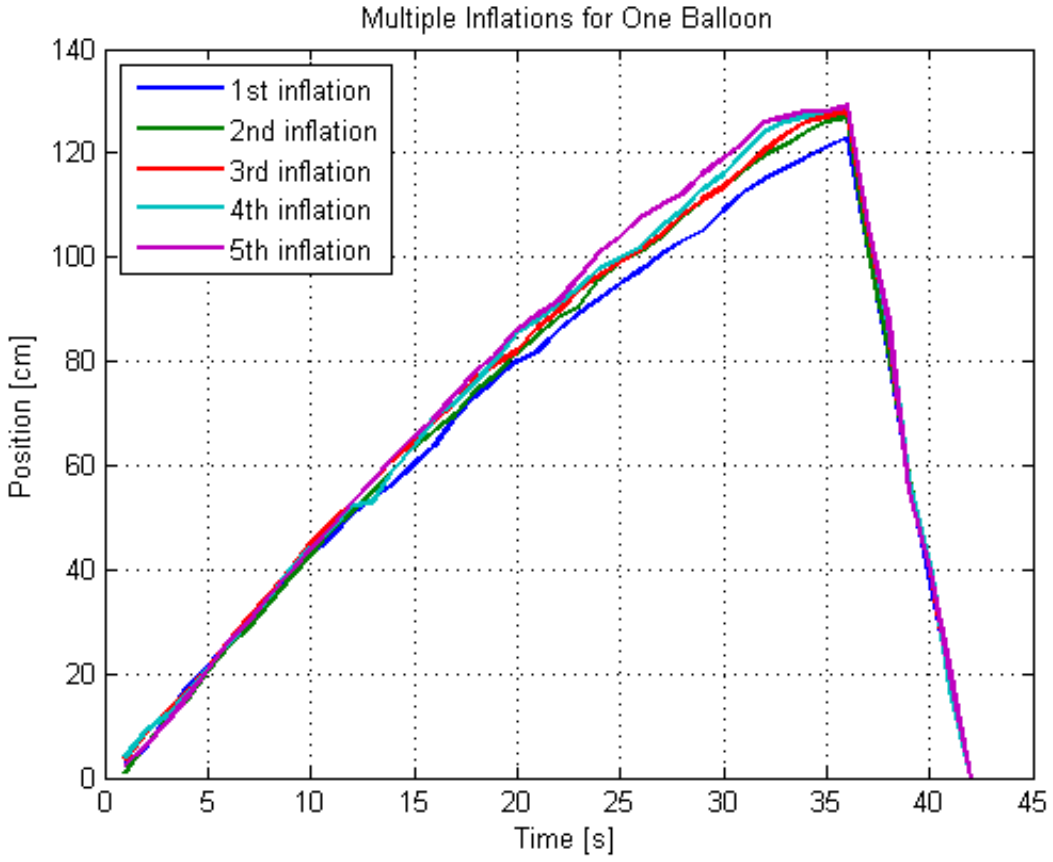
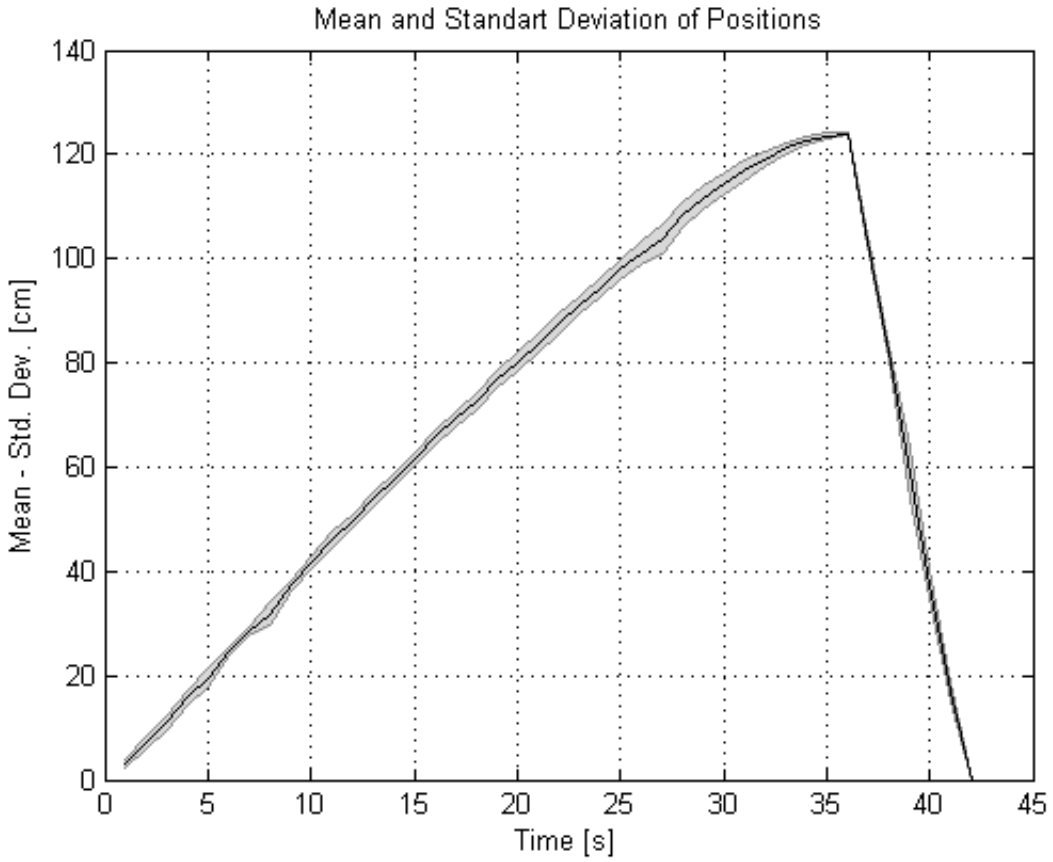
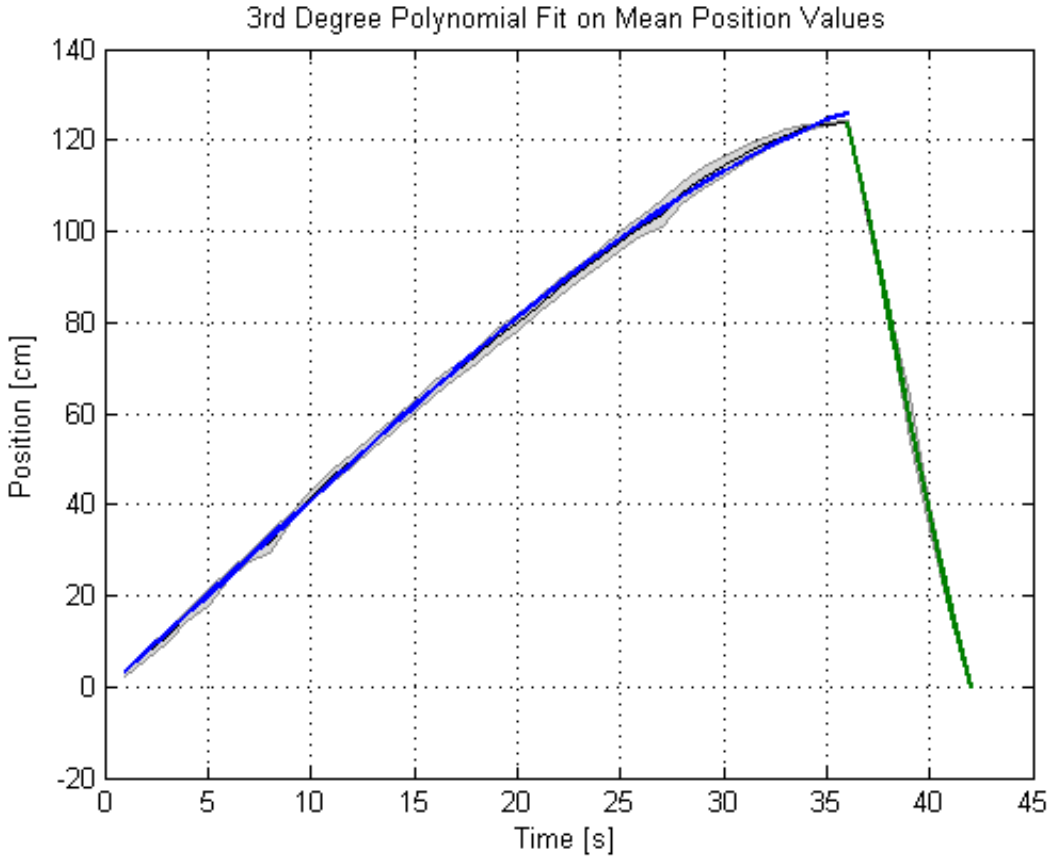
```
clc
close all
clear all
load('bal_dvt')       %HAS TO BE IN THE PATH

t1=1:size(dvt1,1);

figure
plot(t1,dvt1(:,1),t1,dvt1(:,2),t1,dvt1(:,3),t1,dvt1(:,4),t1,dvt1(:,5),'LineWidth',2)
legend({'1st inflation','2nd inflation','3rd inflation',...
    '4th inflation','5th inflation'},'Location','Northwest')
title('Multiple Inflations for One Balloon')
xlabel('Time [s]')
ylabel('Position [cm]')
grid on

for i=1:size(dvt1,1)
    std11(i,1)=std(dvt1(i,1:2));
    m11(i,1)=mean(dvt1(i,1:2));
    std12(i,1)=std(dvt1(i,3:5));
    m12(i,1)=mean(dvt1(i,3:5));
end

for i=1:size(dvt2,1)
    std21(i,1)=std(dvt2(i,1:2));
    m21(i,1)=mean(dvt2(i,1:2));
    std22(i,1)=std(dvt2(i,3:5));
    m22(i,1)=mean(dvt2(i,3:5));
end

for i=1:size(dvt3,1)
    std31(i,1)=std(dvt3(i,1:2));
    m31(i,1)=mean(dvt3(i,1:2));
    std32(i,1)=std(dvt3(i,3:5));
    m32(i,1)=mean(dvt3(i,3:5));
end

for i=1:size(dvt4,1)
    std41(i,1)=std(dvt4(i,1:2));
    m41(i,1)=mean(dvt4(i,1:2));
    std42(i,1)=std(dvt4(i,3:5));
    m42(i,1)=mean(dvt4(i,3:5));
end

for i=1:size(dvt5,1)
    std51(i,1)=std(dvt5(i,1:2));
    m51(i,1)=mean(dvt5(i,1:2));
    std52(i,1)=std(dvt5(i,3:5));
    m52(i,1)=mean(dvt5(i,3:5));
end

for i=1:size(dvt6,1)
    std61(i,1)=std(dvt6(i,1:2));
    m61(i,1)=mean(dvt6(i,1:2));
    std62(i,1)=std(dvt6(i,3:5));
    m62(i,1)=mean(dvt6(i,3:5));
end

for i=1:size(dvt7,1)
```

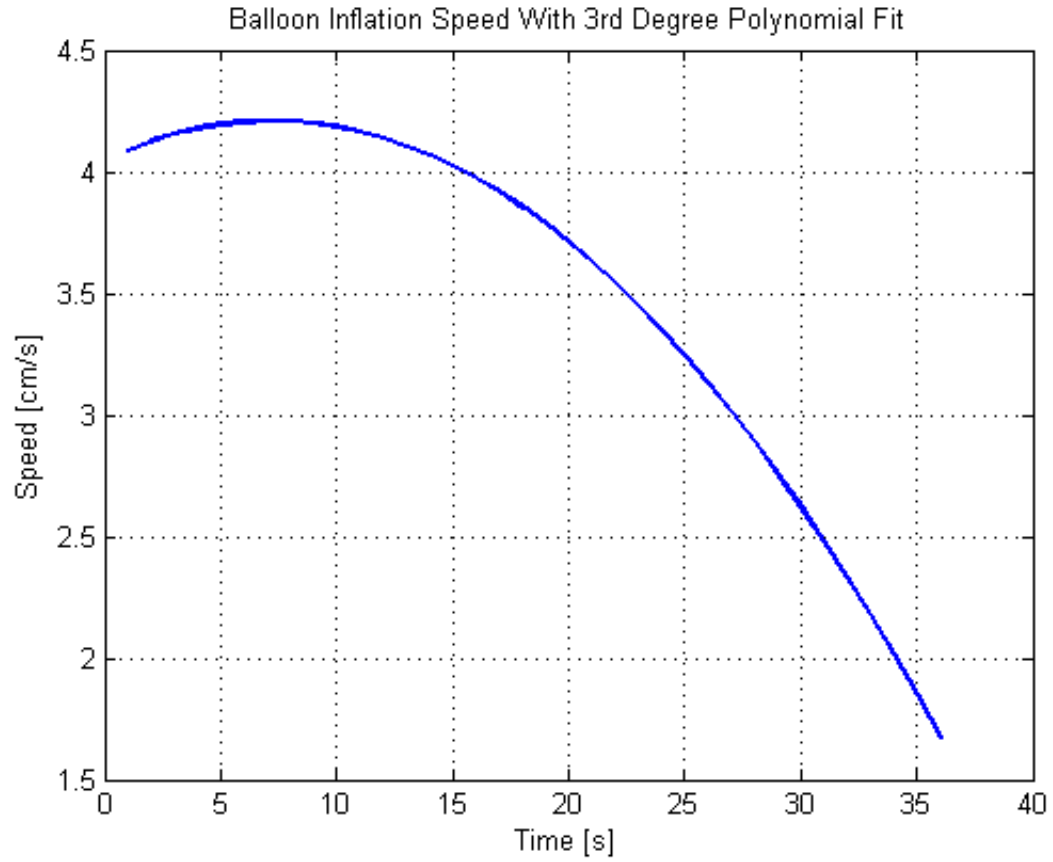
```
    std71(i,1)=std(dvt7(i,1:2));
    m71(i,1)=mean(dvt7(i,1:2));
    std72(i,1)=std(dvt7(i,3:5));
    m72(i,1)=mean(dvt7(i,3:5));
end


%OVERALL MEAN AND STANDART DEVIATION

mean1=mean([m11 m21 m31 m41 m51 m61 m71],2);
std1=mean([std11 std21 std31 std41 std51 std61 std71],2);
mean2=mean([m12 m22 m32 m42 m52 m62 m72],2);
std2=mean([std12 std22 std32 std42 std52 std62 std72],2);
mean_ov=mean([mean1 mean2],2);   %overall mean
std_ov=mean([std1 std2],2);   %mean of std. devs
%OR SHADED ERROR BAR????


%Plot for overall mean and overall std
figure
shadedErrorBar(t1,mean_ov,std_ov)
title('Mean and Standart Deviation of Positions')
xlabel('Time [s]')
ylabel('Mean - Std. Dev. [cm]')
grid on


t1=1:36; t2=36:42;
p_inf=polyfit((1:36)',mean_ov(1:36),3);
s_inf=polyder(p_inf);
p_inf=polyval(p_inf,(1:36));
s_inf=polyval(s_inf,1:36);

p_exh=polyfit((36:42)',mean_ov(36:42),3);
s_exh=polyder(p_exh);
p_exh=polyval(p_exh,(36:42));
s_exh=polyval(s_exh,36:0.2:42);


figure
shadedErrorBar(1:42,mean_ov,std_ov)
hold on
plot(t1,p_inf,t2,p_exh,'LineWidth',2)
grid on
title('3rd Degree Polynomial Fit on Mean Position Values')
xlabel('Time [s]')
ylabel('Position [cm]')

%Speed polyfit plots
figure
plot(t1,s_inf,'LineWidth',2)
title('Balloon Inflation Speed With 3rd Degree Polynomial Fit')
xlabel('Time [s]')
ylabel('Speed [cm/s]')
grid on

figure
plot(1:31,s_exh,'LineWidth',2)
title('Balloon Deflation Speed With 3rd Degree Polynomial Fit')
xlabel('Time [s]')
ylabel('Speed [cm/s]')
```

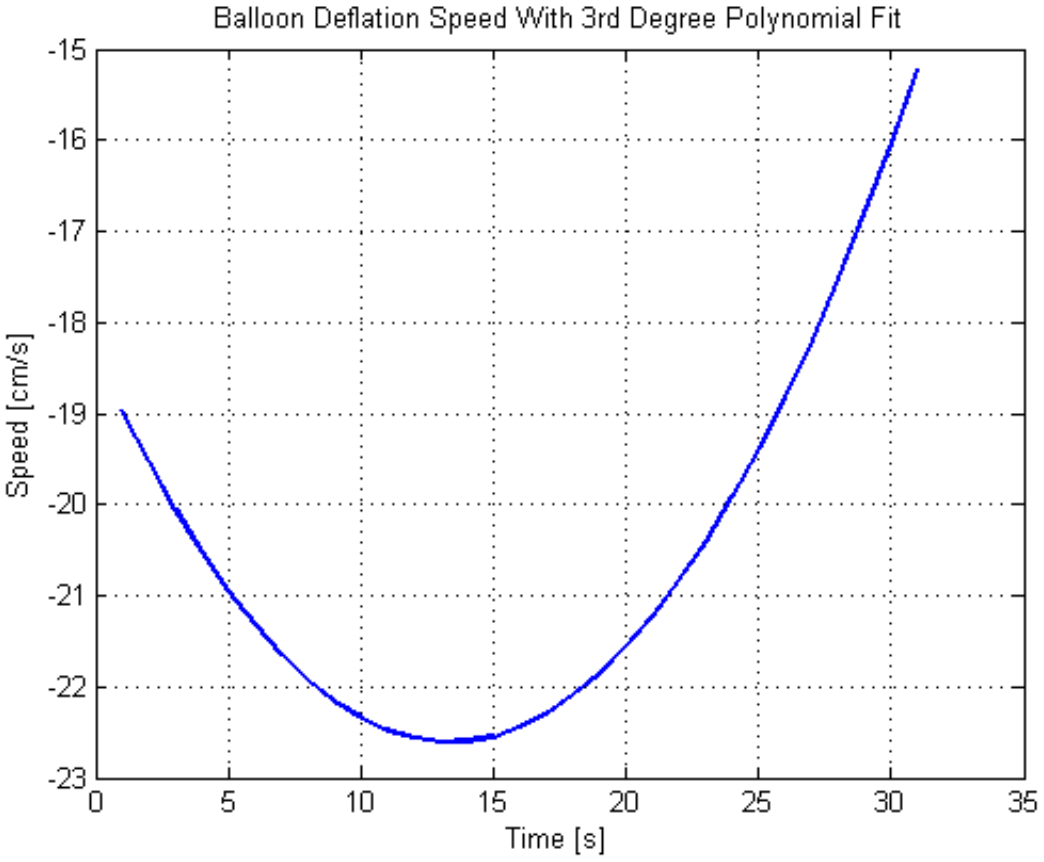
```
grid on

%Force Plot

for i=1:7
    n=1;
    for j=1:size(force1,1)
        if force1(j,i)>0.1
            c(n,i)=force1(j,i);
            n=n+1;
        end
    end
end

t=1:size(c,1);

figure
plot(t,c(:,1),t,c(:,2),t,c(:,3),t,c(:,4),t,c(:,5),t,c(:,6),t,c(:,7))
grid on
xlabel('Time [ms]')
ylabel('Force [N]')
title('Applied Axial Force')



figure
i=1:50;
plot(i,dvt_pwm,'LineWidth',2)
grid on
title('Balloon Inflation at Different Pulse Widths')
xlabel('Time [s]')
ylabel('Distance [cm]')
legend('100% On','66% On','33% On','Location','east')
```

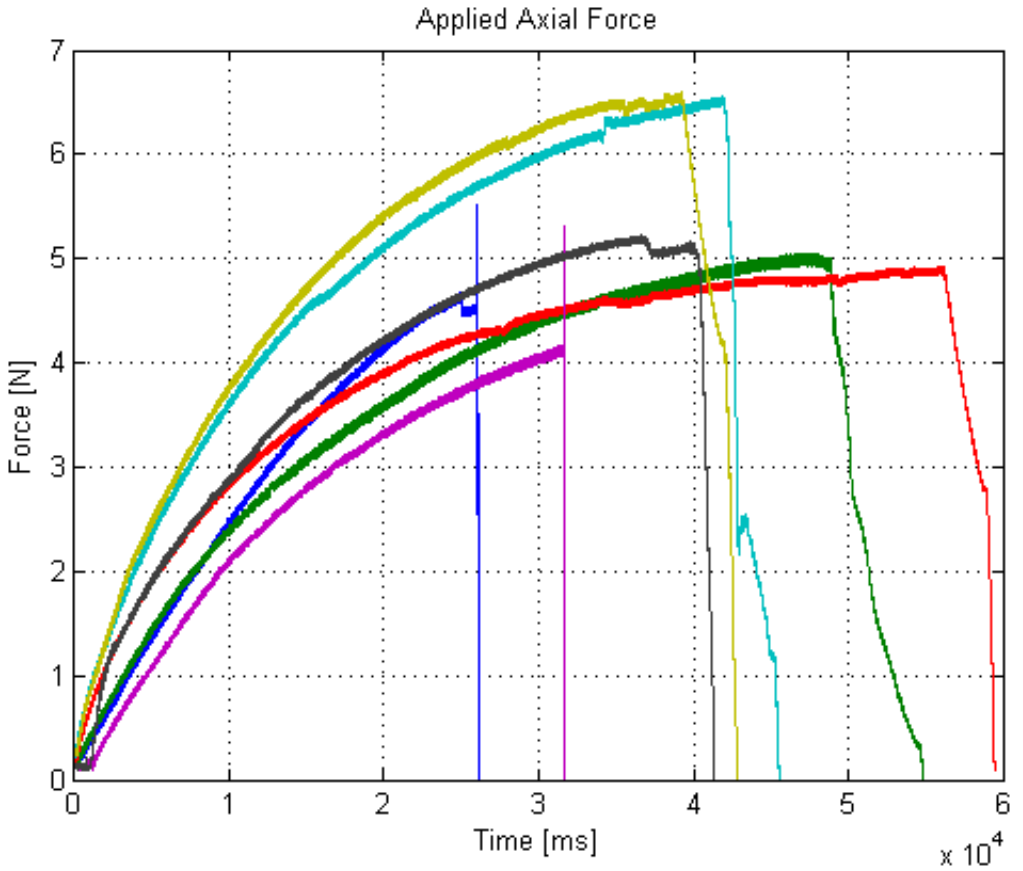

Multiple Inflations for One Balloon
1st inflation
2nd inflation
3rd inflation
4th inflation
5th inflation
Position [cm]
Time [s]
0
20
40
60
80
100
120
140
0
5
10
15
20
25
30
35
40
45

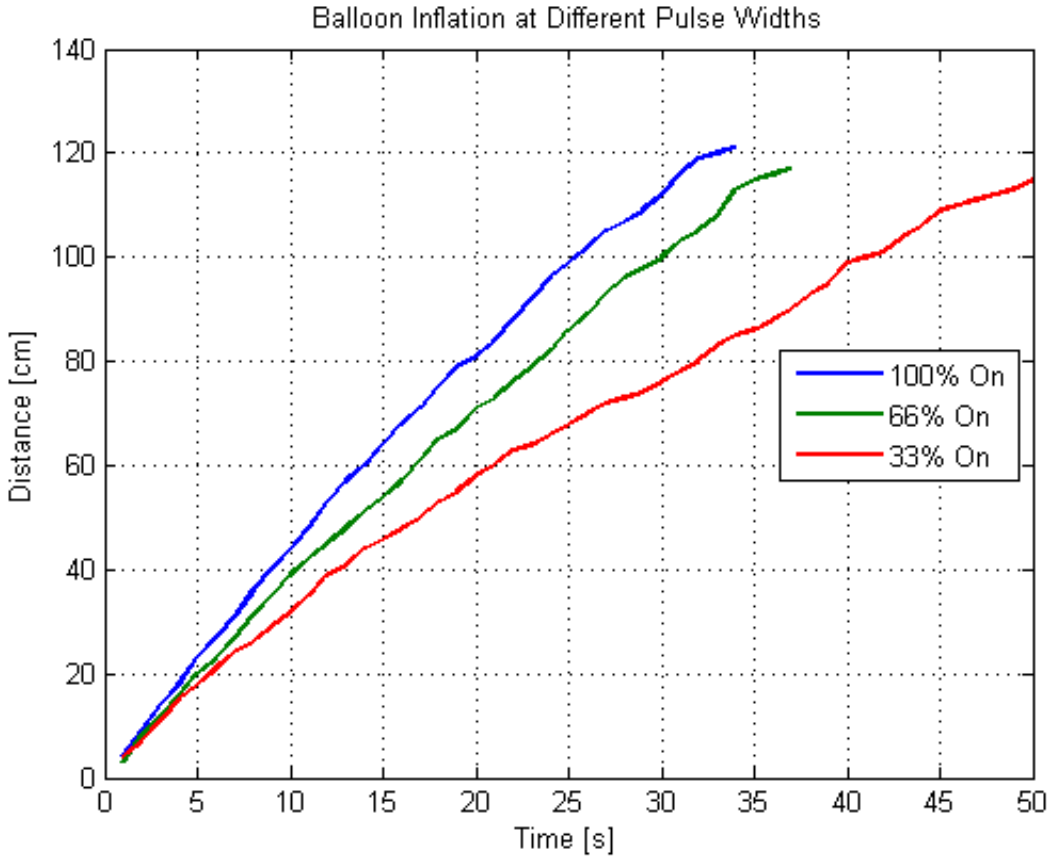

Mean and Standart Deviation of Positions
Mean - Std. Dev. [cm]
Time [s]
0
20
40
60
80
100
120
140
0
5
10
15
20
25
30
35
40
45


3rd Degree Polynomial Fit on Mean Position Values
Position [cm]
Time [s]
140
120
100
80
60
40
20
0
-20
0
5
10
15
20
25
30
35
40
45


Balloon Inflation Speed With 3rd Degree Polynomial Fit
Speed [cm/s]
Time [s]
4.5
4
3.5
3
2.5
2
1.5
0
5
10
15
20
25
30
35
40


Balloon Deflation Speed With 3rd Degree Polynomial Fit
Speed [cm/s]
-15
-16
-17
-18
-19
-20
-21
-22
-23
0
5
10
15
20
25
30
35
Time [s]


Applied Axial Force
Force [N]
7
6
5
4
3
2
1
0
0
1
2
3
4
5
6
Time [ms]
x 10^4



Published with MATLAB® R2014a