Non Linear MOR using PWL models

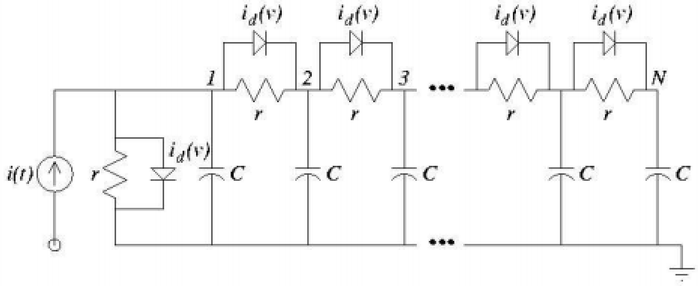

Figure 1: The Non-Linear system under study

The circuit contains 10 stages and non-linearity is introduced because of the presence of diodes. The diode equation used was

$$i = e^{10V}$$

The resulting state space model for the system is

$$\dot{v} = \begin{pmatrix} -2 & 1 & 0 & . & . & 0 \\ 1 & -2 & 1 & 0 & . & 0 \\ 0 & 1 & -2 & 1 & 0 & . \\ . & . & . & . & . & . \\ 0 & . & . & 1 & -2 & 1 \\ 0 & . & . & . & 1 & -1 \end{pmatrix} v + \begin{pmatrix} 2 - e^{10v_1} - e^{10(v_1 - v_2)} \\ e^{10(v_1 - v_2)} - e^{10(v_2 - v_3)} \\ . \\ . \\ . \\ e^{10(v_9 - v_{10})} - 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 0 \\ . \\ . \\ 0 \end{pmatrix} i$$

$$y = \begin{pmatrix} 0 & 0 & . & . & . & 1 \end{pmatrix} v$$

where v is the state space vector. The equations above dont have the familiar format seen in linear systems. We can treat the system matrix as a set of 10 functions acting upon 10 variables and producing 10 variables. This helps us define the Jacobian of the system matrix as

$$Jacobian(f(v_1, v_2, ...v_{10})) = \begin{pmatrix} \frac{\partial f_1}{\partial v_1} & \frac{\partial f_1}{\partial v_2} & . & . & . & \frac{\partial f_1}{\partial v_{10}} \\ \frac{\partial f_2}{\partial v_1} & \frac{\partial f_2}{\partial v_2} & . & . & . & \frac{\partial f_2}{\partial v_{10}} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ \frac{\partial f_{10}}{\partial v_1} & \frac{\partial f_{10}}{\partial v_2} & . & . & . & \frac{\partial f_{10}}{\partial v_{10}} \end{pmatrix}$$

Using the Taylor expansion we can approximate any matrix of non-linear functions at a point. The first order approximation is

$$f(v_1, v_2 ...., v_{10}) = f(V_0) + J_{V_0}(v - V_0)$$

where $V_0$is the point at which we approximate. The non-linear system can be approximated with a linear model expanded at some point in the state space, generally in the trajectory of the response. But using only one linearization point cannot work effectively for strongly non-linear models.

0.1 One Linearization Point

I chose the initial state-space vector for linearising the model. The following becomes the new state space equation

$$\dot{v} = f(V_0) + J_f v - JV_0 + Bu$$

I use the constant term $f(V_0) - JV_0$as a offset in the input. B being a rank 1 matrix the pseudo inverse of the constant term gives the offset with an error approx 10^{-4}. Hence the system becomes

$$\dot{v} = J_f v + B(u + offset)$$

which can now be solved using regular methods. The error is large and the linearization fails to give a good approximation.

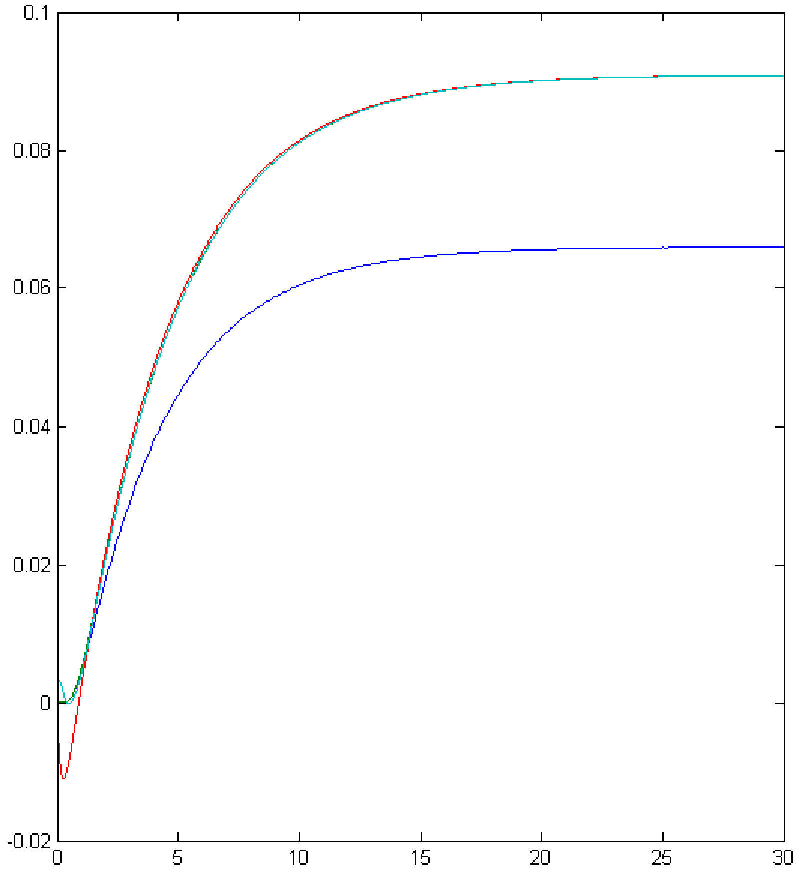

Figure 2: Comparing Linear and non-linear

The green line depicts the solution from Linearized model and the blue one is from actual non-linear system.

0.2 Piece wise Linearization

In this case the points chosen for linearisation belong to the trajectory followed in the state-space. We linearise the model at each of these points individually and use a linear combination of these models to define the system. The weightage given to a particular model depends upon the proximity of the state to the linearisation points.

- **Choosing the points :** Involves running the non-linear system for a test input in this case a unit step response and then choosing points which belong to the trajectory of the path followed.
- **Weightage procedure:** I used an exponential weightage function as mentioned in the paper.

$$w_i = e^{-40\frac{(v-V_i)^2}{\Sigma(v-V_i)^2}}$$

  The weightage given to a linear model depends on the proximity to the linearization point, since linearization works only in a small vicinity of the linearization point, only the model closest will be given priority.

Hence the new state space equation will be

$$\dot{v} = \sum w_i (J_{V_i} * v_i) + B \left(u + \sum w_i * (offset)_i\right)$$

I have chosen 5 points along the state-space path, so giving me 5 linear models. Solving this equation gives a very good approximation to the non-linear model.

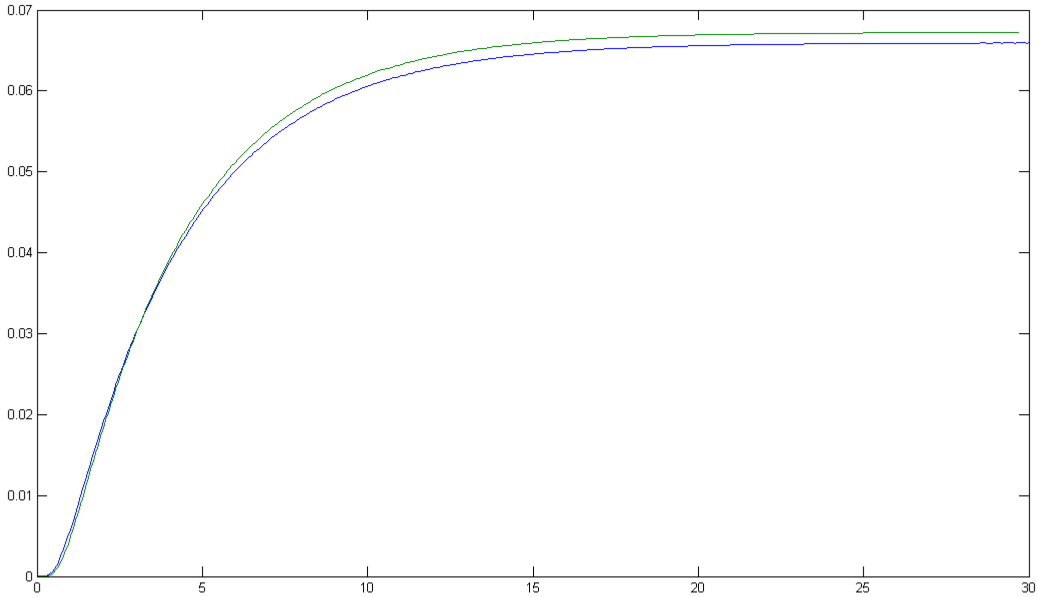

Figure 3: PWL vs Non-Linear

The blue line being the non-linear solution and the green being the PWL solution.

0.3 Reduction using Krylov subspaces and Moment Matching

Since in the wieghing procedure involves distance calculation it is necessary that the linearization points and the state-space trajectory of the reduced model lie in the same subspace. At each point we need to create the l-dimensional subspace to match l-moments.

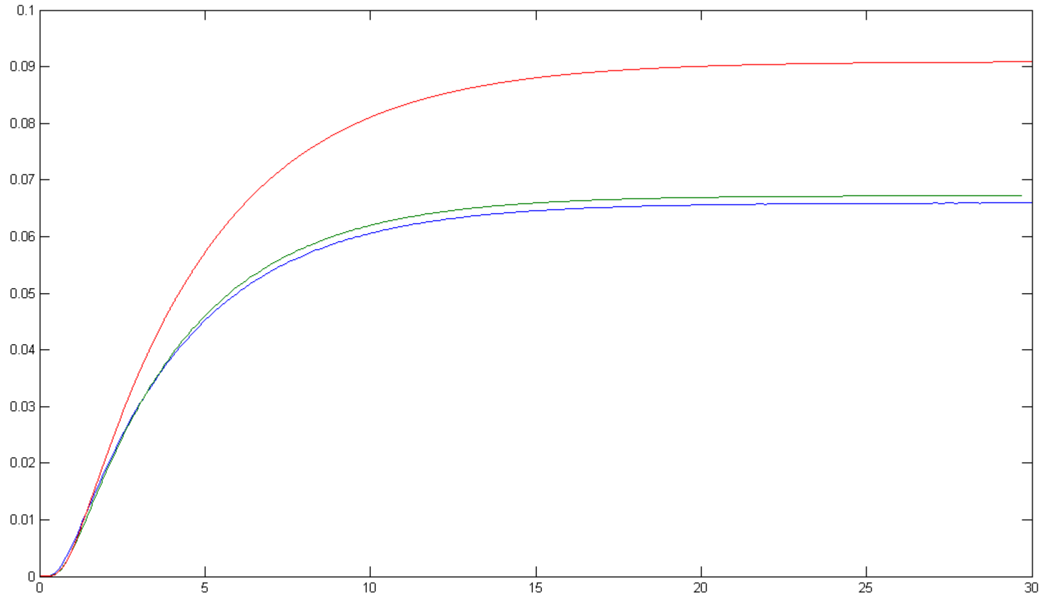

Figure 5: Comparision between solutions obtained from 1point linearisation, Non-linear and PWL system models

Since the space created at each point is different and we need to accomodate moment matching at all points, we create a space formed by the union of the Krylov spaces at each point and the linearization points.

$$V_{aggregate} = [K(V_1)......K(V_5)\,V_1\,V_2....V_5]$$

Having this space helps us match moments at all the points of consideration. Now we need to eliminate redundant vectors and generate a basis for this space. For which SVD is used and we eliminate the eigenvectors with a small eigenvalue. In this case I was left with 3-D space.

The new equation for reduced space becomes

$$\dot{v} = V^T * \sum w_i\,(J_{V_i} * V * z_i) + V^T * B\left(u + \sum w_i * (offset)_i\right)$$

The following is the solution obtained by solving a 3D reduced order model.

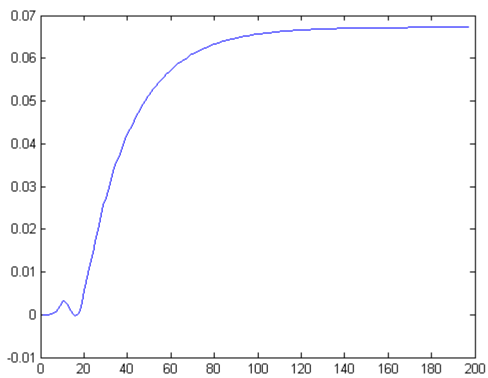

Figure 4: Solution obtained from PWL-Reduced order model