Univerza v Ljubljani
Fakulteta za elektrotehniko in računalništvo
Tržaška 25
61000 Ljubljana

# Poročilo z Laboratorijskih vaj pri predmetu Robotika 1 Matrika JACOBI za robot PUMA

Simon MALI

22. februar 1996

## 1 Naloga

Najprej po D-H analitično izračunajte kinematiko robota PUMA. Pri tem izhajajte iz postavitve koordinatnih sistemov kot jih prikazuje priložena skica. Iz izpisa morajo biti razvidne matrike A in matrika $T_6$. Izračunajte tudi matrike $T_6$. Analitično izračunajte Jacobijevo matriko in pri tem uporabite algoritem, ki ga je podal prof. Paul.

Poročilo:

- D-H izračun
- postopek J izračuna
- J rezultat

## 2 HDH Parametri robota

$$DH = Rot(z,\theta)Trans(z,d)Trans(x,a)Rot(x,\alpha)$$

Slika 1: Skica robota

| $A_n$ | $\theta$ | d | a | $\alpha$ |
|---|---|---|---|---|
| 1 | $\theta_1$ | 0 | 0 | $-\pi/2$ |
| 2 | $\theta_2$ | $d_2$ | $a_2$ | 0 |
| 3 | $\theta_3+\pi/2$ | 0 | $-a_3$ | $\pi/2$ |
| 4 | $\theta_4$ | $d_4$ | 0 | $-\pi/2$ |
| 5 | $\theta_5$ | 0 | 0 | $\pi/2$ |
| 6 | $\theta_6$ | $d_6$ | 0 | $\pi/2$ |

$$A_n = \begin{bmatrix} \cos\theta_n & -\cos\alpha_n\sin\theta_n & \sin\alpha_n\sin\theta_n & a_n\cos\theta_n \\ \sin\theta_n & \cos\alpha_n\cos\theta_n & sin\alpha_n\cos\theta_n & a_n\sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# 3 Postopek izračuna matrike J

Naš cilj je izračun $dT_6$; torej ${}^{T_6}\Delta_i$ v odvisnosti od $dq_i$. Ker ima robot 6 prostostnih stopenj je dimenzija te matrike $1\times6\times6$. Ta matrika popisuje spremembo koordinat vrha v odvisnosti od spremembe notranjih koordinat. Običajno nam izračun te matrike ne predstavlja večji problem. Povsem drugače pa je z izračunom inverzne kinematike, ki za bolj zapletene mehanizme včasih tudi ni rešljiva(singularnosti).

$$J\cdot dq_i = \begin{vmatrix} {}^{T_6}d_x \\ {}^{T_6}d_y \\ {}^{T_6}d_z \\ {}^{T_6}\delta_x \\ {}^{T_6}\delta_y \\ {}^{T_6}\delta_z \end{vmatrix} = \begin{bmatrix} {}^{T_6}d_{1x} & {}^{T_6}d_{2x} & \cdots & \cdots & \cdots & {}^{T_6}d_{6x} \\ {}^{T_6}d_{1y} & \ddots & & & & \vdots \\ {}^{T_6}d_{1z} & & \ddots & & & \vdots \\ {}^{T_6}\delta_{1x} & & & \ddots & & \vdots \\ {}^{T_6}\delta_{1y} & & & & \ddots & \vdots \\ {}^{T_6}\delta_{1z} & \cdots & \cdots & \cdots & \cdots & {}^{T_6}\delta_{6z} \end{bmatrix} \cdot \begin{vmatrix} dq_1 \\ dq_2 \\ dq_3 \\ dq_4 \\ dq_5 \\ dq_6 \end{vmatrix} \tag{1}$$

ali krajše

$$\begin{vmatrix} {}^{T_6}d \\ {}^{T_6}\delta \end{vmatrix} = |J||dq_i| \tag{2}$$

Za rotacijski sklep so diferencjalni pomiki in zasuki

$${}^{T_6}d_{ix} = -n_xp_y + n_yp_x \tag{3}$$

$${}^{T_6}d_{iy} = -o_xp_y + o_yp_x \tag{4}$$

$${}^{T_6}d_{iz} = -a_xp_y + a_yp_x \tag{5}$$

$${}^{T_6}\delta_{ix} = n_z \tag{6}$$

$${}^{T_6}\delta_{iy} = n_y \tag{7}$$

$${}^{T_6}\delta_{iz} = n_z \tag{8}$$

Prizmatski sklepi imajo $\delta = 0$

# 4 Program

Program izdelan v programskem jeziku *Mathematica izračuna kinematiko robota PUMA.*

```
Needs["Geometry`Robotics`"];
A1 = HDHMatrix[theta1, 0, 0, -Pi/2];
A2 = HDHMatrix[theta2, d2, a2, 0];
A3 = HDHMatrix[theta3 + Pi/2, 0, -a3, Pi/2];
A4 = HDHMatrix[theta4, d4, 0, -Pi/2];
A5 = HDHMatrix[theta5, 0, 0, Pi/2];
A6 = HDHMatrix[theta6, d6, 0, 0];

T60 = A1.A2.A3.A4.A5.A6;
T61 = A2.A3.A4.A5.A6;
T62 = A3.A4.A5.A6;
T63 = A4.A5.A6;
T64 = A5.A6;
T65 = A6;


Jaco[x_] := Block[ {},
nx = x[[1,1]]; ox = x[[1,2]]; ax = x[[1,3]]; px = x[[1,4]];
ny = x[[2,1]]; oy = x[[2,2]]; ay = x[[2,3]]; py = x[[2,4]];
nz = x[[3,1]]; oz = x[[3,2]]; az = x[[3,3]]; pz = x[[3,4]];
Q = {};
tdx = px * ny - py * nx;
tdy = oy * px - ox * py;
tdz = ay * px - ax * py;
ddx = nz;
ddy = oz;
ddz = az;

Q = Append[Q, tdx];
Q = Append[Q, tdy];
Q = Append[Q, tdz];
Q = Append[Q, ddx];
Q = Append[Q, ddy];
Q = Append[Q, ddz];
Q = SimplifyTrig[Expand[Q]]
]

Print["Podmatrike ..."];

J1v = Jaco[T60];
J2v = Jaco[T61];
J3v = Jaco[T62];
J4v = Jaco[T63];
J5v = Jaco[T64];
J6v = Jaco[T65];

Print["Transponiram vrstice v stolpce..."];

J1 = Transpose[{J1v}];
J2 = Transpose[{J2v}];
J3 = Transpose[{J3v}];
J4 = Transpose[{J4v}];
J5 = Transpose[{J5v}];
J6 = Transpose[{J6v}];
```

```
Print[Šestavljam Jacobi matriko..."];
J = {};
J = Append[J, J1];
J = Append[J, J2];
J = Append[J, J3];
J = Append[J, J4];
J = Append[J, J5];
J = Append[J, J6];

Print["Poenostavljam Jacobi matriko ..."];

J = SimplifyTrig[J];

Print["Izpis"];

Save["j1.prn", J1];
Save["j2.prn", J2];
Save["j3.prn", J3];
Save["j4.prn", J4];
Save["j5.prn", J5];
Save["j6.prn", J6];
Save["j.prn", J];

J1t = TeXForm[J1];
J2t = TeXForm[J2];
J3t = TeXForm[J3];
J4t = TeXForm[J4];
J5t = TeXForm[J5];
J6t = TeXForm[J6];
Jt  = TeXForm[J]

Save["j1.TeX", J1t];
Save["j2.TeX", J2t];
Save["j3.TeX", J3t];
Save["j4.TeX", J4t];
Save["j5.TeX", J5t];
Save["j6.TeX", J6t];
Save["j.TeX", Jt];

Print["Konec"];
```

# 5 Jacobijeva matrika

*Izpisana je skupna Jacobijeva matrika z rezultatom kot ga je izpisal paket Mathematica*

```
J = {{{-(d2*Cos[Pi/2 + theta2 + theta3]*Cos[theta4]*Cos[theta5]*
         Cos[theta6]) + a2*Cos[theta2]*Cos[theta5]*Cos[theta6]*Sin[theta4] -
      a3*Cos[Pi/2 + theta2 + theta3]*Cos[theta5]*Cos[theta6]*Sin[theta4] +
      d6*Cos[theta6]*Sin[Pi/2 + theta2 + theta3]*Sin[theta4] +
      d4*Cos[theta5]*Cos[theta6]*Sin[Pi/2 + theta2 + theta3]*Sin[theta4] +
      d2*Cos[theta6]*Sin[Pi/2 + theta2 + theta3]*Sin[theta5] +
```

```
  a2*Cos[theta2]*Cos[theta4]*Sin[theta6] -
  a3*Cos[Pi/2 + theta2 + theta3]*Cos[theta4]*Sin[theta6] +
  d4*Cos[theta4]*Sin[Pi/2 + theta2 + theta3]*Sin[theta6] +
  d6*Cos[theta4]*Cos[theta5]*Sin[Pi/2 + theta2 + theta3]*Sin[theta6] +
  d2*Cos[Pi/2 + theta2 + theta3]*Sin[theta4]*Sin[theta6] +
  d6*Cos[Pi/2 + theta2 + theta3]*Sin[theta5]*Sin[theta6]},
 {a2*Cos[theta2]*Cos[theta4]*Cos[theta6] -
  a3*Cos[Pi/2 + theta2 + theta3]*Cos[theta4]*Cos[theta6] +
  d4*Cos[theta4]*Cos[theta6]*Sin[Pi/2 + theta2 + theta3] +
  d6*Cos[theta4]*Cos[theta5]*Cos[theta6]*Sin[Pi/2 + theta2 + theta3] +
  d2*Cos[Pi/2 + theta2 + theta3]*Cos[theta6]*Sin[theta4] +
  d6*Cos[Pi/2 + theta2 + theta3]*Cos[theta6]*Sin[theta5] +
  d2*Cos[Pi/2 + theta2 + theta3]*Cos[theta4]*Cos[theta5]*Sin[theta6] -
  a2*Cos[theta2]*Cos[theta5]*Sin[theta4]*Sin[theta6] +
  a3*Cos[Pi/2 + theta2 + theta3]*Cos[theta5]*Sin[theta4]*Sin[theta6] -
  d6*Sin[Pi/2 + theta2 + theta3]*Sin[theta4]*Sin[theta6] -
  d4*Cos[theta5]*Sin[Pi/2 + theta2 + theta3]*Sin[theta4]*Sin[theta6] -
  d2*Sin[Pi/2 + theta2 + theta3]*Sin[theta5]*Sin[theta6]},
 {-(d2*Cos[theta5]*Sin[Pi/2 + theta2 + theta3]) -
  d2*Cos[Pi/2 + theta2 + theta3]*Cos[theta4]*Sin[theta5] +
  a2*Cos[theta2]*Sin[theta4]*Sin[theta5] -
  a3*Cos[Pi/2 + theta2 + theta3]*Sin[theta4]*Sin[theta5] +
  d4*Sin[Pi/2 + theta2 + theta3]*Sin[theta4]*Sin[theta5]},
 {-(Cos[theta4]*Cos[theta5]*Cos[theta6]*Sin[Pi/2 + theta2 + theta3]) -
  Cos[Pi/2 + theta2 + theta3]*Cos[theta6]*Sin[theta5] +
  Sin[Pi/2 + theta2 + theta3]*Sin[theta4]*Sin[theta6]},
 {Cos[theta6]*Sin[Pi/2 + theta2 + theta3]*Sin[theta4] +
  Cos[theta4]*Cos[theta5]*Sin[Pi/2 + theta2 + theta3]*Sin[theta6] +
  Cos[Pi/2 + theta2 + theta3]*Sin[theta5]*Sin[theta6]},
 {Cos[Pi/2 + theta2 + theta3]*Cos[theta5] -
  Cos[theta4]*Sin[Pi/2 + theta2 + theta3]*Sin[theta5]}},

{{d6*Cos[theta4]*Cos[theta6] + d4*Cos[theta4]*Cos[theta5]*Cos[theta6] +
  a2*Cos[theta4]*Cos[theta5]*Cos[theta6]*Sin[Pi/2 + theta3] -
  a3*Cos[theta6]*Sin[theta5] +
  a2*Cos[Pi/2 + theta3]*Cos[theta6]*Sin[theta5] -
  d4*Sin[theta4]*Sin[theta6] - d6*Cos[theta5]*Sin[theta4]*Sin[theta6] -
  a2*Sin[Pi/2 + theta3]*Sin[theta4]*Sin[theta6]},
 {-(d4*Cos[theta6]*Sin[theta4]) - d6*Cos[theta5]*Cos[theta6]*Sin[theta4] -
  a2*Cos[theta6]*Sin[Pi/2 + theta3]*Sin[theta4] -
  d6*Cos[theta4]*Sin[theta6] - d4*Cos[theta4]*Cos[theta5]*Sin[theta6] -
  a2*Cos[theta4]*Cos[theta5]*Sin[Pi/2 + theta3]*Sin[theta6] +
  a3*Sin[theta5]*Sin[theta6] -
  a2*Cos[Pi/2 + theta3]*Sin[theta5]*Sin[theta6]},
 {a3*Cos[theta5] - a2*Cos[Pi/2 + theta3]*Cos[theta5] +
  d4*Cos[theta4]*Sin[theta5] +
  a2*Cos[theta4]*Sin[Pi/2 + theta3]*Sin[theta5]},
 {Cos[theta5]*Cos[theta6]*Sin[theta4] + Cos[theta4]*Sin[theta6]},
 {Cos[theta4]*Cos[theta6] - Cos[theta5]*Sin[theta4]*Sin[theta6]},
 {Sin[theta4]*Sin[theta5]}},

{{d6*Cos[theta4]*Cos[theta6] + d4*Cos[theta4]*Cos[theta5]*Cos[theta6] -
  a3*Cos[theta6]*Sin[theta5] - d4*Sin[theta4]*Sin[theta6] -
  d6*Cos[theta5]*Sin[theta4]*Sin[theta6]},
```

```
 {-(d4*Cos[theta6]*Sin[theta4]) - d6*Cos[theta5]*Cos[theta6]*Sin[theta4] -
   d6*Cos[theta4]*Sin[theta6] - d4*Cos[theta4]*Cos[theta5]*Sin[theta6] +
   a3*Sin[theta5]*Sin[theta6]},
 {a3*Cos[theta5] + d4*Cos[theta4]*Sin[theta5]},
 {Cos[theta5]*Cos[theta6]*Sin[theta4] + Cos[theta4]*Sin[theta6]},
 {Cos[theta4]*Cos[theta6] - Cos[theta5]*Sin[theta4]*Sin[theta6]},
 {Sin[theta4]*Sin[theta5]}},

{{d6*Sin[theta5]*Sin[theta6]}, {d6*Cos[theta6]*Sin[theta5]}, {0},
 {-(Cos[theta6]*Sin[theta5])}, {Sin[theta5]*Sin[theta6]}, {Cos[theta5]}},

{{d6*Cos[theta6]}, {-(d6*Sin[theta6])}, {0}, {Sin[theta6]}, {Cos[theta6]},
 {0}},

{{0}, {0}, {0}, {0}, {0}, {1}}}
```