa5)\*sin(alfa1)\*cos(th5), cos(alfa1)\*cos(alfa5) - cos(th2 + th3 + th4)\*sin(alfa1)\*sin(alfa5)\*cos(th5) + sin(th2 + th3 + th4)\*sin(alfa1)\*sin(alfa5)\*sin(th5), d1 + a2\*sin(alfa1)\*sin(th2) + a4\*cos(th2 + th3)\*sin(alfa1)\*sin(th4) + a4\*sin(th2 + th3)\*sin(alfa1)\*cos(th4) + a3\*sin(alfa1)\*cos(th2)\*sin(th3) + a3\*sin(alfa1)\*cos(th3)\*sin(th2)]

[ 0, 0, 0, 1]

1. wyjaśnić i po wiedziec co zrobilem i czy dobrze?:

