Contents

Robot and targets Poses

```
sawyer = SE3;
sawyer.t = [0 0 0.93];
target1 = SE3;
target1.t = [0.75 -0.35 1.014]
target2 = SE3.rpy([0 0 -0.785]);
target2.t = [1 0 1.014]
target3 = SE3.rpy([0 0 0.586931]);
target3.t = [0.85 0.25 1.014]
```

```
target1 =
         1         0         0      0.75
         0         1         0     -0.35
         0         0         1     1.014
         0         0         0         1


target2 =
    0.7074    0.7068         0         1
   -0.7068    0.7074         0         0
         0         0         1     1.014
         0         0         0         1


target3 =
    0.8326   -0.5538         0      0.85
    0.5538    0.8326         0      0.25
         0         0         1     1.014
         0         0         0         1
```

Part 1: A. Poses of the three targets with the repect to the robot base

frame

```
sawyerTtarget1 = inv(sawyer)*target1
sawyerTtarget2 = inv(sawyer)*target2
sawyerTtarget3 = inv(sawyer)*target3
```

```
sawyerTtarget1 =
         1         0         0      0.75
         0         1         0     -0.35
```

```
         0         0         1     0.084
         0         0         0         1


sawyerTtarget2 =
    0.7074    0.7068         0         1
   -0.7068    0.7074         0         0
         0         0         1     0.084
         0         0         0         1


sawyerTtarget3 =
    0.8326   -0.5538         0      0.85
    0.5538    0.8326         0      0.25
         0         0         1     0.084
         0         0         0         1
```

Part 1: B. The joint variables to grasp the object at each location, ie.

inverse kinematics

```
L(1) = Link('alpha', -pi/2, 'a', 0.081, 'd', 0);
L(2) = Link('alpha', pi/2, 'a', 0, 'd', 0.191);
L(3) = Link('alpha', -pi/2, 'a', 0, 'd', 0.399);
L(4) = Link('alpha', pi/2, 'a', 0, 'd', -0.1683);
L(5) = Link('alpha', -pi/2, 'a', 0, 'd', 0.3965);
L(6) = Link('alpha', pi/2, 'a', 0, 'd', 0.136);
L(7) = Link('alpha', 0, 'a', 0, 'd', 0.1785);
R1 = SerialLink(L, 'name', "Sawyer");

pose1 = sawyerTtarget1*SE3.Ry(pi/2);
pose2 = sawyerTtarget2*SE3.Rx(-pi/2);
pose3 = sawyerTtarget3*SE3.Ry(pi/2);
pose4 = pose1; % discard position
pose4.t = [0.65 0.4 0.084];

q1dwn = R1.ikine(pose1);
% R1.teach(q1dwn);
q1up = q1dwn -[0 pi/6 0 0 0 0 0];
% R1.teach(q1up)

q2dwn = R1.ikine(pose2);
% R1.teach(q2dwn)
q2up = q2dwn -[0 pi/6 0 0 0 0 0];
% R1.teach(q2up)

q3dwn = R1.ikine(pose3);
% R1.teach(q3dwn)
q3up = q3dwn -[0 pi/6 0 0 0 0 0];
% R1.teach(q3up)

q4dwn = R1.ikine(pose4);
% R1.teach(q4dwn);
q4up = q4dwn -[0 pi/6 0 0 0 0 0];
% R1.teach(q4up)
```

- in radians

```
in_radians = array2table([q1dwn; q2dwn; q3dwn;q4dwn], ...
    'VariableNames', {'q1','q2','q3','q4','q5','q6','q7'}, ...
    'RowNames', {'pos1dwn','pos2dwn','pos3dwn','pos4dwn'} )
```

```
in_radians =

  4×7 table

                q1         q2         q3         q4         q5         q6         q7
              _______    _______    _______    _______    _______    _______    _______

    pos1dwn    -1.5598     1.0334     1.4951     1.7613    -0.33971   -0.21166    -0.70001
    pos2dwn    -0.18375    1.4313    -0.46561   -0.085423   -1.2564    -0.94749     0.26679
    pos3dwn    -0.62441    0.78528    1.1561     1.4874    -0.72682   -0.2269     0.00049313
    pos4dwn     1.013      1.3239    -1.3704     1.3283     0.14321   -0.29985     1.1199
```

- in degrees

```
in_degrees = array2table(rad2deg([q1dwn; q2dwn; q3dwn;q4dwn]), ...
    'VariableNames', {'q1','q2','q3','q4','q5','q6','q7'}, ...
    'RowNames', {'pos1dwn','pos2dwn','pos3dwn','pos4dwn'} )
```

```
in_degrees =

  4×7 table

                q1         q2         q3         q4         q5         q6         q7
              _______    _______    _______    _______    _______    _______    _______

    pos1dwn    -89.372    59.208     85.664     100.91    -19.464    -12.127    -40.108
    pos2dwn    -10.528    82.006    -26.677    -4.8944    -71.989    -54.287     15.286
    pos3dwn    -35.776    44.993     66.24      85.221    -41.644        -13    0.028254
    pos4dwn     58.043    75.854    -78.516     76.105     8.2056    -17.18      64.165
```

Part 1: C.The Jacobian matrix at each location using the joint variables

```
disp("Jacobian for target1")
R1.jacob0(q1dwn)
disp("Jacobian for target2")
R1.jacob0(q2dwn)
disp("Jacobian for target3")
R1.jacob0(q3dwn)
disp("Jacobian for target4")
R1.jacob0(q4dwn)
```

```
Jacobian for target1

ans =

    0.3500    0.0009    0.0666   -0.1228    0.0286    0.0000         0
```

```
    0.7500   -0.0840    0.2849    0.4833   -0.1143    0.1150         0
   -0.0000   -0.2772    0.4769   -0.2784    0.0775   -0.1365         0
   -0.0000    0.9999    0.0094    0.0700    0.9777    0.0000    1.0000
    0.0000    0.0110   -0.8590    0.5113    0.1353    0.7648   -0.0000
    1.0000    0.0000    0.5119    0.8566   -0.1607    0.6442   -0.0000

Jacobian for target2

ans =

   -0.0000    0.0826    0.0089   -0.0201   -0.0031    0.1218         0
    1.0000   -0.0153    0.0414   -0.2274    0.1592   -0.1217         0
    0.0000   -0.9022   -0.0082   -0.4536    0.1189    0.0471         0
   -0.0000    0.1827    0.9736    0.2247    0.9666    0.1865    0.7068
    0.0000    0.9832   -0.1809    0.8671   -0.1407   -0.1863    0.7074
    1.0000    0.0000    0.1391   -0.4446    0.2141   -0.9646    0.0000

Jacobian for target3

ans =

   -0.2500    0.0681   -0.1354   -0.3386    0.0477    0.0000         0
    0.8500   -0.0491    0.4275    0.2298   -0.0164   -0.0001         0
    0.0000   -0.4625    0.3597   -0.4020    0.1325   -0.1785         0
    0.0000    0.5846    0.5736   -0.2896    0.8114   -0.5538    0.8326
    0.0000    0.8113   -0.4133    0.7053    0.5395    0.8326    0.5538
    1.0000    0.0000    0.7072    0.6471   -0.2250   -0.0005    0.0000

Jacobian for target4

ans =

   -0.4000    0.0445    0.0129   -0.0974    0.0402    0.0000         0
    0.6500    0.0713    0.1449   -0.5369    0.0940   -0.1607         0
    0.0000   -0.6024   -0.5147   -0.1700    0.1041   -0.0778         0
   -0.0000   -0.8484    0.5132   -0.0422    0.9554    0.0000    1.0000
   -0.0000    0.5293    0.8227    0.3086   -0.2659    0.4358   -0.0000
    1.0000    0.0000    0.2444   -0.9503   -0.1287   -0.9001    0.0000
```

Published with MATLAB® R2021b