July 2021

01211433 Homework # 1

Using RTSX toolbox, below is Scilab script to plot 3-segment quintic polynomial trajectory in Figure 1

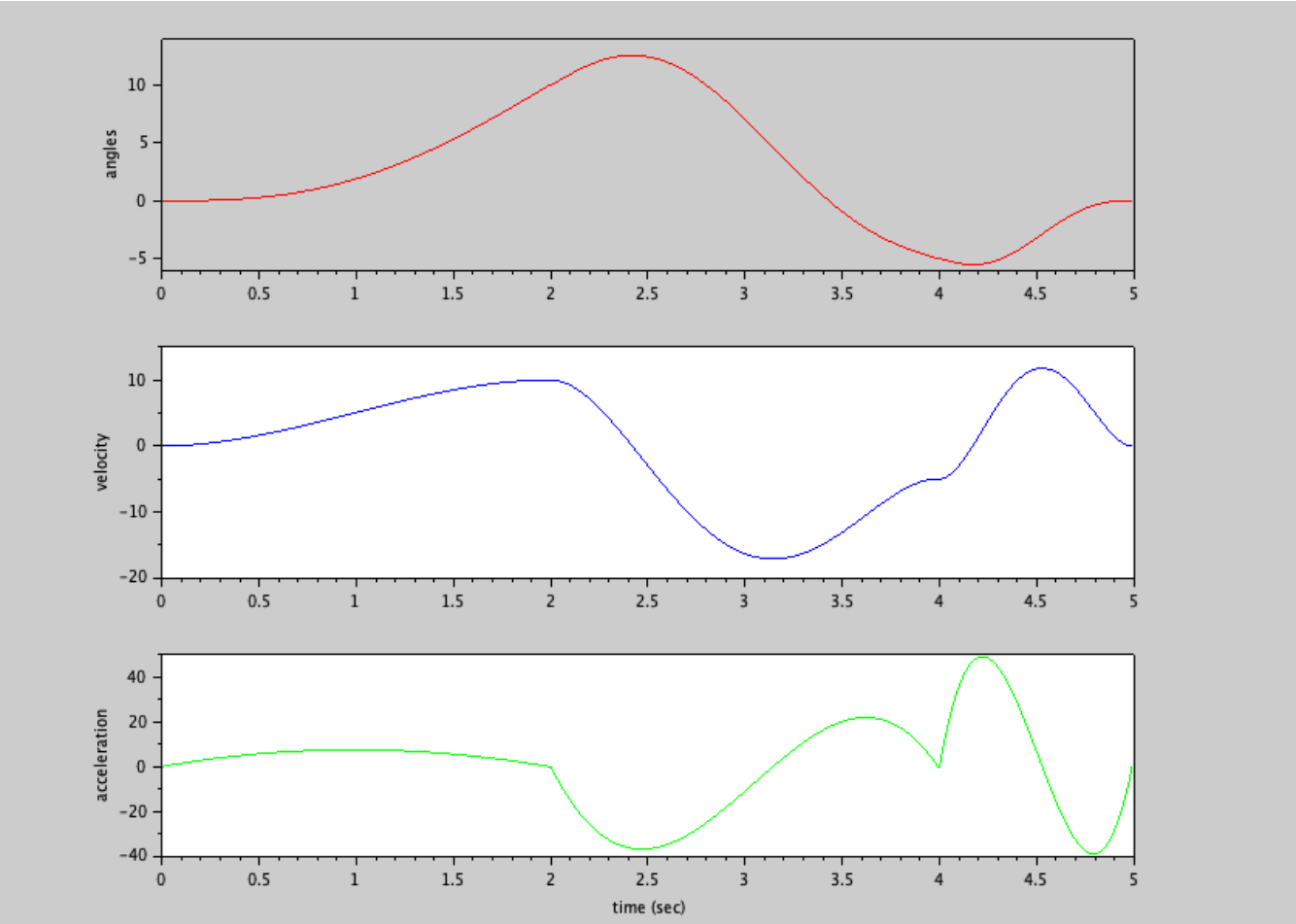


Figure 1 3-segment quintic polynomial trajectory

```
// #Scilab script example. Do not run this cell in notebook.
// #command trajectory generation

t1=0:0.01:1.99;
t2 = 0:0.01:0.99;

// joint 1 trajectory
[q11,qd11,qdd11]=qpoly(0,10,t1,0,10);  // segment 1
[q12,qd12,qdd12]=qpoly(10,-5,t1,10,-5);  // segment 2
[q13,qd13,qdd13]=qpoly(-5,0,t2,-5,0);  // segment 3

q1 = [q11;q12;q13];     // position (angles)
qd1 = [qd11;qd12;qd13];  // velocity
qdd1 = [qdd11;qdd12;qdd13];  // acceleration

t=0:0.01:4.99;
t=t';

figure(1);
subplot(311), plot(t, q1,'r-')
//xlabel('time (sec)')
ylabel('angles')

subplot(312), plot(t, qd1,'b-')
//xlabel('time (sec)')
ylabel('velocity')

subplot(313), plot(t, qdd1, 'g-')
xlabel('time (sec)')
ylabel('acceleration')
```

Here is Scilab script for qpoly function

```
// #Scilab script. Do not run this cell in notebook.
function [s,sd,sdd] = qpoly(q0, qf, t, qd0, qdf)

    t0 = t;
    nargin=argn(2);
    nargout = argn(1);
    if isscalar(t)
        t = (0:t-1)';
    else
        t = t(:);
    end
    if nargin < 4
        qd0 = 0;
    end
    if nargin < 5
        qdf = 0;
    end

    tf = max(t);

    // solve for the polynomial coefficients using least squares
    X = [
        0           0           0         0       0   1
        tf^5        tf^4        tf^3      tf^2    tf  1
        0           0           0         0       1   0
        5*tf^4      4*tf^3      3*tf^2    2*tf    1   0
        0           0           0         2       0   0
        20*tf^3     12*tf^2     6*tf      2       0   0
    ];
    b = [q0 qf qd0 qdf 0 0]';
    coeffs = (inv(X)*b)';
    // coefficients of derivatives
    coeffs_d = coeffs(1:5) .* (5:-1:1);
    coeffs_dd = coeffs_d(1:4) .* (4:-1:1);

    // evaluate the polynomials
    p = polyval(coeffs, t);
    pd = polyval(coeffs_d, t);
    pdd = polyval(coeffs_dd, t);

    select nargout

        case 1
            s = p;
        case 2
            s = p;
            sd = pd;
        case 3
            s = p;
            sd = pd;
            sdd = pdd;
    end

endfunction
```

Use Julia to generate the same trajectory. The position, velocity, and acceleration plot should resemble Figure 1. For simplicity, assume that t argument is always a vector in qpoly() function

Solution

```
#import Pkg
#Pkg.add("Plots")
using Plots
```

```
qpoly (generic function with 3 methods)
```

```
function qpoly(q0,qf,t,qd0=0,qdf=0)
    tf = t[end]
    T = [0 0 0 0 0 1;tf^5 tf^4 tf^3 tf^2 tf 1; 0 0 0 0 1 0; 5tf^4 4tf^3 3tf^2 2tf 1 0;0 0 0 2 0 0; 20tf^3 12tf^2 6tf 2 0 0]
    b = [q0 qf qd0 qdf 0 0]'
    coeffs = inv(T)*b
    coeffs_d = coeffs[1:5].*reverse(collect(1:5))
    coeffs_dd = coeffs_d[1:4].*reverse(collect(1:4))

    q = polyval(coeffs, t)
    q_d = polyval(coeffs_d,t)
    q_dd = polyval(coeffs_dd, t)
    return q, q_d, q_dd
end
```

```
polyval (generic function with 1 method)
```

```
function polyval(P,X)
    Y = zeros(size(X))
    n = size(P,1)
    for i=1:n
        Y = Y + P[i]*X.^(n-i)
    end
    return Y
end
```

```
[0.0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, (
```

```
begin
    t1 = collect(0:0.01:2);
    t2 = collect(0:0.01:1);
end
```

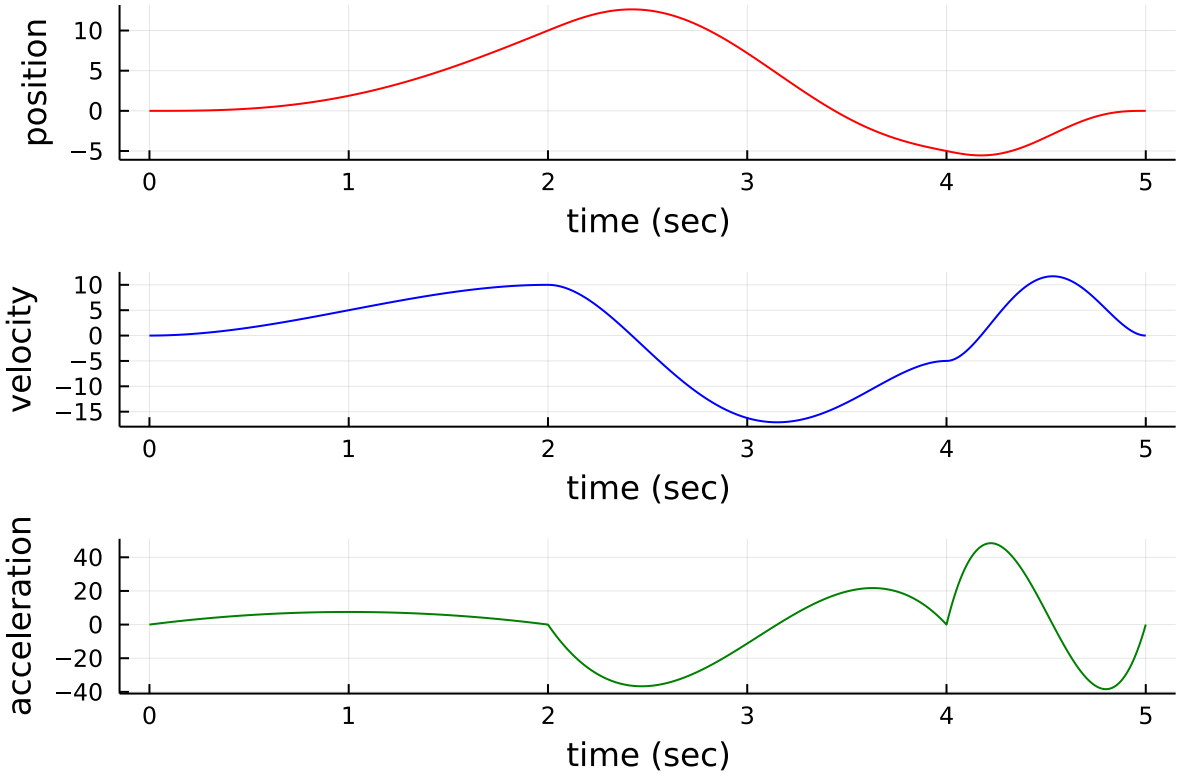
```
([-5.0, -5.04992, -5.09938, -5.14793, -5.19517, -5.2407, -5.28418, -5.32525, -5.3636,
```

```
begin
    q1,qd1,qdd1 = qpoly(0,10,t1,0,10)
    q2,qd2,qdd2 = qpoly(10,-5,t1,10,-5)
    q3,qd3,qdd3 = qpoly(-5,0,t2,-5,0)
end
```

```
[0.0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, (
```

```
begin
    q = [q1; q2; q3]
    qd = [qd1; qd2; qd3]
    qdd = [qdd1; qdd2; qdd3]
    tvec = [t1; 2 .+ t1; 4 .+ t2]
end
```

พล็อตแนววิถีได้ดังนี้




```
begin
    plot_q = plot(tvec, q, ylabel="position",color="red")
    plot_qd = plot(tvec, qd, ylabel="velocity",color="blue")
    plot_qdd = plot(tvec, qdd, ylabel="acceleration",color="green")
    plotall = plot(plot_q, plot_qd, plot_qdd,layout=(3,1),legend=false)
    xlabel!("time (sec)")
end
```