ME572: Phase #2

Due on April 20, 2012

RR robot from homework #7. The analysis is the same as for HW 7. The peak joint velocity is in joint 1 at position 12.

Symbolic jacobian matrix for RR.

$$
\begin{aligned}
J(1,1) &= \left[-20 \cdot \sin(\theta_1 + \theta_2) - 20 \cdot \sin(\theta_1)\right] \\
J(1,2) &= \left[-20 \cdot \sin(\theta_1 + \theta_2)\right] \\
J(2,1) &= \left[20 \cdot \cos(\theta_1 + \theta_2) + 20 \cdot \cos(\theta_1)\right] \\
J(2,2) &= \left[20 \cdot \cos(\theta_1 + \theta_2)\right] \\
J(3,1) &= \left[0\right] \\
J(3,2) &= \left[0\right] \\
J(4,1) &= \left[0\right] \\
J(4,2) &= \left[0\right] \\
J(5,1) &= \left[0\right] \\
J(5,2) &= \left[0\right] \\
J(6,1) &= \left[1\right] \\
J(6,2) &= \left[1\right]
\end{aligned}
$$

Joint position and velocities for trajectory1.dat.

```
21 % Number of positions
%  t(s)    Theta1   Theta2   Omega1   Omega2
   0.000   81.469  197.254    0.000    0.000
   0.025   81.672  196.852    0.284   -0.562
   0.050   82.282  195.646    0.568   -1.122
   0.075   83.097  194.041    0.569   -1.120
   0.100   83.913  192.439    0.571   -1.118
   0.125   84.732  190.839    0.574   -1.116
   0.150   85.557  189.241    0.579   -1.115
   0.175   86.392  187.645    0.588   -1.114
   0.200   87.246  186.051    0.606   -1.113
   0.225   88.140  184.458    0.648   -1.112
   0.250   89.140  182.865    0.778   -1.111
   0.275   90.652  181.274    1.680   -1.111
   0.300  264.697  180.320   17.302    1.107
   0.325  268.186  181.910   -0.056    1.111
   0.350  267.780  183.502   -0.407    1.112
   0.375  267.130  185.095   -0.486    1.112
   0.400  266.410  186.688   -0.516    1.113
   0.425  265.660  188.283   -0.530    1.114
   0.450  264.894  189.880   -0.539    1.115
   0.475  264.312  191.079   -0.272    0.558
   0.500  264.118  191.478    0.000    0.000
```

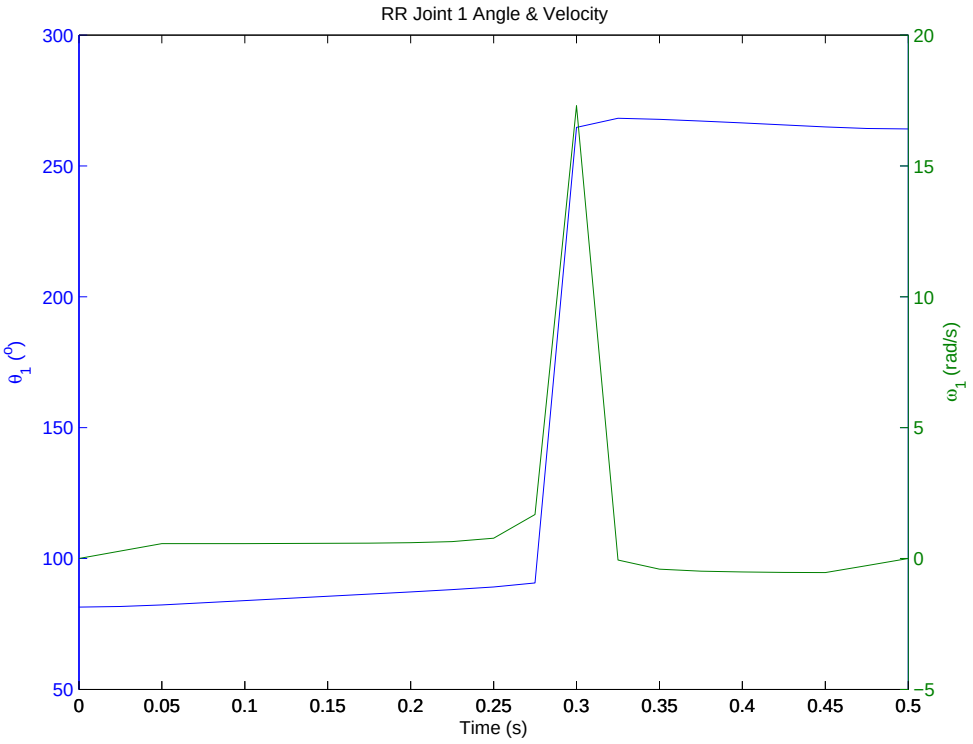

Figure 1: RR Joint 1 Angle and Velocity

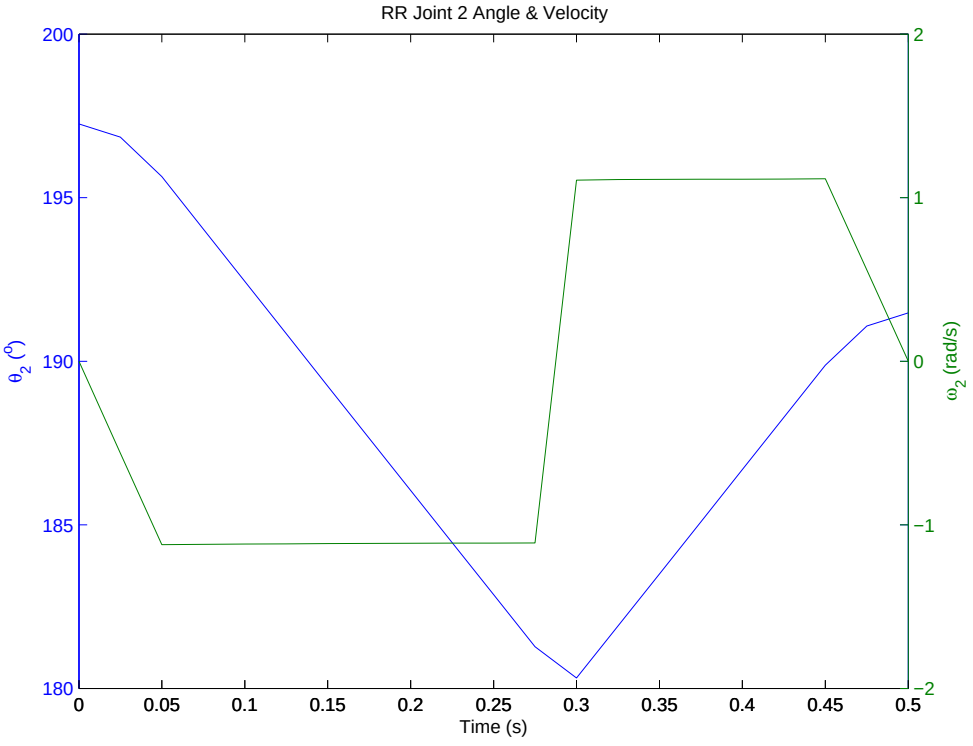

Figure 2: RR Joint 2 Angle and Velocity

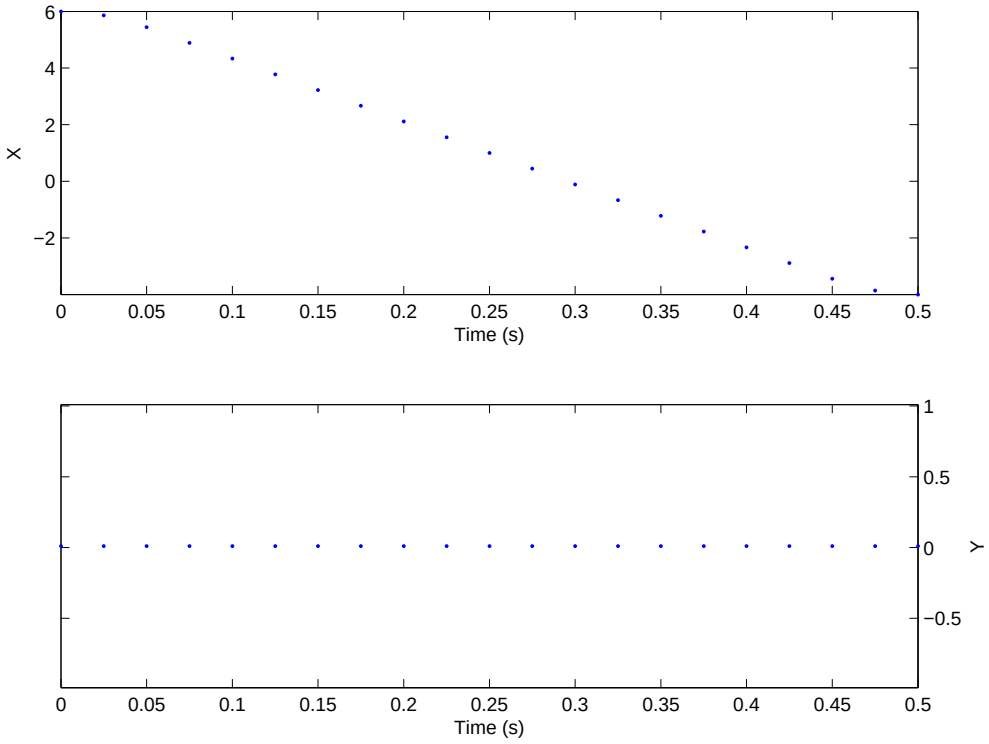

Figure 3: Position of the end effector for each of the coordinates vs time

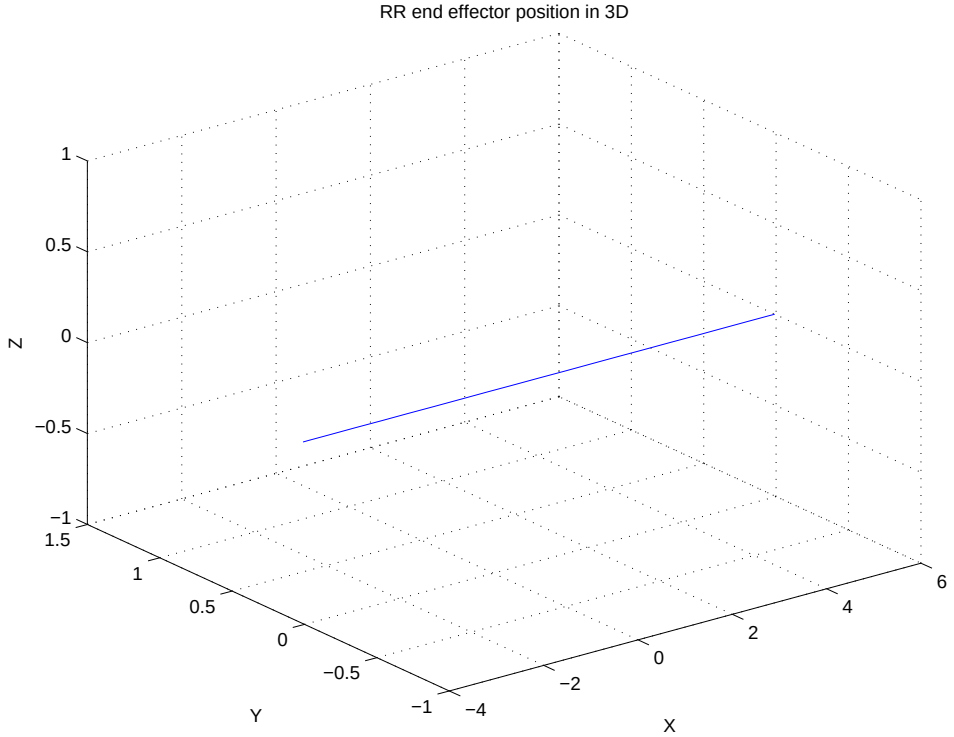

Figure 4: RR end effector position in 3D

World transforms of the robot at positions calculated for coordinate interpolated motion.

```
POSITION 1: INPUT: JOINT VARIABLES (  81.469, 197.254,)
OUTPUT:     0.152     0.988     0.000     6.000
           -0.988     0.152     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 2: INPUT: JOINT VARIABLES (  81.672, 196.852,)
OUTPUT:     0.148     0.989     0.000     5.861
           -0.989     0.148     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 3: INPUT: JOINT VARIABLES (  82.282, 195.646,)
OUTPUT:     0.138     0.990     0.000     5.444
           -0.990     0.138     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 4: INPUT: JOINT VARIABLES (  83.097, 194.041,)
OUTPUT:     0.124     0.992     0.000     4.889
           -0.992     0.124     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 5: INPUT: JOINT VARIABLES (  83.913, 192.439,)
OUTPUT:     0.111     0.994     0.000     4.333
           -0.994     0.111     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 6: INPUT: JOINT VARIABLES (  84.732, 190.839,)
OUTPUT:     0.097     0.995     0.000     3.778
           -0.995     0.097     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 7: INPUT: JOINT VARIABLES (  85.557, 189.241,)
OUTPUT:     0.084     0.996     0.000     3.222
           -0.996     0.084     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 8: INPUT: JOINT VARIABLES (  86.392, 187.645,)
OUTPUT:     0.070     0.998     0.000     2.667
           -0.998     0.070     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000
```

```
POSITION 9: INPUT: JOINT VARIABLES (  87.246, 186.051,)
OUTPUT:     0.058     0.998     0.000     2.111
           -0.998     0.058     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 10: INPUT: JOINT VARIABLES (  88.140, 184.458,)
OUTPUT:     0.045     0.999     0.000     1.556
           -0.999     0.045     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 11: INPUT: JOINT VARIABLES (  89.140, 182.865,)
OUTPUT:     0.035     0.999     0.000     1.000
           -0.999     0.035     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 12: INPUT: JOINT VARIABLES (  90.652, 181.274,)
OUTPUT:     0.034     0.999     0.000     0.444
           -0.999     0.034     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 13: INPUT: JOINT VARIABLES ( 264.697, 180.320,)
OUTPUT:     0.087    -0.996     0.000    -0.111
            0.996     0.087     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 14: INPUT: JOINT VARIABLES ( 268.186, 181.910,)
OUTPUT:    -0.002    -1.000     0.000    -0.667
            1.000    -0.002     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 15: INPUT: JOINT VARIABLES ( 267.780, 183.502,)
OUTPUT:    -0.022    -1.000     0.000    -1.222
            1.000    -0.022     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000

POSITION 16: INPUT: JOINT VARIABLES ( 267.130, 185.095,)
OUTPUT:    -0.039    -0.999     0.000    -1.778
            0.999    -0.039     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000
```

```
POSITION 17: INPUT: JOINT VARIABLES ( 266.410, 186.688,)
OUTPUT:    -0.054    -0.999     0.000    -2.333
            0.999    -0.054     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000


POSITION 18: INPUT: JOINT VARIABLES ( 265.660, 188.283,)
OUTPUT:    -0.069    -0.998     0.000    -2.889
            0.998    -0.069     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000


POSITION 19: INPUT: JOINT VARIABLES ( 264.894, 189.880,)
OUTPUT:    -0.083    -0.997     0.000    -3.444
            0.997    -0.083     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000


POSITION 20: INPUT: JOINT VARIABLES ( 264.312, 191.079,)
OUTPUT:    -0.094    -0.996     0.000    -3.861
            0.996    -0.094     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000


POSITION 21: INPUT: JOINT VARIABLES ( 264.118, 191.478,)
OUTPUT:    -0.098    -0.995     0.000    -4.000
            0.995    -0.098     0.000     0.010
            0.000     0.000     1.000     0.000
            0.000     0.000     0.000     1.000
```

Manipulability of the robot.

```
%  t(s)       ax       by
   0.000   5.926   20.022
   0.025   5.792   20.020
   0.050   5.390   20.015
   0.075   4.850   20.009
   0.100   4.307   20.006
   0.125   3.760   20.003
   0.150   3.211   20.002
   0.175   2.661   20.001
   0.200   2.108   20.000
   0.225   1.554   20.000
   0.250   1.000   20.000
   0.275   0.445   20.000
   0.300   0.112   20.000
   0.325   0.667   20.000
   0.350   1.222   20.000
```

```
0.375   1.776   20.000
0.400   2.329   20.000
0.425   2.881   20.001
0.450   3.431   20.002
0.475   3.842   20.004
0.500   3.979   20.004
```

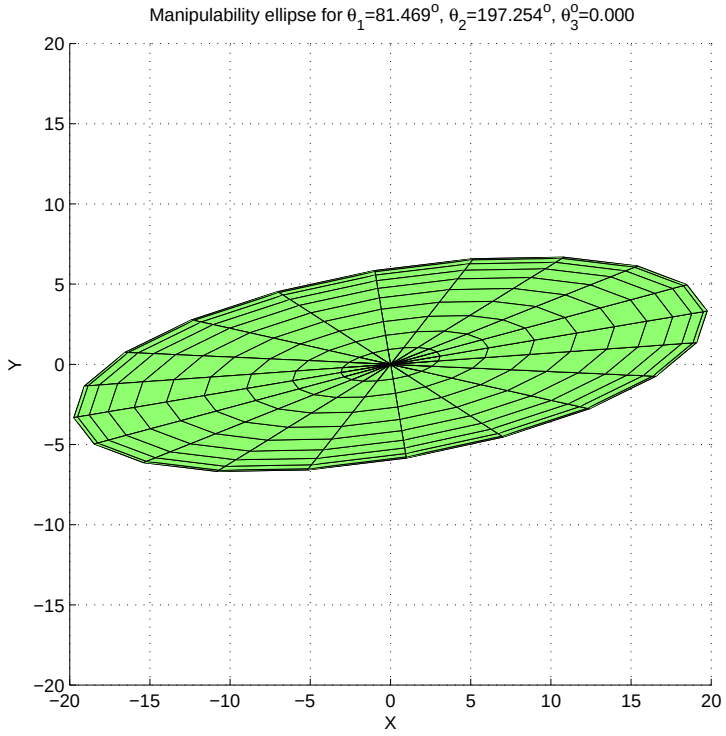

Figure 5: Manipulability of RR at position 0.

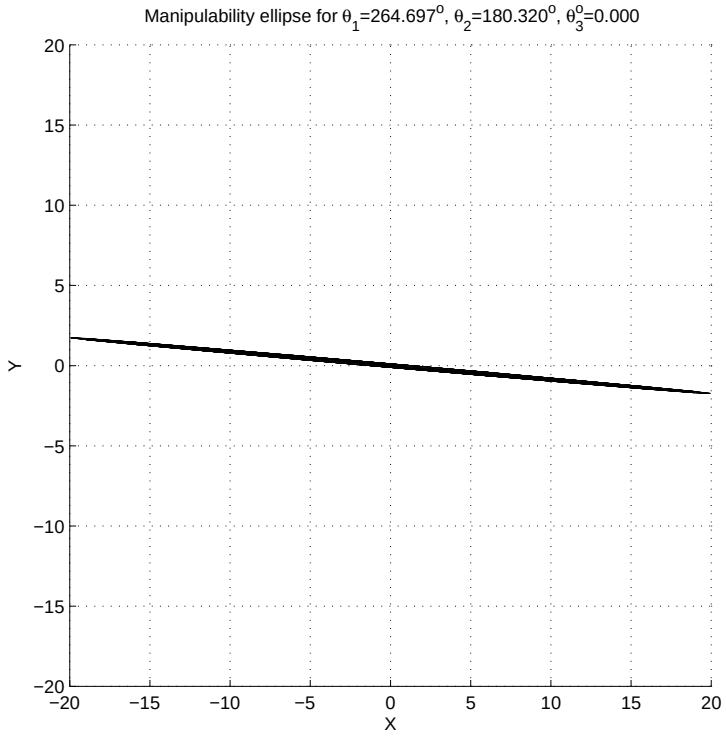

Figure 6: Manipulability of RR at position 12.

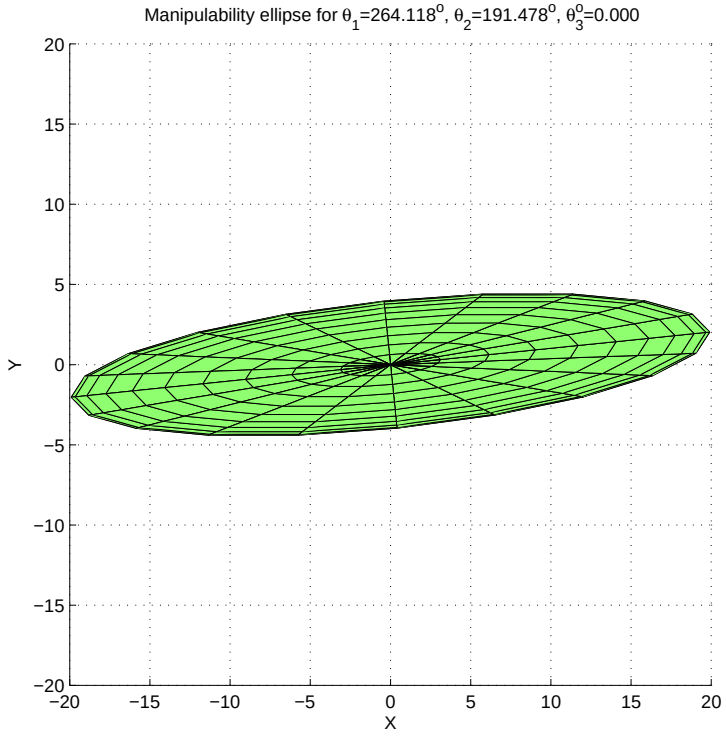

Figure 7: Manipulability of RR at position 20.

RRR robot from homework #4. The peak joint velocity is in joint 2 at position 10.
Symbolic jacobian matrix for RRR.

$$
\begin{aligned}
J(1,1) &= \left[5 \cdot \cos(\theta_1) \cdot \sin(\theta_2) - 4 \cdot \sin(\theta_1) + 4 \cdot \cos(\theta_1) \cdot \cos(\theta_2) \cdot \sin(\theta_3) + 4 \cdot \cos(\theta_1) \cdot \cos(\theta_3) \cdot \sin(\theta_2)\right] \\
J(1,2) &= \left[\sin(\theta_1) \cdot (4 \cdot \cos(\theta_2 + \theta_3) + 5 \cdot \cos(\theta_2))\right] \\
J(1,3) &= \left[4 \cdot \cos(\theta_2 + \theta_3) \cdot \sin(\theta_1)\right] \\
J(2,1) &= \left[4 \cdot \cos(\theta_1) + 5 \cdot \sin(\theta_1) \cdot \sin(\theta_2) + 4 \cdot \cos(\theta_2) \cdot \sin(\theta_1) \cdot \sin(\theta_3) + 4 \cdot \cos(\theta_3) \cdot \sin(\theta_1) \cdot \sin(\theta_2)\right] \\
J(2,2) &= \left[-\cos(\theta_1) \cdot (4 \cdot \cos(\theta_2 + \theta_3) + 5 \cdot \cos(\theta_2))\right] \\
J(2,3) &= \left[-4 \cdot \cos(\theta_2 + \theta_3) \cdot \cos(\theta_1)\right] \\
J(3,1) &= \left[0\right] \\
J(3,2) &= \left[-4 \cdot \sin(\theta_2 + \theta_3) - 5 \cdot \sin(\theta_2)\right] \\
J(3,3) &= \left[-4 \cdot \sin(\theta_2 + \theta_3)\right] \\
J(4,1) &= \left[0\right] \\
J(4,2) &= \left[\cos(\theta_1)\right] \\
J(4,3) &= \left[\cos(\theta_1)\right] \\
J(5,1) &= \left[0\right] \\
J(5,2) &= \left[\sin(\theta_1)\right] \\
J(5,3) &= \left[\sin(\theta_1)\right] \\
J(6,1) &= \left[1\right] \\
J(6,2) &= \left[0\right] \\
J(6,3) &= \left[0\right]
\end{aligned}
$$

Joint position and velocities for trajectory2.dat.

```
17 % Number of positions
%  t(s)    Theta1   Theta2   Theta3   Omega1   Omega2   Omega3
   0.000    0.000  326.976   50.858    0.000    0.000    0.000
   0.015  359.861  324.008   57.465   -0.331   -6.535   14.560
   0.030  359.406  316.558   74.095   -0.751  -10.385   23.232
   0.045  358.701  308.475   92.237   -0.895   -8.619   19.415
   0.060  357.859  301.530  107.937   -1.070   -7.626   17.298
   0.075  356.851  295.261  122.191   -1.282   -7.008   15.969
   0.090  355.645  289.414  135.512   -1.531   -6.634   15.086
   0.105  354.209  283.791  148.200   -1.814   -6.498   14.477
   0.120  352.519  278.115  160.433   -2.122   -6.853   13.991
   0.135  350.560  271.129  172.081   -2.436  -11.350   12.651
   0.150  348.336  228.896  174.008   -2.736  -89.233  -11.495
   0.165  345.866  174.785  162.643   -3.006  -37.545  -13.887
   0.180  343.182  154.756  150.488   -3.233  -13.147  -14.384
   0.195  340.323  148.097  137.893   -3.412   -3.653  -14.956
   0.210  337.331  147.121  124.708   -3.544    0.919  -15.779
   0.225  335.022  148.445  114.271   -1.808    1.539   -8.325
   0.240  334.244  149.172  110.656    0.000    0.000    0.000
```

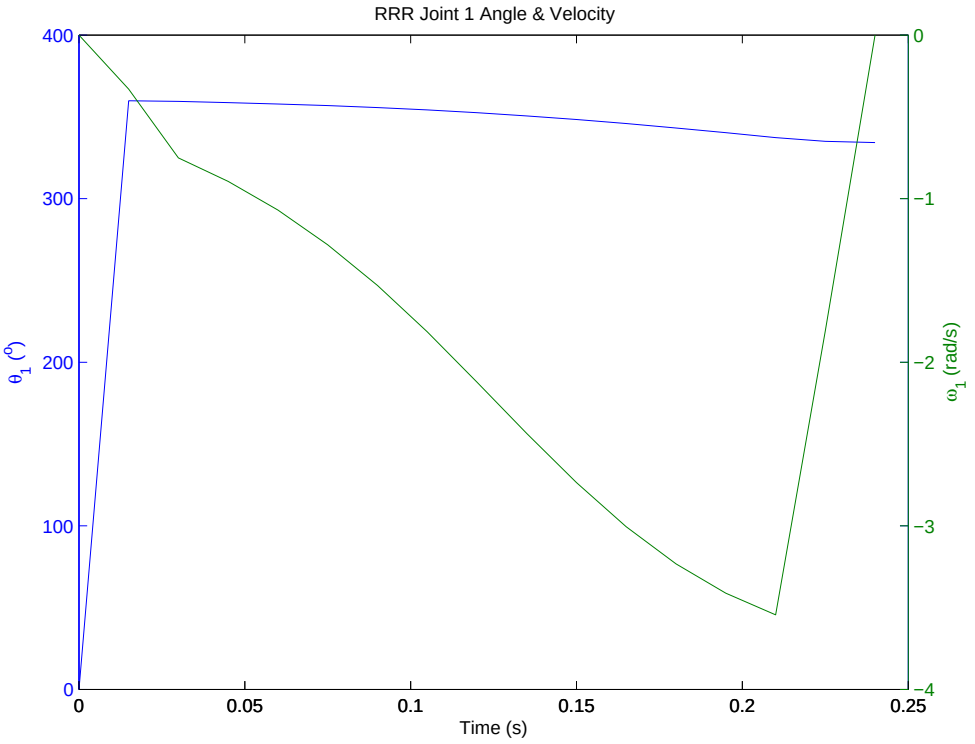

Figure 8: RRR Joint 1 Angle and Velocity

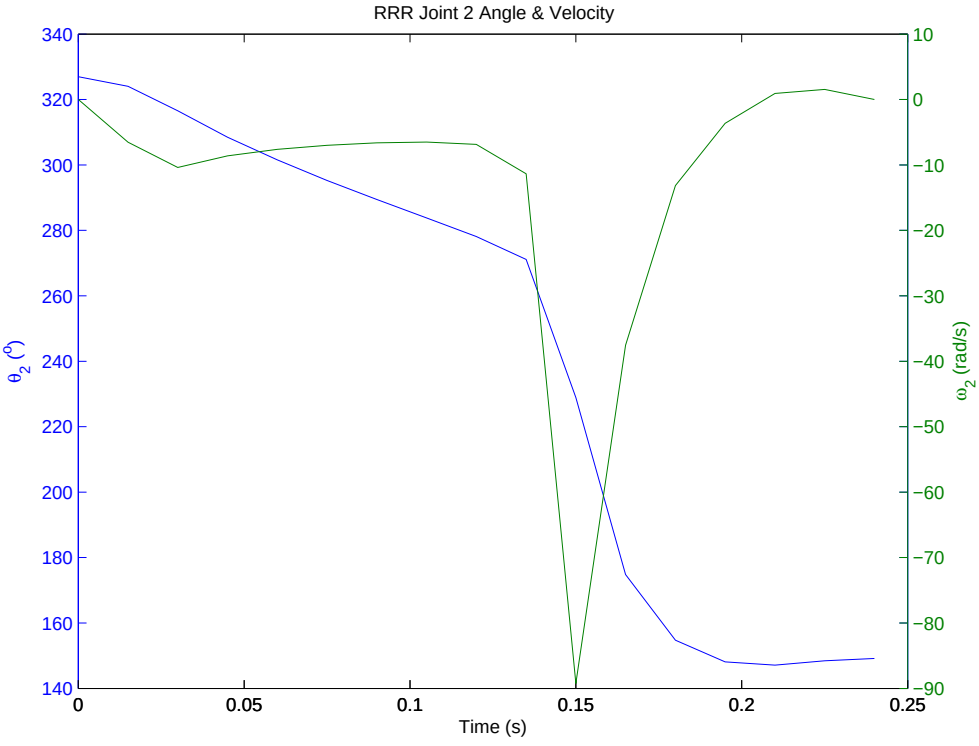

Figure 9: RRR Joint 2 Angle and Velocity

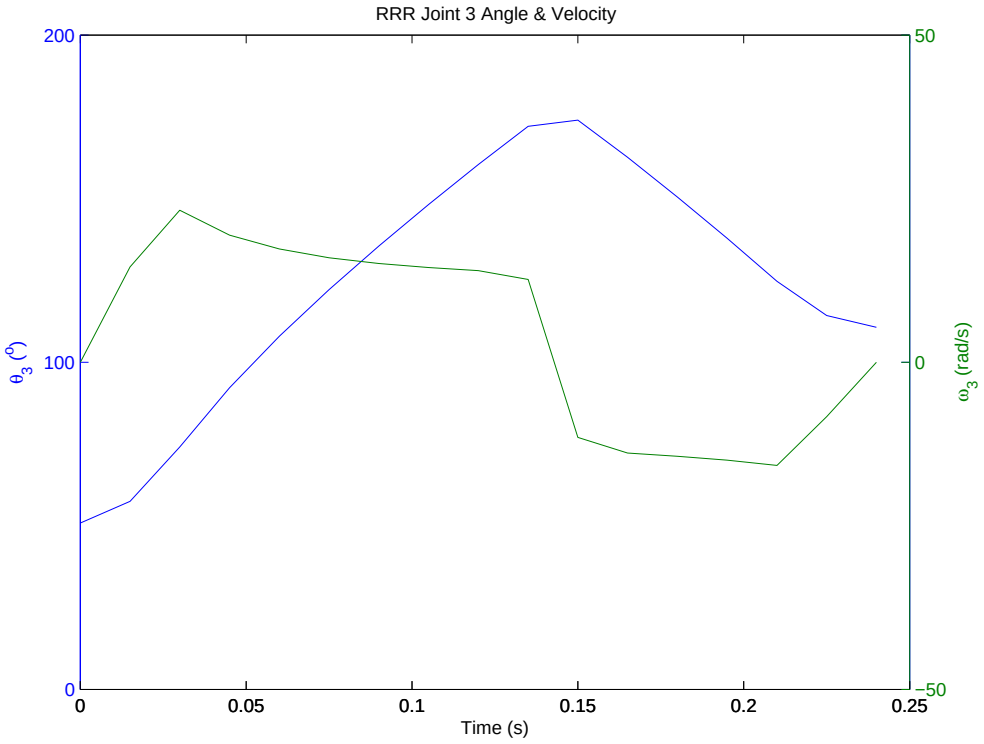

Figure 10: RRR Joint 3 Angle and Velocity

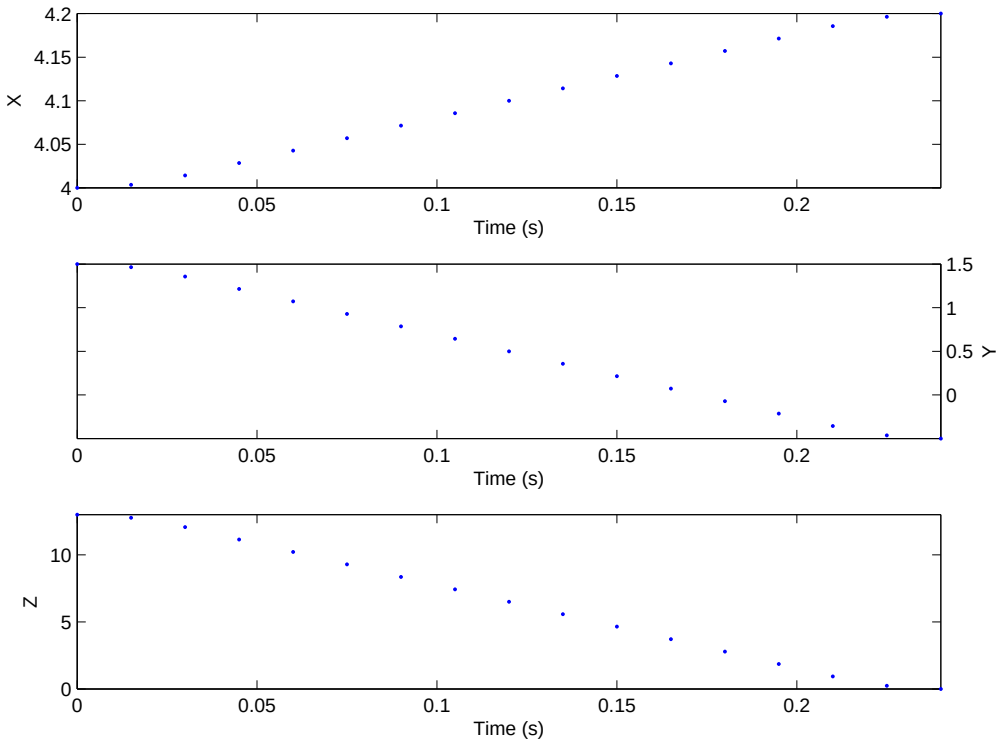

Figure 11: Position of the end effector for each of the coordinates vs time

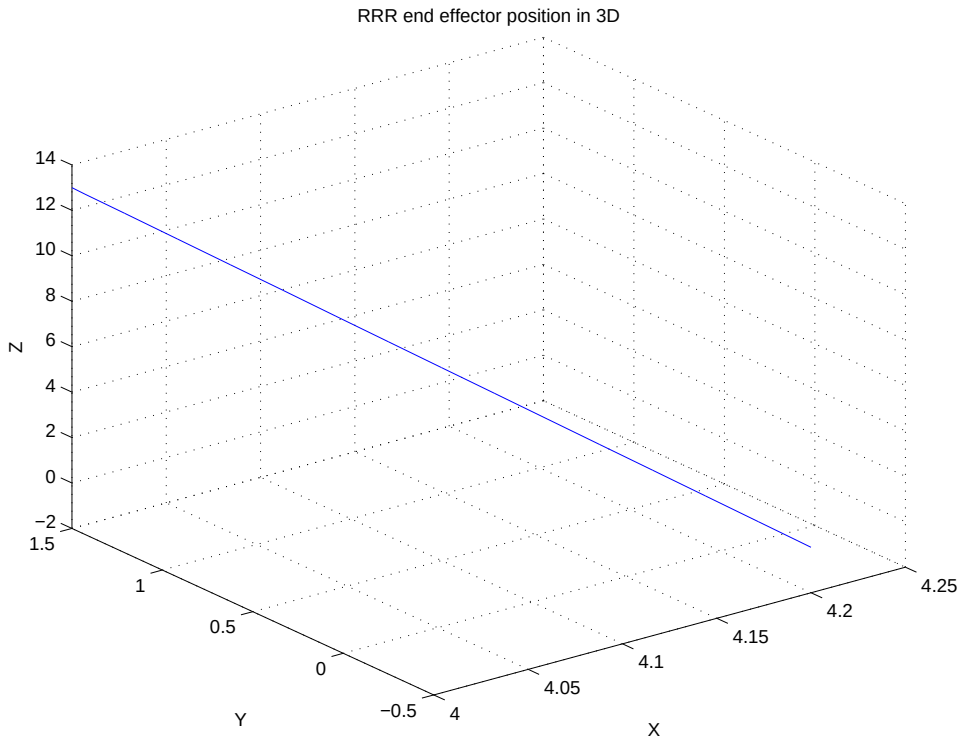

Figure 12: RRR end effector position in 3D

World transforms of the robot at positions calculated for coordinate interpolated motion.

```
POSITION 1: INPUT: JOINT VARIABLES (   0.000, 326.976,  50.858,)
OUTPUT:     1.000     0.000     0.000     4.000
            0.000     0.952    -0.306     1.500
            0.000     0.306     0.952    13.000
            0.000     0.000     0.000     1.000

POSITION 2: INPUT: JOINT VARIABLES ( 359.861, 324.008,  57.465,)
OUTPUT:     1.000     0.002    -0.001     4.004
           -0.002     0.931    -0.366     1.464
            0.000     0.366     0.931    12.768
            0.000     0.000     0.000     1.000

POSITION 3: INPUT: JOINT VARIABLES ( 359.406, 316.558,  74.095,)
OUTPUT:     1.000     0.009    -0.005     4.014
           -0.010     0.860    -0.510     1.357
            0.000     0.510     0.860    12.071
            0.000     0.000     0.000     1.000

POSITION 4: INPUT: JOINT VARIABLES ( 358.701, 308.475,  92.237,)
OUTPUT:     1.000     0.017    -0.015     4.029
           -0.023     0.758    -0.652     1.214
            0.000     0.652     0.758    11.143
            0.000     0.000     0.000     1.000

POSITION 5: INPUT: JOINT VARIABLES ( 357.859, 301.530, 107.937,)
OUTPUT:     0.999     0.024    -0.028     4.043
           -0.037     0.649    -0.759     1.071
            0.000     0.760     0.650    10.214
            0.000     0.000     0.000     1.000

POSITION 6: INPUT: JOINT VARIABLES ( 356.851, 295.261, 122.191,)
OUTPUT:     0.998     0.030    -0.046     4.057
           -0.055     0.537    -0.842     0.929
            0.000     0.843     0.538     9.286
            0.000     0.000     0.000     1.000

POSITION 7: INPUT: JOINT VARIABLES ( 355.645, 289.414, 135.512,)
OUTPUT:     0.997     0.032    -0.069     4.071
           -0.076     0.423    -0.903     0.786
            0.000     0.906     0.424     8.357
            0.000     0.000     0.000     1.000

POSITION 8: INPUT: JOINT VARIABLES ( 354.209, 283.791, 148.200,)
OUTPUT:     0.995     0.031    -0.096     4.086
           -0.101     0.308    -0.946     0.643
            0.000     0.951     0.309     7.429
            0.000     0.000     0.000     1.000
```

```
POSITION 9: INPUT: JOINT VARIABLES ( 352.519, 278.115, 160.433,)
OUTPUT:     0.991     0.026    -0.128     4.100
           -0.130     0.197    -0.972     0.500
            0.000     0.980     0.199     6.500
            0.000     0.000     0.000     1.000

POSITION 10: INPUT: JOINT VARIABLES ( 350.560, 271.129, 172.081,)
OUTPUT:     0.986     0.019    -0.163     4.114
           -0.164     0.117    -0.980     0.357
            0.000     0.993     0.118     5.571
            0.000     0.000     0.000     1.000

POSITION 11: INPUT: JOINT VARIABLES ( 348.336, 228.896, 174.008,)
OUTPUT:     0.979     0.148    -0.138     4.129
           -0.202     0.717    -0.667     0.214
            0.000     0.681     0.733     4.643
            0.000     0.000     0.000     1.000

POSITION 12: INPUT: JOINT VARIABLES ( 345.866, 174.785, 162.643,)
OUTPUT:     0.970     0.225     0.094     4.143
           -0.244     0.895     0.372     0.071
            0.000    -0.384     0.923     3.714
            0.000     0.000     0.000     1.000

POSITION 13: INPUT: JOINT VARIABLES ( 343.182, 154.756, 150.488,)
OUTPUT:     0.957     0.167     0.236     4.157
           -0.289     0.552     0.782    -0.071
            0.000    -0.817     0.577     2.786
            0.000     0.000     0.000     1.000

POSITION 14: INPUT: JOINT VARIABLES ( 340.323, 148.097, 137.893,)
OUTPUT:     0.942     0.093     0.324     4.171
           -0.337     0.259     0.905    -0.214
            0.000    -0.961     0.275     1.857
            0.000     0.000     0.000     1.000

POSITION 15: INPUT: JOINT VARIABLES ( 337.331, 147.121, 124.708,)
OUTPUT:     0.923     0.012     0.385     4.186
           -0.385     0.029     0.922    -0.357
            0.000    -0.999     0.032     0.929
            0.000     0.000     0.000     1.000

POSITION 16: INPUT: JOINT VARIABLES ( 335.022, 148.445, 114.271,)
OUTPUT:     0.906    -0.054     0.419     4.196
           -0.422    -0.115     0.899    -0.464
            0.000    -0.992    -0.127     0.232
            0.000     0.000     0.000     1.000
```

```
POSITION 17: INPUT: JOINT VARIABLES ( 334.244, 149.172, 110.656,)
OUTPUT:     0.901    -0.077     0.428     4.200
           -0.435    -0.159     0.886    -0.500
            0.000    -0.984    -0.177     0.000
            0.000     0.000     0.000     1.000
```

Manipulability of the robot.

% t(s)	ax	by	cz
0.000	1.307	1.822	9.771
0.015	1.307	1.994	9.539
0.030	1.239	2.452	8.853
0.045	1.123	2.917	7.961
0.060	1.001	3.268	7.106
0.075	0.873	3.536	6.305
0.090	0.734	3.738	5.581
0.105	0.574	3.885	4.968
0.120	0.384	3.980	4.515
0.135	0.165	3.987	4.302
0.150	0.173	2.345	5.365
0.165	0.648	1.745	5.703
0.180	0.644	3.248	5.340
0.195	0.757	4.027	5.290
0.210	0.885	4.049	5.890
0.225	0.980	3.855	6.524
0.240	1.010	3.775	6.746

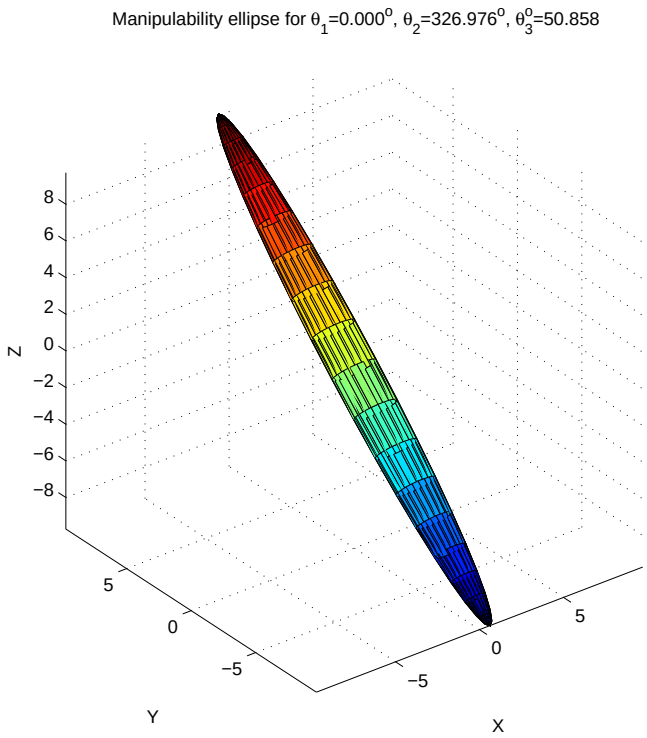

Figure 13: Manipulability of RRR at position 0.

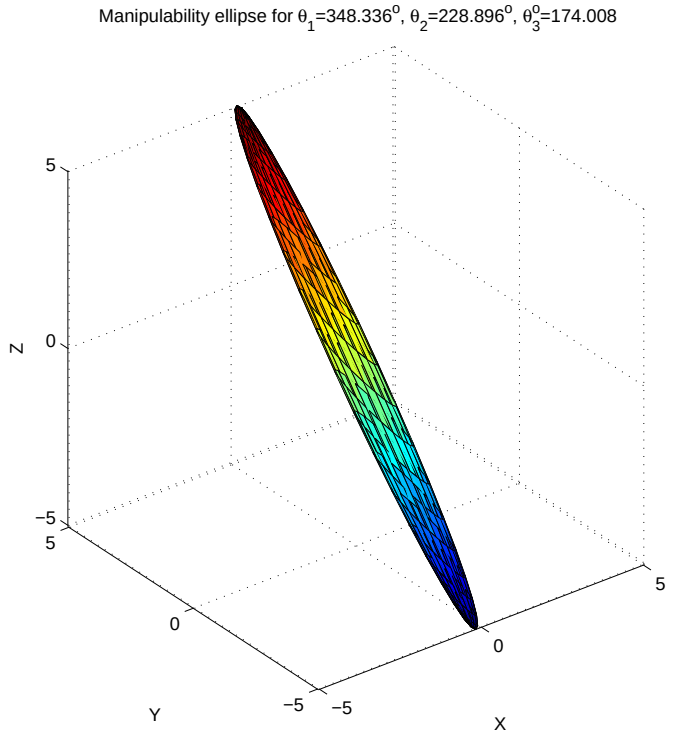

Figure 14: Manipulability of RRR at position 10.

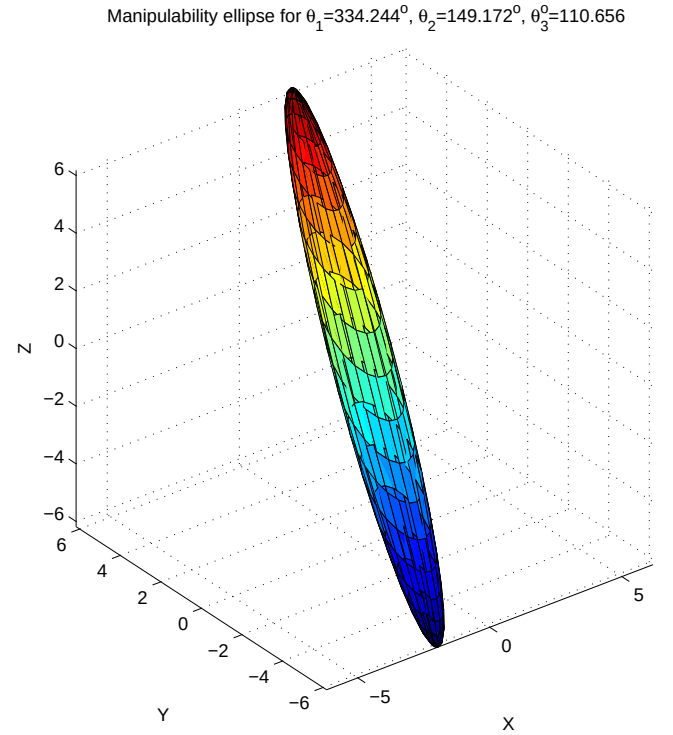

Figure 15: Manipulability of RRR at position 16.

Changing the time step to 0.0001 s gives a finer step of the X, Y and Z positions. It also shows where the actual peak angular joint rate is as it approaches the singularity.

RR robot for dT=0.0001s.

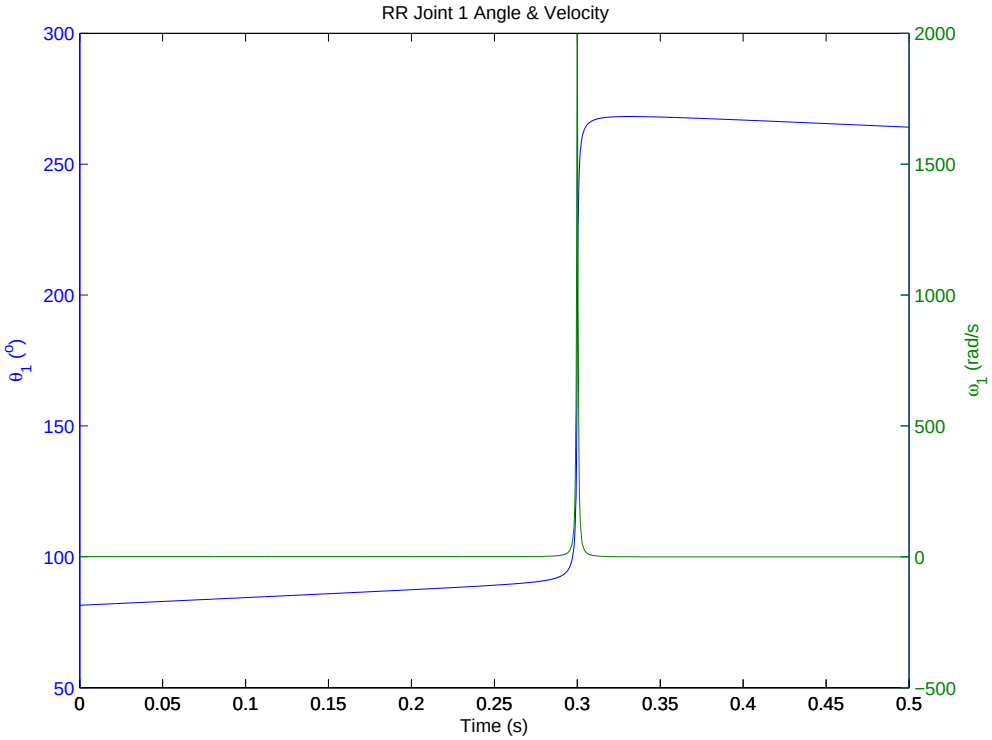

Figure 16: RR Joint 1 Angle and Velocity

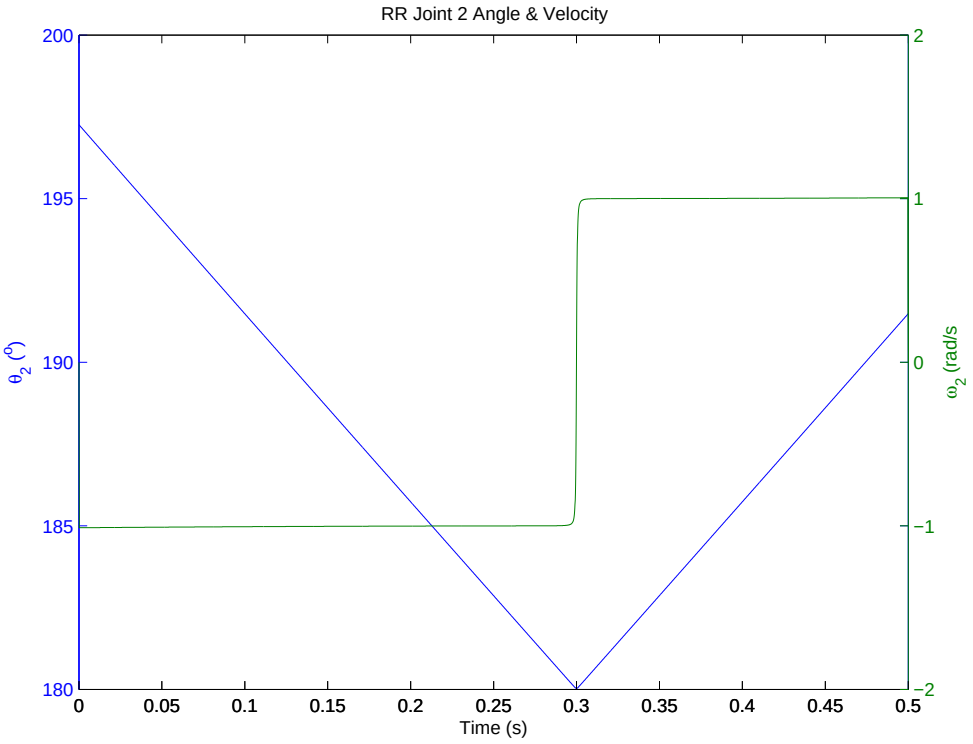

Figure 17: RR Joint 2 Angle and Velocity

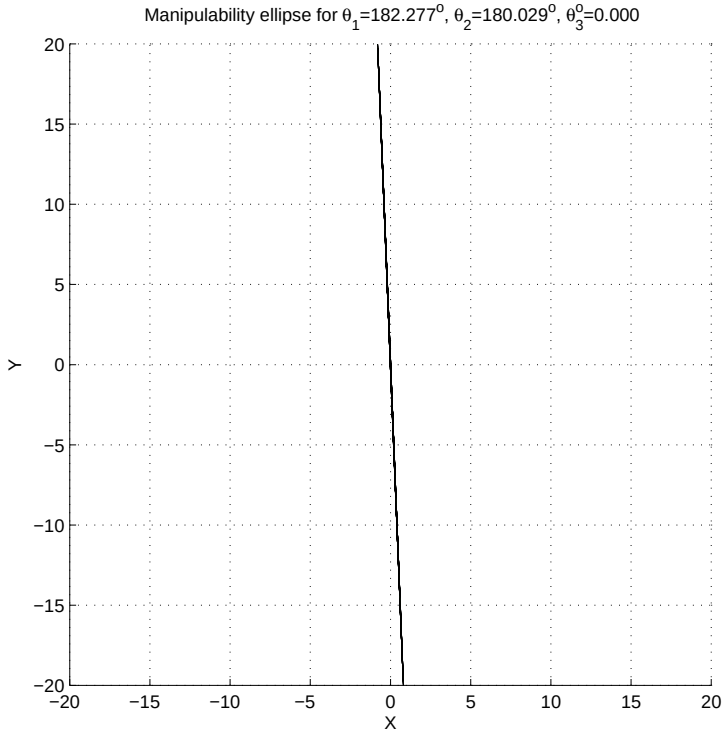

Figure 18: Manipulability of RR at position 3000.

RRR robot for dT=0.0001s.

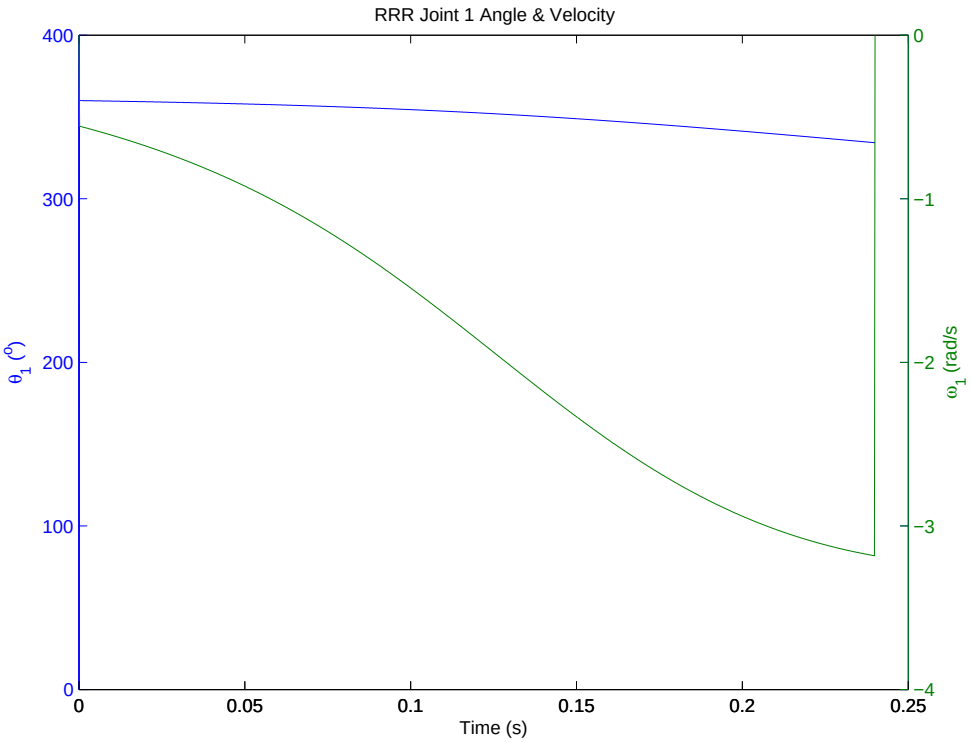

Figure 19: RRR Joint 1 Angle and Velocity

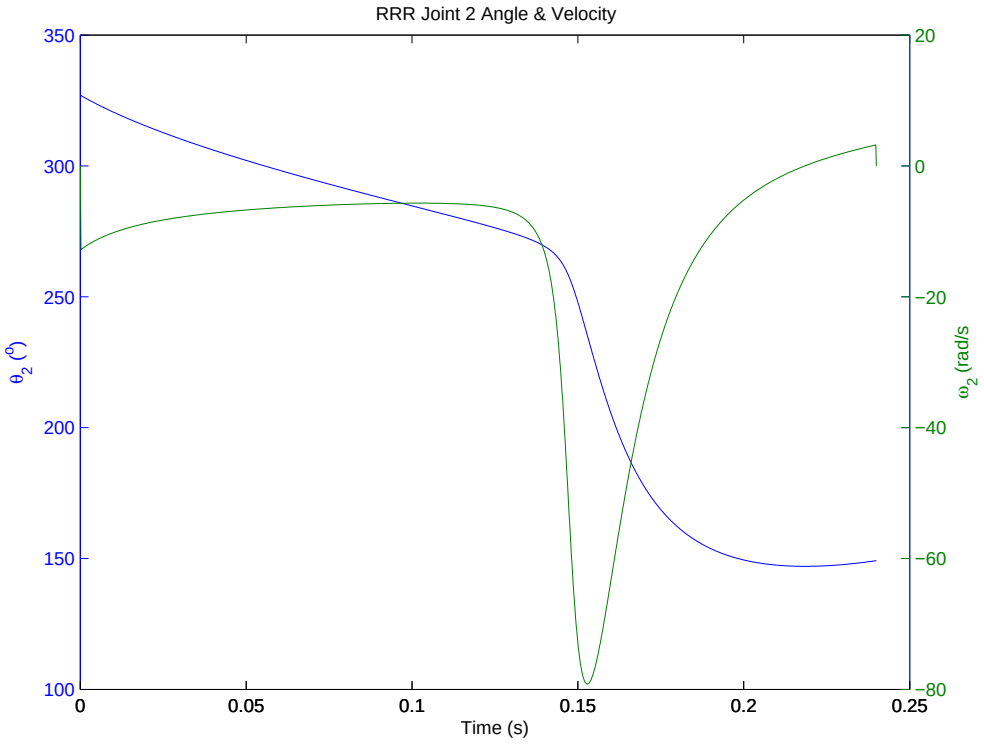

Figure 20: RRR Joint 2 Angle and Velocity

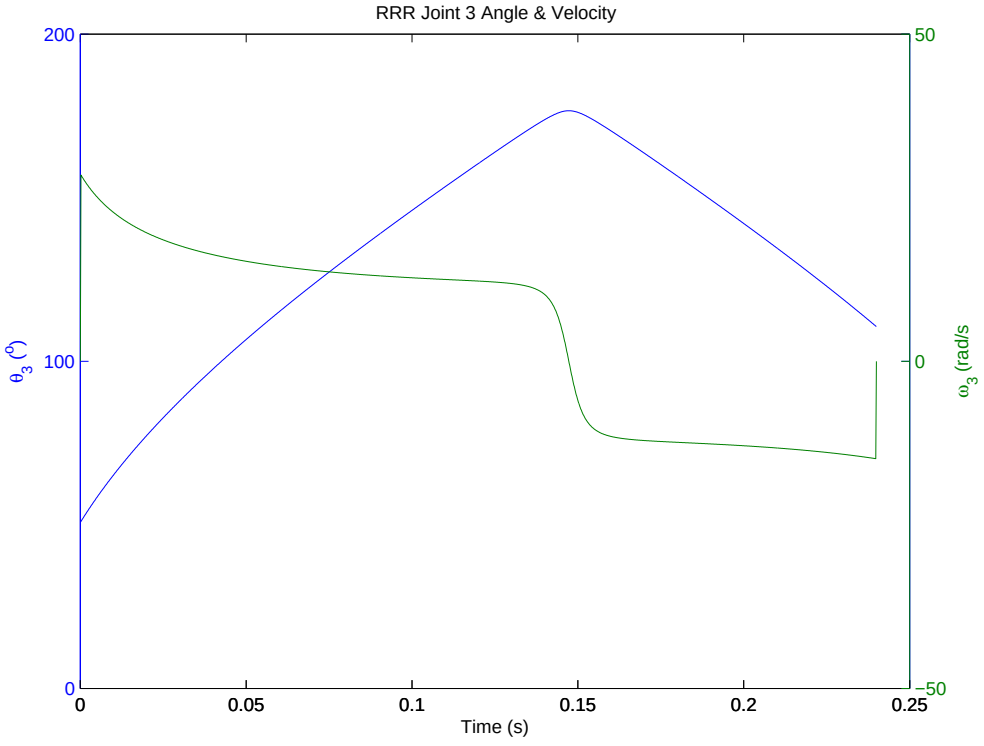

Figure 21: RRR Joint 3 Angle and Velocity

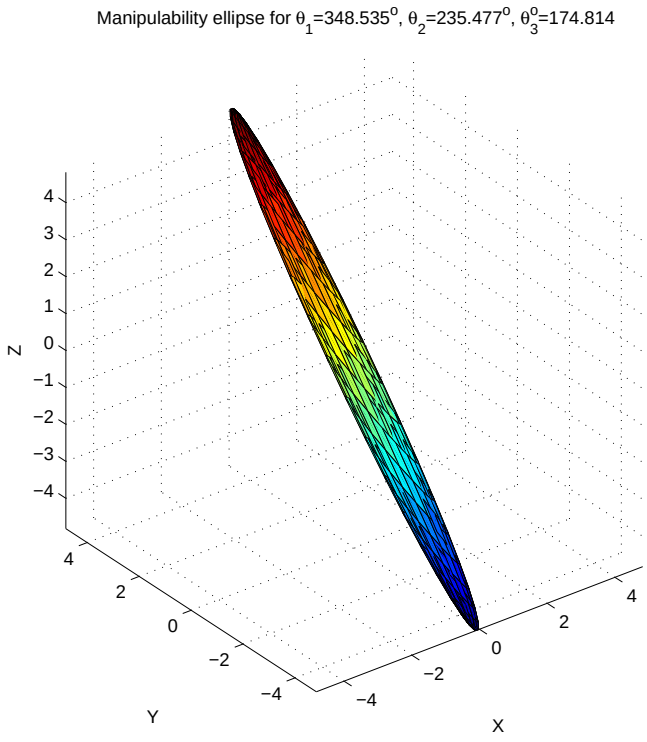

Figure 22: Manipulability of RRR at position 1528.

Appendices

M-Code

phase1.m

This has been slightly edited since last time to work better in the phase2 workflow.

```
function varargout=phase1(file)
error(nargchk(0, 1, nargin, 'struct'))
error(nargchk(0, 4, nargout, 'struct'))
if nargin<1||¬exist(file,'file')
    file='trajectory1_robot.dat';
else
    if strcmpi(file,'prompt')
        [filename, pathname] = uigetfile( ...
            {'*.dat';'*.txt';'*.*'}, ...
            'Pick a robot datafile');
        feval(mfilename,fullfile(pathname,filename));
        return;
    end
end
fid=fopen(file,'r');
n=fscanf(fid,'%d',1);
m=fscanf(fid,'%d',1);
robotType=fscanf(fid,'%s',1);
Q=fscanf(fid,'%f',[n,m]);
fclose(fid);
Tm=cell(1,m);
for i=1:m
    Tm_tmp=Tworld(robotType,Q(:,i));
    Tm{i}=Tm_tmp{end};
end
if nargout==3
    for i=1:m
        P(i,:)=Tm{i}(1:3,4)';
    end
    varargout{1}=P(:,1);
    varargout{2}=P(:,2);
    varargout{3}=P(:,3);
else
    [p,n]=fileparts(file);
    phase1Tex=fullfile(p,[n '_tm.tex']);
    if nargout==1
        varargout{1}=phase1Tex;
    end
    fid=fopen(phase1Tex,'w');
    fprintf(fid,'\\newpage\nWorld transforms of the robot at positions calculated for ...
        coordinate interpolated motion.\n');
    for i=1:m
        fprintf(fid,'\\begin{verbatim}\n');
        fprintf(fid,'POSITION %d: INPUT: JOINT VARIABLES (',i);
        fprintf(fid,'% 8.3f,',Q(:,i));
        fprintf(fid,')\nOUTPUT:');
        fprintf(fid,'% 10.3f',Tm{i}(1,:));
        fprintf(fid,'\n       ');
        fprintf(fid,'% 10.3f',Tm{i}(2,:));
        fprintf(fid,'\n       ');
        fprintf(fid,'%10.3f',Tm{i}(3,:));
        fprintf(fid,'\n       ');
        fprintf(fid,'%10.3f',Tm{i}(4,:));
        fprintf(fid,'\n');
```

```
54          fprintf(fid,'\\end{verbatim} \\pagebreak[1]');
55      end
56      fclose(fid);
57  end
58  end
59  function m=phiR(theta)
60  theta=theta.*pi./180;
61  m=[cos(theta) -sin(theta) 0 0
62      sin(theta)  cos(theta) 0 0
63      0            0          1 0
64      0            0          0 1];
65  end
66  function m=phiP(X)
67  m=[1 0 0 0
68      0 1 0 0
69      0 0 1 X
70      0 0 0 1];
71  end
```

phase2.m

Main program for phase 2.

```
1   function phase2(file)
2   close all;
3   error(nargchk(0, 1, nargin, 'struct'))
4   error(nargchk(0, 0, nargout, 'struct'))
5   if nargin<1||¬exist(file,'file')
6       file='trajectory1.dat';
7       feval(mfilename,file);
8       file='trajectory2.dat';
9       feval(mfilename,file);
10  %       file='trajectory1_slow.dat';
11  %       feval(mfilename,file);
12  %       file='trajectory2_slow.dat';
13  %       feval(mfilename,file);
14      return;
15  else
16      if strcmpi(file,'prompt')
17          [filename, pathname] = uigetfile( ...
18              {'*.dat';'*.txt';'*.*'}, ...
19              'Pick a robot datafile');
20          feval(mfilename,fullfile(pathname,filename));
21          return;
22      end
23  end
24  %% Data Input
25  % Open the data file.
26  fid=fopen(file,'r');
27  % Read in times.
28  Tf=fscanf(fid,'%f',1);
29  dT=fscanf(fid,'%f',1);
30  % Read in Initial Position.
31  Px(1)=fscanf(fid,'%f',1);
32  Py(1)=fscanf(fid,'%f',1);
33  Pz(1)=fscanf(fid,'%f',1);
34  % Read in Final Position.
35  Px(2)=fscanf(fid,'%f',1);
36  Py(2)=fscanf(fid,'%f',1);
```

```
Pz(2)=fscanf(fid,'%f',1);
% Open the TexFile that is used to write the outputs to.
[p,n]=fileparts(file);
texFid=fopen(fullfile(p,[n '.tex']),'w');
%% Calculations
t=[0:dT:Tf]';
[X,Vx,¬]=trajPlanner(Px(1),Px(2),Tf,dT,2,2); % Plan the X Path
[Y,Vy,¬]=trajPlanner(Py(1),Py(2),Tf,dT,2,2); % Plan the Y Path
[Z,Vz,¬]=trajPlanner(Pz(1),Pz(2),Tf,dT,2,2); % Plan the Z Path
% Determine robot type based on Z. If it's all 0 then it's just an RR.
if all(Z==0)
    robotType='RR';
else
    robotType='RRR';
end
%%
theta=invKin(robotType,[X Y Z]); % Do inverse kinematics.
dTheta=zeros(size(theta,1),3); % Preallocate for speed.
%% Calculate the Jacobian symbolically.
if strcmpi(robotType,'RRR')
    syms theta1 theta2 theta3 real;
    J=Jcalc(robotType,[theta1 theta2 theta3]);
else
    syms theta1 theta2 real;
    J=Jcalc(robotType,[theta1 theta2]);
end
dataOut=fullfile(p,[n '_jacobian.tex']);
fid=fopen(dataOut,'w');
fprintf(fid,'\\begin{eqnarray}\n');
for i=1:size(J,1)
    for j=1:size(J,2)
        if i==size(J,1) && j==size(J,2)
            fprintf(fid,'J(%d,%d)&=&%s\n \\nonumber',i,j,mat2latex(simplify(J(i,j))));
        else
            fprintf(fid,'J(%d,%d)&=&%s\n \\nonumber \\\\',i,j,mat2latex(simplify(J(i,j))));
        end
    end
end
fprintf(fid,'\\end{eqnarray}\n');
fprintf(texFid,'Symbolic jacobian matrix for %s.\n',robotType);
fprintf(texFid,'\\input{%s}\n',dataOut);% Write it to the main file to include.
%% Calculate numerically.
for i=1:size(theta,1) % For each position.
    J=Jcalc(robotType,theta(i,:)); % Calculate the Jacobian.
    switch robotType
        % Get the top Jacobian and calculate joint rates.
        case 'RR',
            JTop=J(1:2,1:2);
            J_inv=JTop^-1;
            dTheta(i,1:2)=[J_inv*[Vx(i) Vy(i)]']';
        case 'RRR',
            JTop=J(1:3,1:3);
            J_inv=JTop^-1;
            dTheta(i,:)=[J_inv*[Vx(i) Vy(i) Vz(i)]']';
    end
    % Get the manipulability.
    [ax(i),by(i),cz(i),Beta(i),R{i}]=manipulability(J);
end
% dTheta=dTheta.*180/pi; % Just to check answers against HW7
%% Position of the Joints
dataOut=fullfile(p,[n '_trajectory.tex']);
fid=fopen(dataOut,'w');
```

```
99  fprintf(fid,'\\begin{verbatim}\n');
100 fprintf(fid,'%d %% Number of positions\n',length(X));
101 if strcmpi(robotType,'RRR')
102     fprintf(fid,'%%  t(s) ...
            %9s%9s%9s%9s%9s%9s\n','Theta1','Theta2','Theta3','Omega1','Omega2','Omega3');
103     fprintf(fid,'% 8.3f % 8.3f % 8.3f % 8.3f % 8.3f % 8.3f % ...
            8.3f\n',[t,theta(:,1),theta(:,2),theta(:,3),dTheta(:,1),dTheta(:,2),dTheta(:,3)]');
104 else
105     fprintf(fid,'%%  t(s) %9s%9s%9s%9s\n','Theta1','Theta2','Omega1','Omega2');
106     fprintf(fid,'% 8.3f % 8.3f % 8.3f % 8.3f % ...
            8.3f\n',[t,theta(:,1),theta(:,2),dTheta(:,1),dTheta(:,2)]');
107 end
108 fprintf(fid,'\\end{verbatim}\n');
109 fclose(fid);
110 fprintf(texFid,'Joint position and velocities for %s.\n',[n '.dat']);
111 fprintf(texFid,'\\input{%s}\n',dataOut);% Write it to the main file to include.
112
113 %% Joint positions & velocities.
114 [¬,n] = fileparts(file);
115
116 % Determine the number of joints to print out.
117 if strcmpi(robotType,'RR')
118     q=2;
119 else
120     q=3;
121 end
122 for i=1:q
123     close all;
124     h=plotyy(t,theta(:,i),t,dTheta(:,i));
125     xlabel('Time (s)');
126     ylabel(h(1),sprintf('\\theta_%d (^o)',i));
127     ylabel(h(2),sprintf('\\omega_%d (rad/s)',i));
128     title(sprintf('%s Joint %d Angle & Velocity',robotType,i));
129     jointFile=sprintf('%s_joint%d',n,i);
130     % print('--depsc2',jointFile);
131     fprintf(texFid,'\\fig{%s}{%s Joint %d Angle and Velocity}\n',jointFile,robotType,i);
132 end
133
134 %% Plot the EECS
135 close all; % New figure & increment.
136 subplot(q,1,1); % New sub plot with the number of axes
137 plot(t,X,'.');axis('tight'); % Plot X
138 ylabel('X'); % Label
139 xlabel('Time (s)');
140 subplot(q,1,2);
141 plot(t,Y,'.');axis('tight'); % Plot Y
142 ylabel('Y'); % Label
143 xlabel('Time (s)');
144 set(gca,'YAxisLocation','Right'); % Flip the axes to the other side for readability.
145 if strcmpi(robotType,'RRR'); % If it has a 3rd linkage, print the 3rd dimension.
146     subplot(313);
147     plot(t,Z,'.');axis('tight');
148     ylabel('Z');
149     xlabel('Time (s)');
150 end
151 eecsFile=sprintf('%s_EECS',n);
152 % print('--depsc2',eecsFile);
153 fprintf(texFid,'\\fig{%s}{Position of the end effector for each of the coordinates vs ...
        time}\n',eecsFile);
154 % Now in 3D
155 close all;
156 plot3(X,Y,Z);
```

```
157 title(sprintf('%s end effector position in 3D',robotType));
158 xlabel('X');
159 ylabel('Y');
160 zlabel('Z');
161 grid('on');
162 eecsFile3=sprintf('%s_EECS_3D',n);
163 % print('-depsc2',eecsFile3);
164 fprintf(texFid,'\\fig{%s}{%s end effector position in 3D}\n',eecsFile3,robotType);
165
166 %% Do Phase 1 & Forward Kinematics.
167 [p,n] = fileparts(file);
168 phase1_file=fullfile(p,[n '_robot.dat']); % Create the .dat file to write for phase1.
169 fid=fopen(phase1_file,'w'); % Open it.
170 fprintf(fid,'%d %d %s\n',2+~all(Z==0),length(X),robotType); % Write the first line.
171 if strcmpi('RRR',robotType)
172     fprintf(fid,'%.10f\t%.10f\t%.10f\n',[theta(:,1),theta(:,2),theta(:,3)]'); % Print each ...
        of the joint values.
173 else
174     fprintf(fid,'%.10f\t%.10f\n',[theta(:,1),theta(:,2)]'); % Print each of the joint values.
175 end
176 fclose(fid); % Close the file.
177 [X2,Y2,Z2]=feval('phase1',phase1_file); % Get the EECS position based on forward kinematics.
178 phase1Tex= feval('phase1',phase1_file); % Print phase1 to a .tex file.
179 fprintf(texFid,'\\input{%s}\n',phase1Tex);% Write it to the main file to include.
180
181 %% Manipulability.
182 [p,n] = fileparts(file);
183 dataOut=fullfile(p,[n '_manip.tex']);
184 fid=fopen(dataOut,'w');
185 fprintf(fid,'Manipulability of the robot.\n \\pagebreak[1]');
186 fprintf(fid,'\\begin{verbatim}\n');
187 if strcmpi(robotType,'RRR')
188     fprintf(fid,'%%  t(s)%9s%9s%9s\n','ax','by','cz');
189     fprintf(fid,'% 8.3f% 8.3f % 8.3f % 8.3f\n',[t';ax;by;cz]);
190 else
191     fprintf(fid,'%%  t(s)%9s%9s\n','ax','by');
192     fprintf(fid,'% 8.3f% 8.3f % 8.3f\n',[t';ax;by]);
193 end
194 fprintf(fid,'\\end{verbatim}\n');
195 fprintf(texFid,'\\input{%s}\n',dataOut);% Write it to the main file to include.
196
197 %% Plot the ellipsoid.
198 [ix,iy]=find(abs(dTheta)==max(max(abs(dTheta)))); % Find the point where joint velocity is ...
    near the maximum.
199
200 if strcmpi('RRR',robotType)
201     for i=[1 ix length(ax)]
202         [Ex,Ey,Ez]=ellipsoid(0,0,0,ax(i),by(i),cz(i));
203         % Rotate each Point
204         for j=1:size(Ex,1)
205             for k=1:size(Ex,2)
206                 E=R{i}*[Ex(j,k) Ey(j,k) Ez(j,k)]';
207                 Ex(j,k)=E(1);
208                 Ey(j,k)=E(2);
209                 Ez(j,k)=E(2);
210             end
211         end
212         close all;
213         surf(Ex,Ey,Ez)
214         title(sprintf('Manipulability ellipse for \\theta_1=%.3f^o, \\theta_2=%.3f^o, ...
            \\theta_3^o=%.3f', theta(i,1), theta(i,2),theta(i,3)))
215         axis equal
```

```
216         axis([-1 1 -1 1 -1 1]*max(axis));
217         xlabel('X');
218         ylabel('Y');
219         zlabel('Z');
220         eFile=sprintf('%s_ell%d',n,i);
221         % print('--depsc2',eFile);
222         fprintf(texFid,'\\fig{%s}{Manipulability of %s at position ...
                %d.}\n',eFile,robotType,i-1);
223     end
224 else
225     for i=[1 ix length(ax)]
226         % 2D Way
227 %         [ex,ey] = calculateEllipse(0, 0, ax(i), by(i), Beta(i));
228 %         close all;
229 %         plot(ex,ey);
230 %         title(sprintf('Manipulability ellipse for \\theta_1=%.3f^o, \\theta_2=%.3f^o', ...
    theta(i,1), theta(i,2)))
231 %         axis equal
232 %         axis([-1 1 -1 1]*max(axis));
233 %         xlabel('X');
234 %         ylabel('Y');
235
236         % 3D Way
237         [Ex,Ey,Ez]=ellipsoid(0,0,0,ax(i),by(i),0);
238         % Rotate each Point
239         for j=1:size(Ex,1)
240             for k=1:size(Ex,2)
241                 E=R{i}*[Ex(j,k) Ey(j,k)]';
242                 Ex(j,k)=E(1);
243                 Ey(j,k)=E(2);
244             end
245         end
246         close all;
247         surf(Ex,Ey,Ez);
248         title(sprintf('Manipulability ellipse for \\theta_1=%.3f^o, \\theta_2=%.3f^o, ...
                \\theta_3^o=%.3f', theta(i,1), theta(i,2),theta(i,3)))
249         axis equal
250         axis([-1 1 -1 1 -1 1]*ceil(max(axis)));
251         xlabel('X');
252         ylabel('Y');
253         zlabel('Z');
254         eFile=sprintf('%s_ell%d',n,i);
255         % View from the top.
256         set(gca,'CameraPosition',[0 0 1])
257         % print('--depsc2',eFile);
258         fprintf(texFid,'\\fig{%s}{Manipulability of %s at position ...
                %d.}\n',eFile,robotType,i-1);
259     end
260 end
261 %% Sanity check the CIM trajectory planning against the output from phase1
262 if max(abs(X-X2))>.1
263     error('Something went wrong with forward kinematics. Check phase1.m and shape.m');
264 else
265     disp('EECS X-axis validated');
266 end
267 if max(abs(Y-Y2))>.1
268     error('Something went wrong with forward kinematics. Check phase1.m and shape.m');
269 else
270     disp('EECS Y-axis validated');
271 end
272 if max(abs(Z-Z2))>.1
273     error('Something went wrong with forward kinematics. Check phase1.m and shape.m');
```

```
274 else
275     disp('EECS Z-axis validated');
276 end
277 %%
278 fclose('all');
```

JCalc.m

Jacobian calculator.

```
1  function J=Jcalc(robotType,Q)
2  [jointType,~]=shape(robotType);
3  [Tm,n]=Tworld(robotType,Q);
4  % Get the final P position.
5  Pe=Tm{end}(1:3,4);
6  % For each link find out the Jacobian column.
7  for i=1:n
8      switch jointType(i)
9          case 0, % Rotation joint
10             an=Tm{i}(1:3,3);
11             Pn=Tm{i}(1:3,4);
12             J(:,i)=[cross(an,Pe-Pn);an];
13         case 1, % Prismatic joint
14             an=Tm{i}(1:3,3);
15             J(:,i)=[an;0;0;0];
16         otherwise,
17             error('Unknown joint type %d',jointType(i))
18     end
19 end
20 % Clean up symbolic stuff to make it more readable.
21 if ~isnumeric(J)
22     J=simplify(J);
23 end
24 end
```

Tworld.m

Calculate T_w and return it. This was pulled out of phase1 and made its own function because it was used multiple times in multiple different functions. (Such as in the Jacobian)

```
1  function [Tm,n]=Tworld(robotType,Q)
2  % This us used so often, make it a new function.
3
4  % Get data from the shape matrix.
5  [jointType,T]=shape(robotType);
6  n=length(jointType);
7  if length(Q)<n % Number of input variables has to be the same as the number of transforms.
8      error('Incorrect number of input parameters. There are %d joints but only %d variables ...
           specified',n,length(Q));
9  end
10 Tm=cell(1,n+1); % Empty cell array for speed.
11 Tm{1}=eye(4); % Initial one is an identity. Because of Matlab's 1 indexing everything is ...
       actually n-1.
12 for j=1:length(jointType)
13     switch jointType(j)
14         case 0,
15             phi=phiR(Q(j));
```

```
16          case 1,
17              phi=phiP(Q(j));
18          otherwise,
19              error('Unknown joint type %d',jointType(j))
20      end
21      % Find each intermediate transform matrix.
22      Tm{j+1}=Tm{j}*phi*T{j};
23  end
24  end
25  % Sub functions
26  function m=phiR(theta)
27  if isnumeric(theta)
28      theta=theta*pi/180;
29  end
30  m=[cos(theta) -sin(theta) 0 0
31      sin(theta)  cos(theta) 0 0
32      0            0          1 0
33      0            0          0 1];
34  end
35  function m=phiP(X)
36  m=[1 0 0 0
37      0 1 0 0
38      0 0 1 X
39      0 0 0 1];
40  end
```

shape.m

New shape.m function with the added RR.

```
1   function [jointType,T]=shape(robotType)
2   switch robotType
3       case 'RR', % From homework 7
4           jointType=[0 0];
5           L1=20;
6           L2=20;
7           % T1
8           T{1}=[1 0 0 L1
9               0 1 0  0
10              0 0 1  0
11              0 0 0  1];
12          % T2
13          T{2}=[1 0 0 L2
14              0 1 0  0
15              0 0 1  0
16              0 0 0  1];
17          return;
18      case 'RRP', % From homework 4
19          jointType=[0 0 1];
20          % Joint parameters
21          L1=5;
22          d1=4;
23          % T1
24          T{1}=[0 0 1 d1
25              1 0 0 0
26              0 1 0 L1
27              0 0 0 1];
28          % T2
29          T{2}=[0 1 0 0
```

```
            0 0 1 0
            1 0 0 0
            0 0 0 1];
        % T3
        T{3}=eye(4);
        return
    case 'RRR', % From homework 4
        d1=4;
        L3=4;
        L1=5;
        L2=5;
        jointType=[0 0 0];
        T{1}=[0 0 1 d1
            1 0 0 0
            0 1 0 L1
            0 0 0 1];
        T{2}=[1 0 0 0
            0 1 0 L2
            0 0 1 0
            0 0 0 1];
        T{3}=[0 1 0 0
            0 0 1 L3
            1 0 0 0
            0 0 0 1];
    otherwise,
        error('Unknown robot type %s',robotType)
end

end
```

manipulability.m

Manipulability calculator.

```
function [ax,by,cz,Beta,R]=manipulability(J)
n=min([size(J,2) 3]);
Jtop=J(1:n,1:n);
B=Jtop*Jtop';
[R,L1]=eigs(B^-1);
E=diag(L1);
R1=R*sign(R(1,1));
ax=1/sqrt(E(1));
by=1/sqrt(E(2));
Beta=acos(R1(1,1));
if n==3
    cz=1/sqrt(E(3));
else
    cz=NaN;
end
```