Table of Contents

Battery equivalemt circuit model and simulation

```
clear all
clc
```

```
% *Discharge current is positive direction*
```

States definitions

```
syms v_batt i_in C_sei C_dl C_w R_dl R_sei R_0 L_0 v_ocv
syms i_batt(t) v_sei(t) v_dl(t) v_w(t) % states

syms i_sei i_oth i_w i_dl% temp variables
```

```
% Loop 1
% eqn1 = v_ocv == L_0*diff(i_batt(t),t) + i_batt*R_0 + v_sei + v_batt
eqn1 = L_0*diff(i_batt(t),t) == v_ocv - i_batt*R_0 - v_sei - v_batt
```

eqn1(t) =

*L_0*diff(i_batt(t), t) == v_ocv - v_batt - v_sei(t) - R_0*i_batt(t)*

Loop 2

```
% eqn2 = v_ocv == L_0*diff(i_batt(t),t) +  R_0*i_batt + i_oth*R_sei +
 v_dl
% eqn2 = v_ocv == v_ocv - i_batt*R_0 - v_sei - v_batt +  R_0*i_batt +
 i_oth*R_sei + v_dl
% eqn2 = 0 == - v_sei - v_batt + i_oth*R_sei + v_dl
% eqn2a = i_batt == i_sei + i_oth
% eqn2a = i_oth == i_batt - i_sei
% eqn2a = i_oth == i_batt - C_sei*diff(v_sei,t)
% eqn2 = 0 == - i_batt*R_0 - v_sei - v_batt +  R_0*i_batt + (i_batt -
 C_sei*diff(v_sei,t))*R_sei + v_dl
% eqn2 = (C_sei*diff(v_sei,t))*R_sei - i_batt*R_sei == - v_sei -
 v_batt + v_dl
% eqn2 = (C_sei*diff(v_sei,t))*R_sei == i_batt*R_sei - v_sei - v_batt
 + v_dl
eqn2 = diff(v_sei,t) == (i_batt*R_sei - v_sei - v_batt + v_dl)*(1/
(C_sei*R_sei))
```

```
%
% Loop 3
% eqn3a = i_oth*R_sei + v_dl == v_sei
% eqn3a = i_oth == (v_sei - v_dl)/R_sei
% eqn3b = i_oth*R_sei + i_w*R_dl + v_w == v_sei
% eqn3b = i_w*R_dl + v_w == v_dl
% eqn3b = i_w == (v_dl - v_w)/R_dl
% eqn3 = C_w*diff(v_w,t) == i_w
eqn3 = diff(v_w,t) == (v_dl - v_w)*1/(R_dl*C_w);
```

```
% Loop 4
% eqn4 = i_dl + i_w == i_oth
% eqn4 = i_dl + i_w == (v_sei - v_dl)/R_sei
% eqn4 = i_dl + (v_dl - v_w)/R_dl == (v_sei - v_dl)/R_sei
% eqn4 = i_dl == (v_sei - v_dl)/R_sei - (v_dl - v_w)/R_dl
% eqn4 = i_dl == v_sei/R_sei - v_dl/R_sei - v_dl/R_dl + v_w/R_dl
% eqn4 = C_dl*diff(v_dl,t) == v_sei/R_sei - v_dl*(1/R_sei + 1/R_dl) +
 v_w/R_dl
eqn4 = diff(v_dl,t) == (v_sei/R_sei - v_dl*(1/R_sei + 1/R_dl) + v_w/
R_dl)*(1/C_dl)
```

eqn2(t) =

*diff(v_sei(t), t) == -(v_batt - v_dl(t) + v_sei(t) - R_sei*i_batt(t))/
(C_sei*R_sei)*

```
eqn4(t) =

diff(v_dl(t), t) == (v_w(t)/R_dl - v_dl(t)*(1/R_dl + 1/R_sei) +
 v_sei(t)/R_sei)/C_dl
```

State Matrix

```
% eqn = diff([i_batt(t); v_sei; v_w; v_dl],t) == ...
%     [(1/L_0)*(v_ocv - i_batt*R_0 - v_sei - v_batt); (1/
(C_sei*R_sei))*(i_batt*R_sei - v_sei - v_batt + v_dl); ...
%     (v_dl - v_w)*1/(R_dl*C_w); (v_sei/R_sei - v_dl*(1/R_sei + 1/
R_dl) + v_w/R_dl)*(1/C_dl)]

X = [i_batt; v_sei; v_w; v_dl]
X_dot = diff([i_batt; v_sei; v_w; v_dl],t)
U = v_batt
A = [(1/L_0)*(v_ocv + ([- i_batt*R_0, -v_sei, -v_ocv/v_dl*v_dl, -
v_ocv/v_w*v_w]));...
    (1/(C_sei*R_sei))*([i_batt*R_sei, -v_sei, +v_dl, 0*v_w]); [0, 0,
 1/(R_dl*C_w), -1/(R_dl*C_w)]; ...
    (1/C_dl)*[0, v_sei/R_sei, -v_dl*(1/R_sei + 1/R_dl), +v_w/R_dl]]
B = [(-1/L_0)*v_batt; (-1/(C_sei*R_sei))*v_batt; 0; 0]

% eqn = X_dot == A*X + B*U


X(t) =

i_batt(t)
 v_sei(t)
   v_w(t)
  v_dl(t)


X_dot(t) =

diff(i_batt(t), t)
 diff(v_sei(t), t)
   diff(v_w(t), t)
  diff(v_dl(t), t)


U =

v_batt


A(t) =
```

```
[(v_ocv - R_0*i_batt(t))/L_0,  (v_ocv - v_sei(t))/L_0,
                  0,                  0]
[            i_batt(t)/C_sei, -v_sei(t)/(C_sei*R_sei),
 v_dl(t)/(C_sei*R_sei),                  0]
[                          0,                         0,
       1/(C_w*R_dl),      -1/(C_w*R_dl)]
[                          0,   v_sei(t)/(C_dl*R_sei), -(v_dl(t)*(1/
R_dl + 1/R_sei))/C_dl, v_w(t)/(C_dl*R_dl)]


B =

          -v_batt/L_0
-v_batt/(C_sei*R_sei)
                    0
                    0
```

Published with MATLAB® R2021a