SIMULAZIONE ROBOT 5 DOF

(Prof. Fischetti Pietro)

Simulazione in Python del Robot presentato in [1].

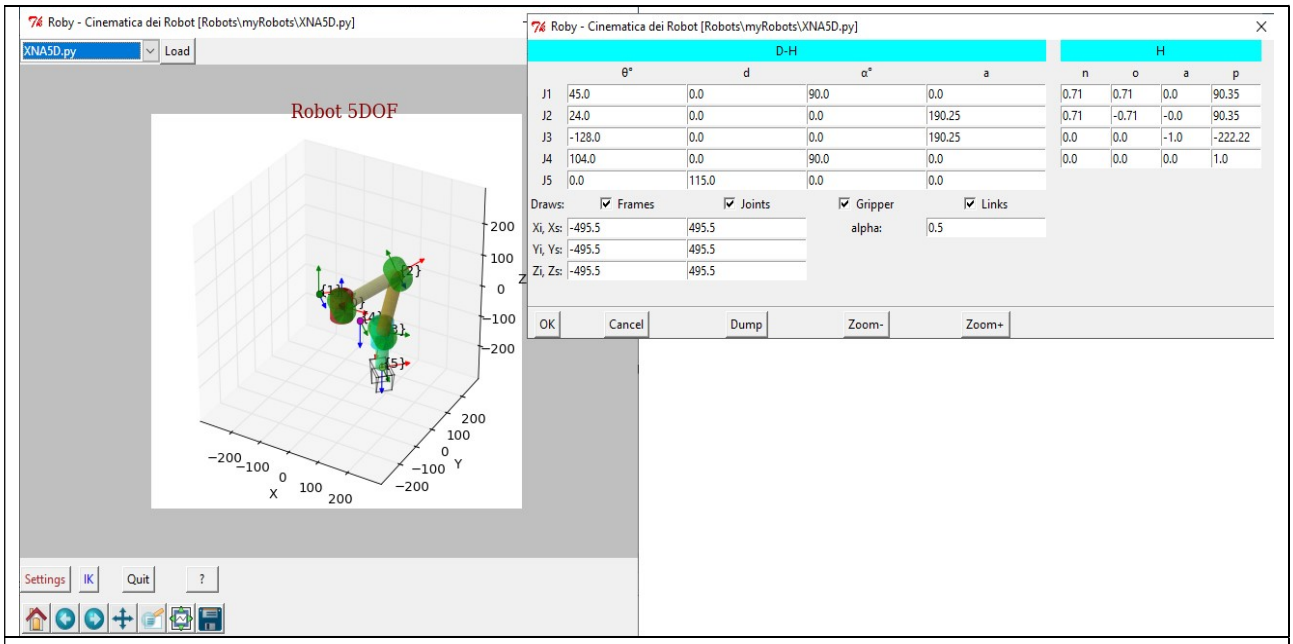
File XNA5D.py

```
import Robots
import sys
import math
sys.path.append('..')

class XNA5D(Robots.Robot):
   """
   Robot 5DOF
   """
   DSpace=3
   def __init__(self):
      Robots.Robot.__init__(self)

      ###Parametri DH per il Robot
      a2=190.25
      a3=190.25
      d5=115
      if 0:
         th1=  0;    d1=0*255;  k1= 90; a1=0
         th2=  0;    d2= 0;     k2=  0; a2=a2
         th3=  0;   d3= 0;     k3=  0; a3=a3
         th4=  0;    d4= 0;     k4= 90; a4=0
         th5=  0;    d5=d5;    k5=  0; a5=0
      if 1:
         th1=  45;    d1=0;  k1= 90; a1=0
         th2=  24;    d2= 0;     k2=  0; a2=a2
         th3=  -128;   d3= 0;     k3=  0; a3=a3
         th4=  104;    d4= 0;     k4= 90; a4=0
         th5=   0;    d5=d5;    k5=  0; a5=0
      if 0:
         th1=  64.65;    d1=0;  k1= 90; a1=0
         th2=  25.09;    d2= 0;     k2=  0; a2=a2
         th3=  -103.5;   d3= 0;     k3=  0; a3=a3
         th4=  78.4;    d4= 0;     k4= 90; a4=0
         th5=   0;    d5=d5;    k5=  0; a5=0
      if 0:
         th1=  45;    d1=0;  k1= 90; a1=0
         th2=  19.12;    d2= 0;     k2=  0; a2=a2
         th3=  -81.12;   d3= 0;     k3=  0; a3=a3
         th4=  62.01;    d4= 0;     k4= 90; a4=0
         th5=   0;    d5=d5;    k5=  0; a5=0
      if 0:
         th1=  25.35;    d1=0*255;  k1= 90; a1=0
         th2=  25.1;    d2= 0;     k2=  0; a2=a2
         th3=  -103.5;   d3= 0;     k3=  0; a3=a3
         th4=  78.41;    d4= 0;     k4= 90; a4=0
```

```
        th5=  0;    d5=d5;   k5=  0; a5=0

    ###Definisco la Matrice DH
    self.DH=[[th1,d1,k1,a1],
          [th2,d2,k2,a2],
          [th3,d3,k3,a3],
          [th4,d4,k4,a4],
          [th5,d5,k5,a5]]


    ##self.H0=self.objMatrix.MatrixCreateH([[0,1,0],[1,0,0],[0,0,1]],[[0,0,255]])
    #self.HEE=self.objMatrix.MatrixCreateH(self.objMatrix.MatrixIdentity(3),[[0,0,LEE*4]])

    H,AA = self.calcDH()

  def DrawJoints(self):
    d=dict()
    d[0]=self.objGD.RJoint.ROLL
    d[1]=self.objGD.RJoint.PITCH
    d[2]=self.objGD.RJoint.PITCH
    d[3]=self.objGD.RJoint.PITCH
    d[4]=self.objGD.RJoint.YAW

    super(type(self),self).DrawJoints(dJointsType=d)


  #Cinematica Inversa
  def updateDH(self,rik):
    self.DH[0][0]=rik[0]#th1a
    self.DH[1][0]=rik[1]#th1b
    self.DH[2][0]=rik[2]#th1b
    self.DH[3][0]=rik[3]
    self.DH[4][0]=rik[4]


  def sol(self,H,p1,m2,m234,m3):
    nx=H[0][0]
    ny=H[1][0]
    nz=H[2][0]
    ox=H[0][1]
    oy=H[1][1]
    oz=H[2][1]
    ax=H[0][2]
    ay=H[1][2]
    az=H[2][2]
    px=H[0][3]
    py=H[1][3]
    pz=H[2][3]
    a1=self.DH[0][self.objMatrix.DH_aIndex]
    a2=self.DH[1][self.objMatrix.DH_aIndex]
    a3=self.DH[2][self.objMatrix.DH_aIndex]
    a4=self.DH[3][self.objMatrix.DH_aIndex]
    a5=self.DH[4][self.objMatrix.DH_aIndex]
```

```
    d5=self.DH[4][self.objMatrix.DH_dIndex]

    q1=math.atan2(py,px)+p1
    S1=math.sin(q1)
    C1=math.cos(q1)
    S5=nx*S1-ny*C1
    C5=ox*S1-oy*C1
    q5=math.atan2(S5,C5)

    C234=(nx*C1)/C5+(ny*S1)/C5
    r1=round(1-C234**2,2)
    S234=m234*math.sqrt(r1)#1-C234**2)
    q234=math.atan2(S234,C234)
    #print 'm234',nx,ny,C1,S1,C5,C234,m234,S234,q234

    d4=d5
    C3=((px*C1+py*S1-d4*S234)**2+(pz+d4*C234)**2-a2**2-a3**2)/(2*a2*a3)

    S3=m3*math.sqrt(round(1-C3**2,2))
    q3=math.atan2(S3,C3)

    C2=((px*C1+py*S1-d4*S234)*(a3*C3+a2)+a3*S3*(pz+d4*C234))/((a3*C3+a2)**2+a3**2*S3**2)
    S2=m2*math.sqrt(round(1-C2**2,2))
    q2=math.atan2(S2,C2)
    q4=q234-q2-q3

    print "q1:{:8.2f}".format(round(math.degrees(q1),2)),
    print "q2:{:8.2f}".format(round(math.degrees(q2),2)),
    print "q3:{:8.2f}".format(round(math.degrees(q3),2)),
    print "q4:{:8.2f}".format(round(math.degrees(q4),2)),
    print "q5:{:8.2f}".format(round(math.degrees(q5),2))
  def IK(self,H):
    self.sol(H,0,1,1,1)
    self.sol(H,0,1,1,-1)
    self.sol(H,0,1,-1,1)
    self.sol(H,0,1,-1,-1)
    self.sol(H,math.radians(180),1,1,1)
    self.sol(H,math.radians(180),1,1,-1)
    self.sol(H,math.radians(180),1,-1,1)
    self.sol(H,math.radians(180),1,-1,-1)
    pass
```

Visualizzazione Grafica e Parametri D-H



Cinematica Diretta – Matrici di Trasformazione

Robot 5DOF

*** TRANSFORM MATRIX ***

H0:

[1.00 0.00 0.00 0.00
 0.00 1.00 0.00 0.00
 0.00 0.00 1.00 0.00
 0.00 0.00 0.00 1.00]

A1:

[0.71 -0.00 0.71 0.00
 0.71 0.00 -0.71 0.00
 0.00 1.00 0.00 0.00
 0.00 0.00 0.00 1.00]

A2:

[0.91 -0.41 0.00 173.80
 0.41 0.91 -0.00 77.38
 0.00 0.00 1.00   0.00
 0.00 0.00 0.00   1.00]

A3:

[-0.62 0.79 -0.00 -117.13
 -0.79 -0.62 0.00 -149.92
  0.00 0.00 1.00   0.00
  0.00 0.00 0.00   1.00]

A4:

[-0.24 -0.00 0.97 -0.00
  0.97 -0.00 0.24 0.00
  0.00 1.00 0.00 0.00
  0.00 0.00 0.00 1.00]

A5:

[1.00 0.00 0.00   0.00
 0.00 1.00 0.00   0.00
 0.00 0.00 1.00 115.00
 0.00 0.00 0.00   1.00]

H:

```
[0.71  0.71  0.00   90.35
 0.71 -0.71 -0.00   90.35
 0.00  0.00 -1.00 -222.22
 0.00  0.00  0.00    1.00 ]

*** SYMBOLIC TRASFORM MATRIX ***
A1:
[   C1  0.00    S1  0.00
    S1  0.00   -C1  0.00
  0.00  1.00  0.00  0.00
  0.00  0.00  0.00  1.00 ]
A2:
[   C2   -S2  0.00 a2*C2
    S2    C2  0.00 a2*S2
  0.00  0.00  1.00  0.00
  0.00  0.00  0.00  1.00 ]
A3:
[   C3   -S3  0.00 a3*C3
    S3    C3  0.00 a3*S3
  0.00  0.00  1.00  0.00
  0.00  0.00  0.00  1.00 ]
A4:
[   C4  0.00    S4  0.00
    S4  0.00   -C4  0.00
  0.00  1.00  0.00  0.00
  0.00  0.00  0.00  1.00 ]
A5:
[   C5   -S5  0.00  0.00
    S5    C5  0.00  0.00
  0.00  0.00  1.00    d5
  0.00  0.00  0.00  1.00 ]

H:
[nx ox az pz
 ny oy ay py
 nz oz az pz
 0  0  0  1  ]
nx=((-S2*S3*C1 + C1*C2*C3)*C4 + (-S2*C1*C3 - S3*C1*C2)*S4)*C5 + S1*S5
ny=((-S1*S2*S3 + S1*C2*C3)*C4 + (-S1*S2*C3 - S1*S3*C2)*S4)*C5 - S5*C1
nz=((-S2*S3 + C2*C3)*S4 + (S2*C3 + S3*C2)*C4)*C5
ox=-((-S2*S3*C1 + C1*C2*C3)*C4 + (-S2*C1*C3 - S3*C1*C2)*S4)*S5 + S1*C5
oy=-((-S1*S2*S3 + S1*C2*C3)*C4 + (-S1*S2*C3 - S1*S3*C2)*S4)*S5 - C1*C5
oz=-((-S2*S3 + C2*C3)*S4 + (S2*C3 + S3*C2)*C4)*S5
ax=(-S2*S3*C1 + C1*C2*C3)*S4 - (-S2*C1*C3 - S3*C1*C2)*C4
ay=(-S1*S2*S3 + S1*C2*C3)*S4 - (-S1*S2*C3 - S1*S3*C2)*C4
az=-(-S2*S3 + C2*C3)*C4 + (S2*C3 + S3*C2)*S4
px=a2*C1*C2 - a3*S2*S3*C1 + a3*C1*C2*C3 + d5*((-S2*S3*C1 + C1*C2*C3)*S4 - (-S2*C1*C3 -
S3*C1*C2)*C4)
py=a2*S1*C2 - a3*S1*S2*S3 + a3*S1*C2*C3 + d5*((-S1*S2*S3 + S1*C2*C3)*S4 - (-S1*S2*C3 -
S1*S3*C2)*C4)
pz=a2*S2 + a3*S2*C3 + a3*S3*C2 + d5*(-(-S2*S3 + C2*C3)*C4 + (S2*C3 + S3*C2)*S4)
```

Cinematica Inversa

```
[ 0.71    0.71    0.00     90.35
  0.71   -0.71   -0.00     90.35
  0.00    0.00   -1.00   -222.22
  0.00    0.00    0.00      1.00   ]
q1:   45.00 q2:  104.00 q3:  128.02 q4: -232.03 q5:   -0.00
q1:   45.00 q2:   23.63 q3: -128.02 q4:  104.39 q5:   -0.00
q1:   45.00 q2:  104.00 q3:  128.02 q4: -232.03 q5:   -0.00
q1:   45.00 q2:   23.63 q3: -128.02 q4:  104.39 q5:   -0.00
q1:  225.00 q2:  156.37 q3:  128.02 q4: -284.39 q5:  180.00
q1:  225.00 q2:   76.00 q3: -128.02 q4:   52.03 q5:  180.00
q1:  225.00 q2:  156.37 q3:  128.02 q4: -284.39 q5:  180.00
q1:  225.00 q2:   76.00 q3: -128.02 q4:   52.03 q5:  180.00
```

Riferimenti.

[1] Ghafil, Mohammed, Hadi “A Virtual Reality Environment for 5-DOF Robot Manipulator based on XNA Framework” International Journal of Computer Applications (0975 – 8887) Volume 113 – No. 3, March 2015