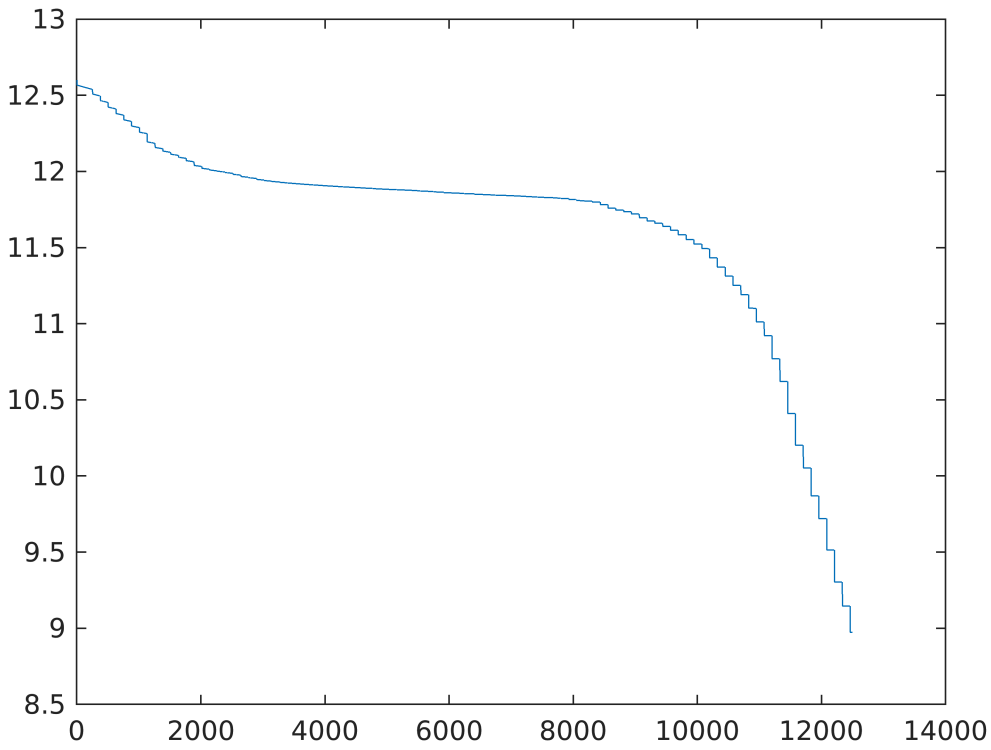
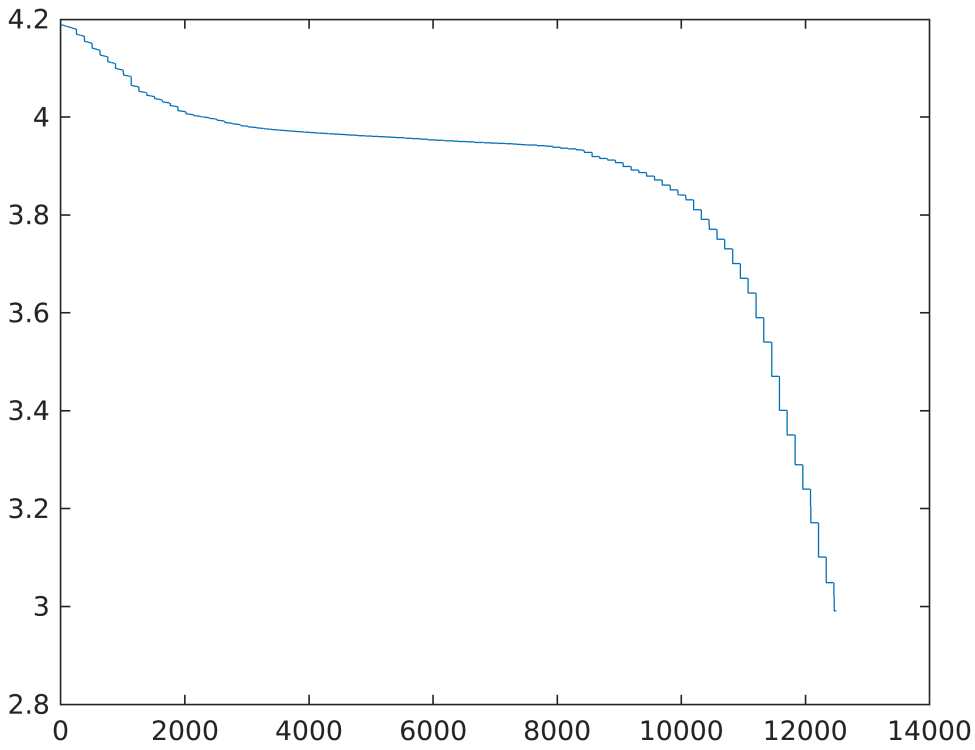
```
load('batteryModelSfun.mat');
load('soc_ocv_new.mat');

%simOut = sim('batteryMine');

%current = simOut.current.Data;
%voltage = simOut.voltage.Data;
%time    = simOut.tout;

load('current.mat');
load('voltage.mat');
load('time.mat');

plot(voltage);
```




```
dt = .01;

z   = battery.z;
ir  = battery.Ir;
h   = battery.h;
```

```
n   = battery.n;
Q   = battery.Q;
g   = battery.g;
M0  = battery.M0;
M   = battery.M;
R0  = battery.R0;
R1  = battery.R1;
C1  = battery.C1;
EOD = battery.EOD;

Imat = exp(-dt/(R1*C1));
Hmat = exp(-abs((n*g*dt)/Q));
```

```
x = [z ; ir ; h];

A = [1 0 0 ; 0 Imat 0 ; 0 0 Hmat];

B = [-n*dt/Q 0 ; (1-Imat) 0 ; 0 (Hmat-1)];

C = [getOCV(z, soc_ocv) -R1 M];

D = [-R0 M0];

y   = zeros(length(voltage), 1);

for k = 1:length(time)
    u = [current(k) ; sign(current(k))];

    x_dot = A*x + B*u;

    C = [getOCV(x_dot(1), soc_ocv)/x_dot(1) -R1 M];

    y(k) = C*x_dot + D*u;

    x = x_dot;
end
plot(y);
```




```
function ocv = getOCV(soc, soc_ocv)

    soc = (soc + 31.8816) /(1.0 + 31.8816);
    if soc < 0.0
        soc = 0;
    elseif soc > 1.0
        soc = 1.0;
    end

    idx = ceil(soc*100) +1;

    if idx > 101
        idx = 101;
    elseif idx < 1
        idx = 1;
    end

    ocv = soc_ocv(idx);
end
```