Simulating the Dynamic Response of Current Carrying MEMS Devices

Mohamad H. Alizade

Supervisor: Dr. Haeri

[1] Base on the paper: Dynamic pull-in for micro-electromechanical device with a current-carrying conductor. Ji-Huan He, Daulet Nurakhmetov et al. (2019)

Problem Definition

The dynamical lumped parameter model arising from current carrying, linear elastic MEMS devices is non-linear. This is due to inverse proportional force in the magnetic force of two wires.

In this short note, we proceed as follow:

i. Present the lumped mechanical model of the system
ii. Drive the steady state equations
iii. Linearize the state equations around an operating point
iv. Compare the linear and non-linear' s pulse response
v. Specifics of the simulations
vi. Yield the second order perturbation correction for the linear model and compare it to the linear response

Lumped Mechanical Model

Two current carrying wires interact according to the Ampere's law. (Eq. 1) Here we consider the motion of a wire with length 'l' and mass 'M' in the magnetic field, restrained by linear elastic springs and dampers. Fig. 1 illustrates the system.

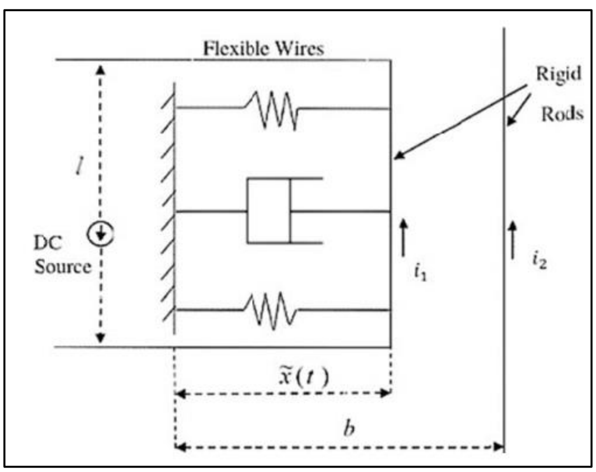

Figure 1 Mems Model

$$\boxed{f = \frac{\mu_0 i_1 i_2}{2\pi r}}$$

Equation 1 Ampere's Law

Where μ_0 is the vacuums' magnetic constant.

Steady-State Model

It's easy to show that the dynamical equation for wire is as follows.

$$\boxed{M\ddot{\tilde{x}} + B\dot{\tilde{x}} + K\tilde{x} - \frac{\mu_0 i_1 i_2 l}{2\pi(b-\tilde{x})} = 0}$$

Equation 2 Model's Dynamics

Where 'B' is the damping ratio and 'K' is the elastic coefficient.

Taking $x = \frac{\tilde{x}}{l}$ and $u = \frac{\mu i_1 i_2 l}{2\pi M b^2}$ we can normalize the system.

$$\boxed{\ddot{x} + \frac{B}{M}\dot{x} + \frac{K}{M}x - \frac{u}{1-x} = 0}$$

Equation 3 Normalized Model

According to the Eq. 3 we define the state space model as follows.

$$\boxed{\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_1 - x_2 + \frac{u}{1-x_1} \end{cases}}$$

Equation 4 State Model

Linearization

Consider the following operating point.

P = (x_1, x_2, u) = (0.2, 0, 0.16)

We proceed to linearize the state Eq. 4 around this operating point. Taking Jacobian's from the equation yields:

$$\dot{\vec{x}} = A\vec{x} + Bu \qquad (5)$$

Where $A = \begin{bmatrix} 0 & 1 \\ -0.75 & -1 \end{bmatrix}$ and $B = \begin{bmatrix} 0 \\ 1.25 \end{bmatrix}$. This result is numerically confirmed by 'main.m' MATLAB file.

Comparing the Pulse Responses

Eigenvalues of matrix 'A' reside on the left-hand side of the imaginary axis. This means any instability is due to the non-linearity. Furthermore, for small inputs we can safely assume that the system is stable.

In Fig. (2:4) linear vs non-linear response of the system for different inputs are compared. The dashed line represents the input ‘u’.

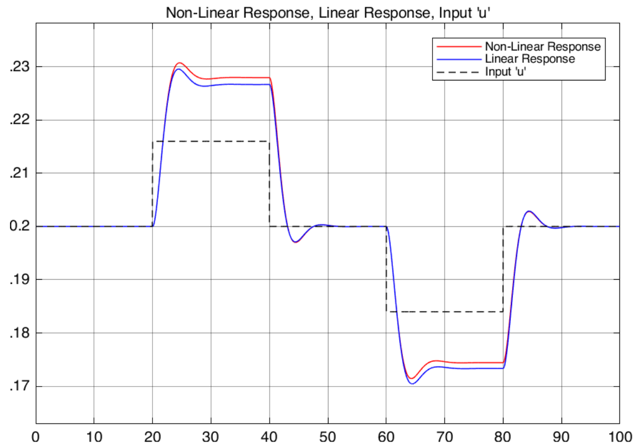

Figure 2 response for 10% change in u

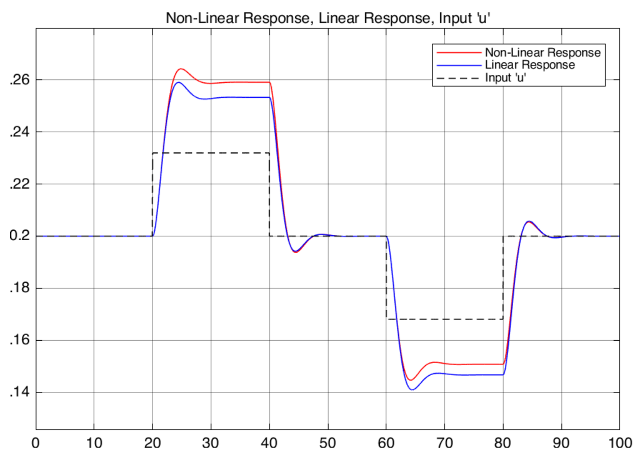

Figure 3 Response for 20% change in u

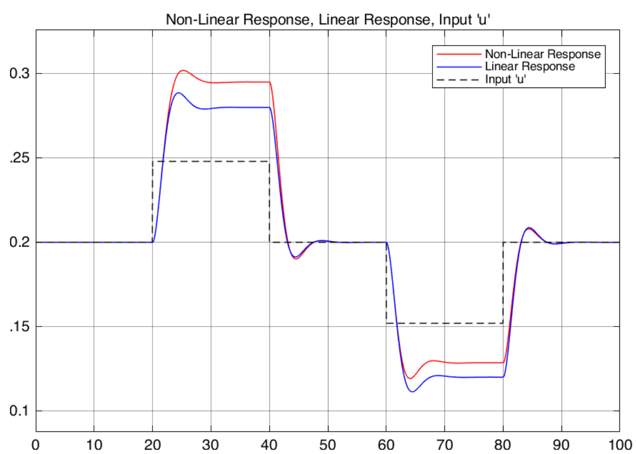

Figure 4 Response for 30% change in u

As we increase the amplitude of the input, linear approximation is less and less accurate. As it can be seen in fig. 4, this results in a noticeable difference.

Specifics of The Simulation

The parameter ‘amp’ in the ‘main.m’ file defines the amplitude of the change in the pulses relative to its operating point. It can be changed to achieve previous results.

To model the non-linear system, we used the following implementation in the Simulink.

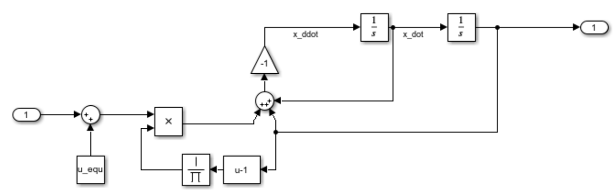

Figure 5 non-linear Simulink implementation

And finally, to compare the linear and non-linear model the following implementation was used.

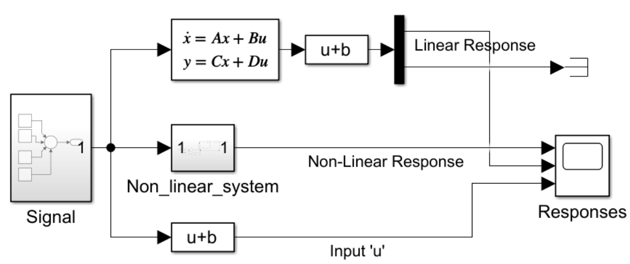

Figure 6 Overall implementation

Where the ‘Non_linear_system’ is the same system as the fig. 5.

Second Order Perturbation*

It’s possible to improve over the linear response by considering the second order terms. In order to achieve this, we let $x = x_{ss} + \beta x^{(1)} + \beta^2 x^{(2)}$ and $u = u_{ss} + \beta u^{(1)}$ where β is a small number and x_j is the jth order perturbation term. Now we apply the Taylor’s series to find its dynamics. (first order behavior remains the same)

$$\beta^2 \dot{x}_1^{(2)} = \beta^2 x_2^{(2)} \qquad (6)$$

$$\beta^2\dot{x}_2^{(2)} = -\beta^2 x_1^{(2)} - \beta^2 x_2^{(2)} + \frac{u_{ss}}{(1-x_{ss})^2}\beta^2 x_1^{(2)} + \frac{\beta^2 x_1^{(1)}}{(1-x_{ss})}\left\{\frac{u_{ss}}{(1-x_{ss})^2}x_1^{(1)} + \frac{1}{(1-x_{ss})}\mathrm{u}^{(1)}\right\} \quad (7)$$

Now We can drop the β coefficient by defining $x^{(2)} \leftarrow \beta^2 x^{(2)}$ and rewriting the second order state equations.

$$\dot{x}_1^{(2)} = x_2^{(2)} \quad (8)$$

$$\dot{x}_2^{(2)} = -x_1^{(2)} - x_2^{(2)} + \frac{u_{ss}}{(1-x_{ss})^2}x_1^{(2)} + \frac{1}{(1-x_{ss})}\left\{\frac{u_{ss}}{(1-x_{ss})^2}x_1^{(1)}.x_1^{(1)} + \frac{x_1^{(1)}.\mathrm{u}^{(1)}}{(1-x_{ss})}\right\} \quad (9)$$

We notice that the dynamical behavior of the second order term is similar to the first order term except the input from the system is based on the first order solution. Let's define $u^{(2)}$ as follows:

$$u^{(2)} = \frac{u_{ss}}{(1-x_{ss})^2}x_1^{(1)}.x_1^{(1)} + \frac{x_1^{(1)}.\mathrm{u}^{(1)}}{(1-x_{ss})} \quad (10)$$

This concludes that,

$$\dot{\vec{x}}^{(2)} = A\vec{x}^{(2)} + Bu^{(2)} \quad (11)$$

Now we'll be able to compare the performance of the second order system with previous results. Fig. (8-10) show the improved accuracy relative to the linearized system. Fig. (7) also shows the Simulink implementation of it.

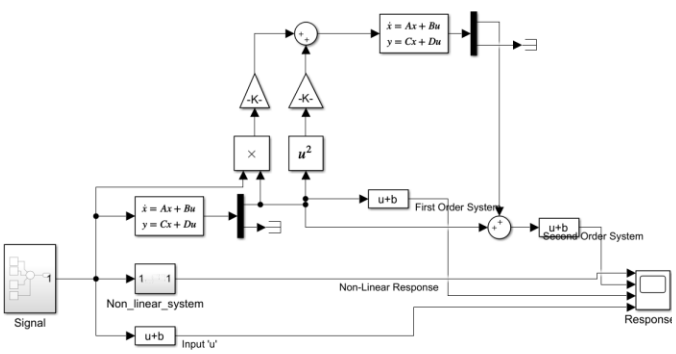

Figure 7 Simulink Implementation of the second order system

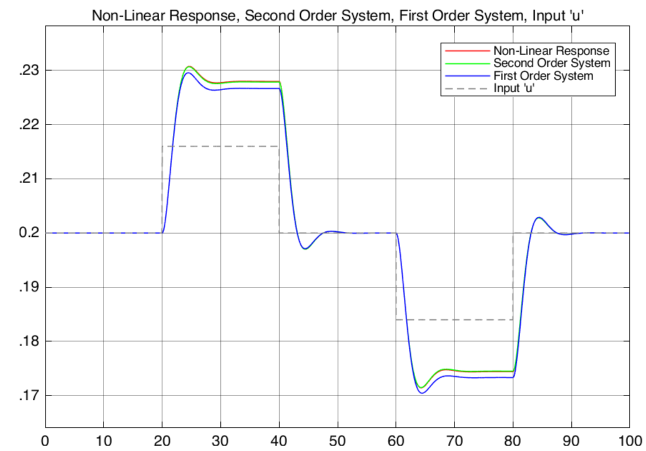

Figure 8 response for 10% change in u

Figure 9 response for 20% change in u

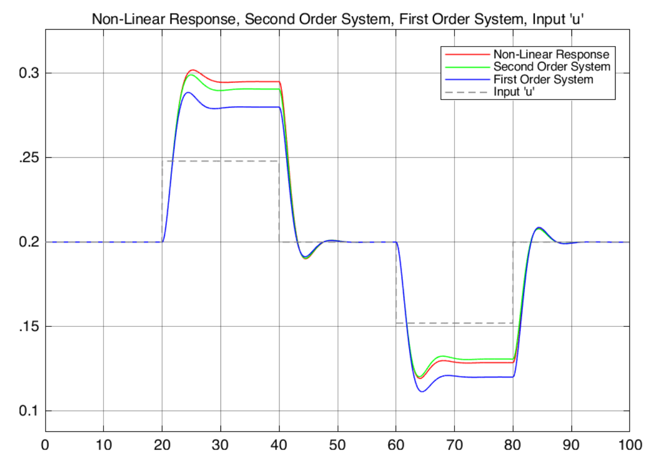

Figure 10 response for 30% change in u