ELEC 4700 Diode Paramater Extraction

Tom Smy

March 10th, 2022. Due March 13th @ midnight.

Goal In this PA you will investigate the use of non-linear curve fitting for extracting paramaters.

The device we wish to model is a diode including breakdown and a parasitic parallel leakage resistance. The expected physical behaviour is:

$$I = \underbrace{I_s(e^{\frac{1.2}{0.025}V} - 1)}_{\text{Ideal Diode}} + \underbrace{G_p V}_{\text{parallel resistor}} \underbrace{- I_b\left(e^{-\frac{1.2}{0.025}(V+V_b)} - 1\right)}_{\text{breakdown}} \quad (1)$$

with the following paramaters to be extracted,

I_s**:** Forward bias saturation current.

I_b**:** Breakdown saturation current.

V_b**:** Breakdown volatage.

G_p**:** Parasitic parallel conductance.

You will be using 3 methods of fitting curves to diode data:

1. Polynomial fitting using *polyfit()* and *polyval()*.
2. Non-linear curve fitting using *fit()*
3. Neural net fitting using the matlab Neural Net toolbox.

Tasks

1. Generate some data for $I_s = 0.01$pA, $I_b = 0.1$pA, $V_b = 1.3$ V and $G_p = 0.1\ \Omega^{-1}$.
   - Create a V vector from -1.95 to 0.7 volts with 200 steps.
   - Create an I vector
   - Create a second I vector with 20% random variation in the current to represent experimental noise.
   - Plot the data using *plot()* and *semilogy()*
2. Polynomial fitting
   - Create a 4th order and 8th order polynomial fit for the two data vectors.
   - Add them to your graphs of the data.
   - Draw some conclusions!
3. Nonlinear curve fitting to a physical model using *fit()*
   - Do *help fit*.
   - To fit data using *fit()* you can pass it a string containing the non-linear function to fit such as:
     fo = fittype('A.(exp(1.2*x/25e-3)-1) + B.*x - C*(exp(1.2*(-(x+D))/25e-3)-1)');*
     where A, B, C and D are parameters to be extracted.
     You can then extract the parameters using:
     ff = fit(V,I,fo)
     A curve can then be generated using,
     If = ff(x)
   - Create three different fitted curves using:
     (a) Only two fitted paramaters A and C by explictly setting B and D to the values used in equation 1 to generate the data.

(b) Using three fitted paramaters A, B and C by explictly setting D to the value used in equation 1 to generate the data.

(c) Fitting all four paramaters A, B, C and D.

- Compare to the original data and draw some conclusions!

4. Fitting using the Neural Net model.

   To use the Neural Net Toolbox you should use this code:

   ```
   inputs = V.';
   targets = I.';
   hiddenLayerSize = 10;
   net = fitnet(hiddenLayerSize);
   net.divideParam.trainRatio = 70/100;
   net.divideParam.valRatio = 15/100;
   net.divideParam.testRatio = 15/100;
   [net,tr] = train(net,inputs,targets);
   outputs = net(inputs);
   errors = gsubtract(outputs,targets);
   performance = perform(net,targets,outputs)
   view(net)
   Inn = outputs
   ```

5. Read about *fitnet()* and associated functions
6. Run the code
7. Compare to the original data and draw some conclusions!

Checkout When you are finished:

1. Create a new repo on your github account called CMPA
2. Clone the repo to your machine
3. Add your code to the repo, commit, and push it back to github
4. Check that it worked, if it did, you're all set