MAE 3260 Final Group Work: Exploring a System of Interest

Report

Outline:

Title: Shake Em Up: Seismic Control for Buildings

Topic of Interest: Implementing an active control system for a multi-story building subject to external forcing

Abstract: We define our system as a multi-story building subject to external forcing such as an earthquake or wind forces. Our goal is to implement an active control system that applies forces on different floors of the building based on the expected building dynamics that act to counteract the forces due to the earthquake. We plan to implement this control system analytically by modifying the governing equation of motion, and with MATLAB to simulate the system.

Students/Roles:

Our group worked together to produce this document. We did not do the 2 pages per person format, as we felt that model did not fit our project. Everyone contributed equally to everything.

Links to Portfolios:

Matthew Sonn
Julie Nam
Shreya Anand
Zawad Dewan

List of MAE 3260 Concepts Used in this Group Work:

Modal Analysis | Vibrations | Feed-Forward Control | Second Order Systems

Introduction and Derivation of the Key Equation of Motion:

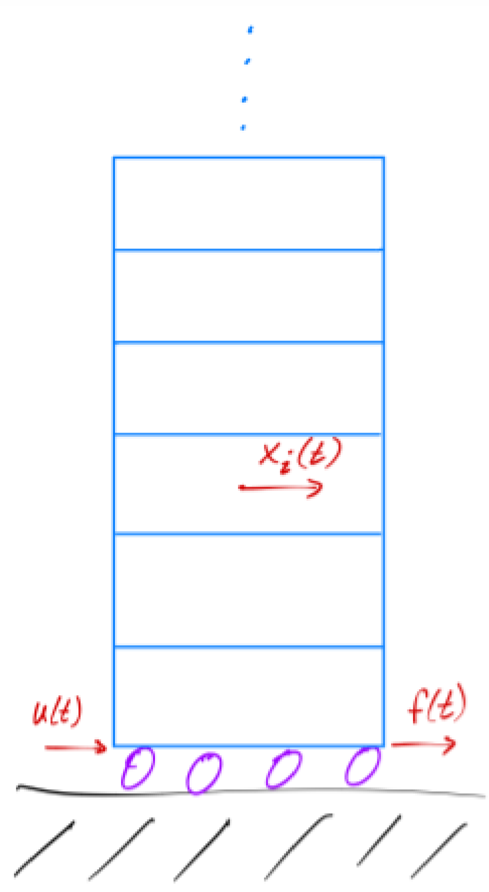

Figure 1: Setup of the dynamics model used in this report

Our project is centered around the modeling of an n-story building experiencing an external disturbance from wind or an earthquake. The model of the building consists of a building on rollers with each story having a displacement of $x_i(t)$. The wind or the earthquake force is represented by f(t). The goal of this project is to develop a sound dynamical model of the system, and then introduce an active, feed-forward control system, u(t), that could be used to reduce the amplitudes of the building's acceleration or position. The general model of the building is shown to the left in Figure 1. We model each floor as being attached to adjacent floors with both a spring of some spring constant k_i, and a damper of some damping constant, b_i. Therefore, as we develop the dynamics model, matrices must be used to represent the damping and spring contributions to the forces. We will now derive the building's equation of motion and the mass, stiffness, and damping matrices, ignoring the control force for now. Looking at floor i, we can use Newton's Second Law to sum up the forces that are acting on just that floor, which are dependent upon the springs and masses, and the positions of floor i relative to both floor i+1 and floor i-1:

$$m_i\ddot{x}_i = -k_i(x_i - x_{i-1}) - k_{i+1}(x_i - x_{i+1}) - b_i(\dot{x}_i - \dot{x}_{i-1}) - b_{i+1}(\dot{x}_i - \dot{x}_{i+1}) + f_i(t)$$

Collecting the coefficients of the acceleration, velocity, and position terms:

$$m_i\ddot{x}_i + (b_i + b_{i+1})\dot{x}_i + (k_i + k_{i+1})x_i - b_i\dot{x}_{i-1} - b_{i+1}\dot{x}_{i+1} - k_i x_{i-1} - k_{i+1}x_{i+1} = f_i(t)$$

If we do this for every floor, we will see similar patterns, and thus can begin to build the mass, stiffness, and damping matrices. We define x(t) and f(t) as an nx1 vector, with each component representing the displacement and force respectively of floor i, and we define the mass, stiffness, and damping matrices as the following:

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m_n \end{bmatrix} \qquad \mathbf{K} = \begin{bmatrix} k_1 + k_2 & -k_2 & \cdots & 0 \\ -k_2 & k_2 + k_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & k_{n-1} + k_n \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} b_1 + b_2 & -b_2 & \cdots & 0 \\ -b_2 & b_2 + b_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_{n-1} + b_n \end{bmatrix}$$

Putting everything together, we can see that our final equation of motion for an n-story building subject to a force on each floor is:

$$\mathbf{M}\,\ddot{\mathbf{x}}(t) \;+\; \mathbf{B}\,\dot{\mathbf{x}}(t) \;+\; \mathbf{K}\,\mathbf{x}(t) \;=\; \mathbf{f}(t)$$

Where **M** is the nxn mass matrix, **K** is the nxn stiffness matrix, and **B** is the nxn damping matrix. **x**(t) is an nx1 vector, and **f**(t) is also an nx1 vector.

Discussion of the Equation of Motion:

There's some key things to note about this ODE. First is the fact that it is second order, meaning solutions will oscillate in time depending on the forcing function, **f**(t), and the damping ratio. We have solved this equation for a few different solutions like a step or impulse free response, and a sinusoidal input. We have also experimentally and analytically determined methods for finding the building's natural oscillation modes with these dynamics. Experimentally, we can perform a sine sweep, sweeping through frequencies from low to high, and measure the accelerations of each floor using accelerometers. We will obtain a Bode Plot that gives information on both the natural modes of oscillation, and the shapes of these modes. For an n story building, we will see n modes (as long as the problem remains one dimensional).

Analytically, we know that solving for the natural frequencies of each mode is an eigenvalue problem that results in a solution for the free response of the system (free response meaning **f**(t)=0 with nonzero initial conditions). This problem is solved using the ansatz $x(t) = C\phi e^{\lambda t}$, where C is a constant solved for from the initial conditions, phi is an eigenvector for the system representing the shape of the mode, and lambda is the eigenvalue, or temporal solution for the mode which can be related to the natural frequency of the particular mode using $\lambda_i = -\omega_{ni}^2$. From the temporal and spatial solutions, we can form the free response of the system, which is simply a superposition of all the modes: $x(t) = \sum_{i=1}^{n} \phi_i[Asin(\omega_{ni}t) + Bcos(\omega_{ni}t)]$, which is a solution for the free response of the system (using the ansatz and ODE).

The natural frequencies of each mode are useful, as we want to ensure the building will never be driven at these frequencies. When a system is driven at or near its natural frequency, the amplitude of the system's oscillations will drastically increase in a phenomenon called resonance, which is not optimal for a building subject to external forces. We can further see the effect of resonance on a Bode plot of an n-story building. At the natural frequencies, the frequency response plot shows a very high amplitude ratio. While in real life, we would never see a building shake 10^5 m away from its resting state, these oscillations would nevertheless result in catastrophic failure.

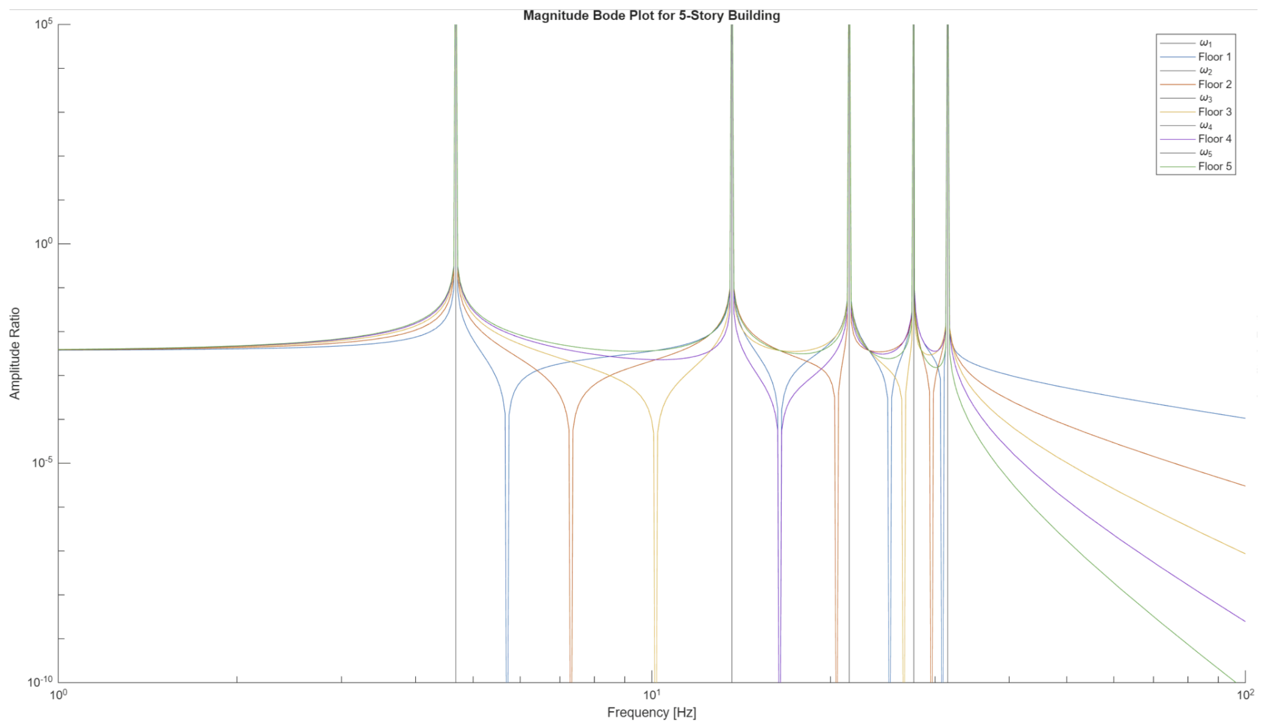

Figure 2: Amplitude Ratio Bode Plot for a 5-story building. Note the large peaks at the locations denoted by ω_i, which correspond to the natural frequencies of the system. These were computed using MATLAB's `eig` function.

If the building does end up experiencing driving frequencies from earthquakes or wind patterns that match the natural frequencies of the building's modes, we can implement an active control system that acts to reduce the amplitude of these oscillations.

Active Control System:

Having a passive system is not sufficient to control building stability since it is important to adapt to different earthquake frequencies. Thus, an active system utilizing both feedback and feedforward control is used to mitigate earthquake vibration effects. The feedforward control system uses an accelerometer to measure the ground acceleration, and this data is sent to a controller to then send signals to actuators before the building reacts. The actuator then applies a control force to counteract the motion of the ground so that the building moves less. This is useful for disturbances that can be easily predicted or measured but does not account for unexpected ones. Therefore, a feedback system is also used to measure the building's displacement/velocity and make corrections to any errors.

A basic theoretical framework of the control system will now be explained. The equation of motion for an earthquake excited N-story building with ground acceleration $\ddot{x}_g(t)$ and active control forces: $M\ddot{x}(t) + B\dot{x}(t) + Kx(t) = Hu(t) - Mr\ddot{x}_g(t)$ where **M** is the nxn mass matrix, **B** is the nxn damping matrix, **K** is the nxn stiffness matrix (same as described in the equation of motion for uncontrolled system), **x**(t) is an nx1 vector of the floor displacements relative to the ground, **u**(t) is the mx1 vector of control forces (m represents number of actuators, **H** is the nxm matrix defining where actuators are placed, and **r** is the influence vector that ensures the ground accelerations affects each floor of the building, which means that it is a nx1 matrix of 1's (Abdulateef).

In order to better understand the control system, this can be presented in the state space form. State vector **z** will be defined as $\begin{bmatrix} x \\ \dot{x} \end{bmatrix}$. The state space of the above model is $\dot{z} = Az + Bf + E\ddot{x}_g$, where $A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}B \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ M^{-1}H \end{bmatrix}$, and $E = \begin{bmatrix} 0 \\ -r \end{bmatrix}$ (Tai). These state space equations are derived by rearranging the ODE of the building with the ground acceleration and control forces.

The total control law equation is expressed by $u(t) =- K_{fb}z - K_{ff}\ddot{x}_g$ where K_{fb} is the feedback gain while K_{ff} is the feedforward gain. Both the feedback and feedforward gains can be determined by an optimal control theory such as the Linear Quadratic Regulator and includes both displacement and velocity feedback. The LQR algorithm utilizes the Ricatti equation to minimize the quadratic cost function J, which means that it aims to find a gain value that balances cost and performance (Zhang).

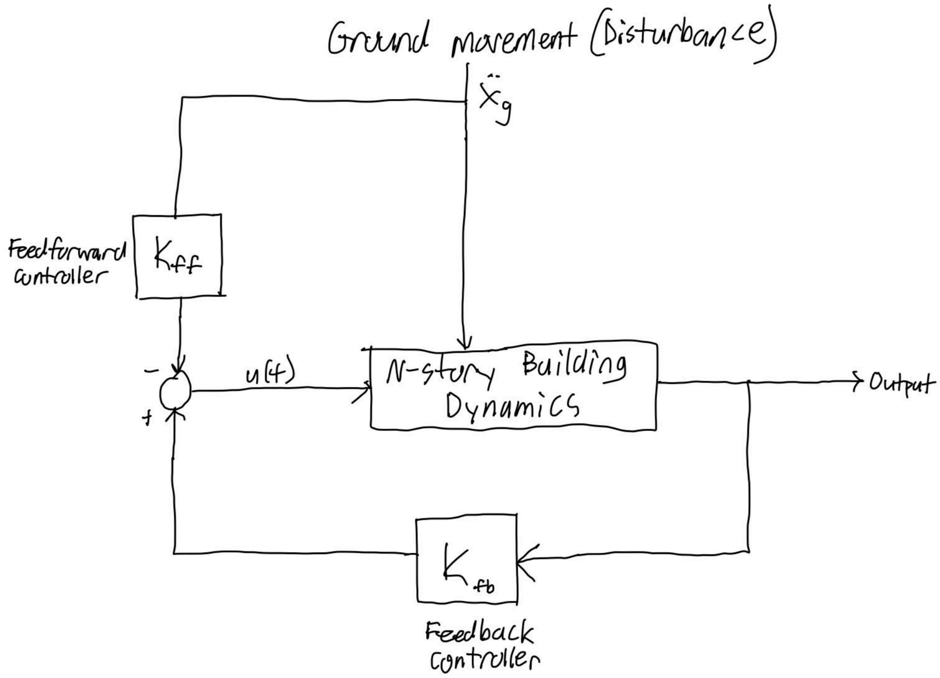

Figure 3: A simplified block diagram of the feedback-feedforward control system for N-story building

Control Implementation in MATLAB:

For our implementation of the control law, we chose to implement feedforward control for a 10-story building with no damping. To simulate seismic activity, we obtained ground motion data from the PEER Ground Motion Database and used the values of acceleration as the ground acceleration term for our model. The equations of motion for our 10-story system were integrated using MATLAB's built-in `ode45` solver. For non-stiff differential equations, `ode45` strikes a good balance between speed and accuracy, which is why we chose to use it for our simulation. For every time step, MATLAB uses the state space model to update the values of displacement on each floor.

```
function deriv = odefun(t, y)
    global M;
    global n;
    global K;
    global H;
    global K_ff;
    x = y(1:n);
    x_dot = y(n+1:end);
    x_ddot = -inv(M)*K*x + inv(M)*H*K_ff*ground_accel(t)*ones(n, 1)...
                - ones(n, 1)*ground_accel(t);
    deriv = [x_dot; x_ddot];
end
```

Figure 5: MATLAB Function that computes the derivative of the state space vector. This is later passed to `ode45`.

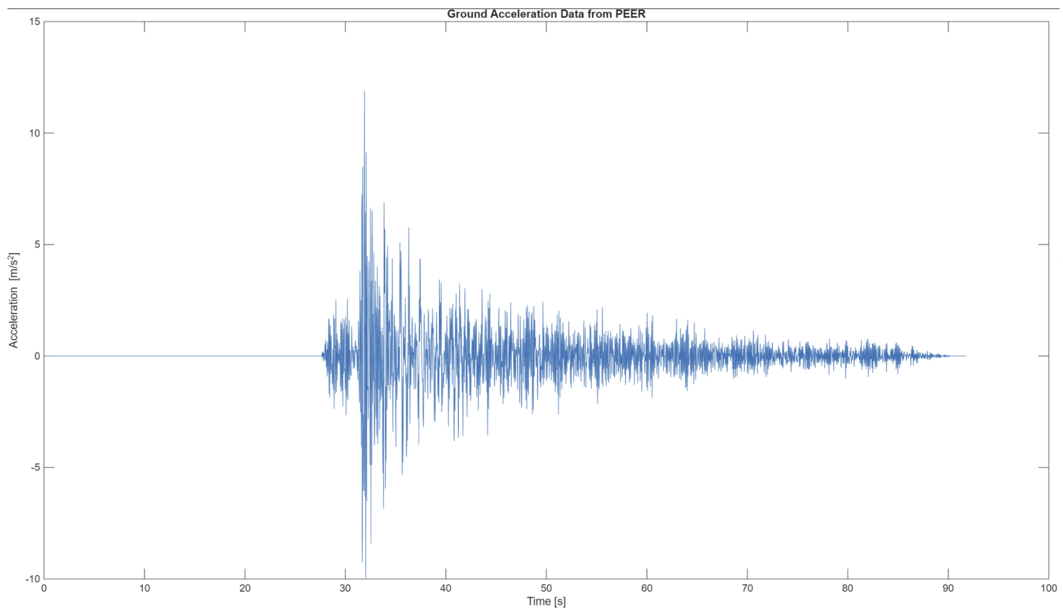

Figure 4: Ground acceleration data from an earthquake in Sparks, OK. Data of this seismic event was obtained from the PEER Ground Motion Database.

The maximum displacement experienced by our 10-story building was 0.335 m under the earthquake loading. For testing feedforward, we limited ourselves to only 3 actuators. In the case of n-actuators for an n-story system, the feedforward case is trivial: the gain K_{ff} is the same as the system mass matrix, as this exactly cancels out the ground acceleration. Moreover, in the real world, we'd expect the development and installation of such a control system to be somewhat costly, further motivating our exploration of a limited number of actuators. The following table shows a list of actuator placements, feedforward gains, and max displacements:

Actuator Placement	Feedforward Gain K_{ff}	Max Displacement
Floors 3, 2, and 8	18555	0.2237 m, on floor 7
Floors 1, 5, and 10	23900	0.1772 m, on floor 10
Floors 1, 2, and 3	10014	0.2860 m, on floor 10
Floors 4, 6, and 8	15507	0.2226 m, on floor 10

Preliminary analyses of a few actuator placements suggest that the more spread out the actuators are, the more favorable the system response.

Visualization via Matlab

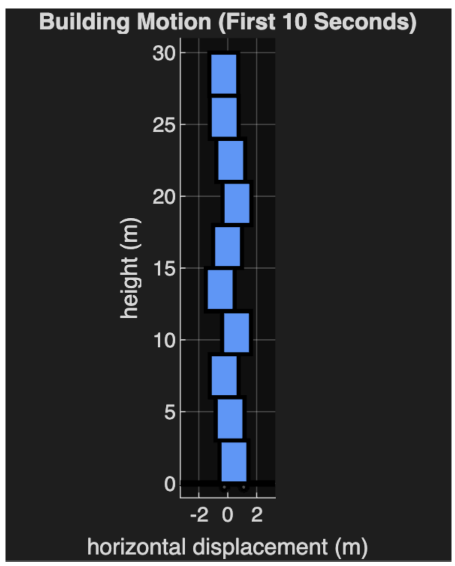

Figure 5: A floor-by-floor visualization of a simulated earthquake response for a 10 story building with feedforward control on floors 1, 5, and 10.

We designed this matlab script to take in the position data per floor per time as well as the times that data was recorded and visualize what the building, roughly, would look like during such a response over time. As we can see, the displacement at the peak of the oscillation for the floors at which the control is implemented is much less than other floors surrounding it.

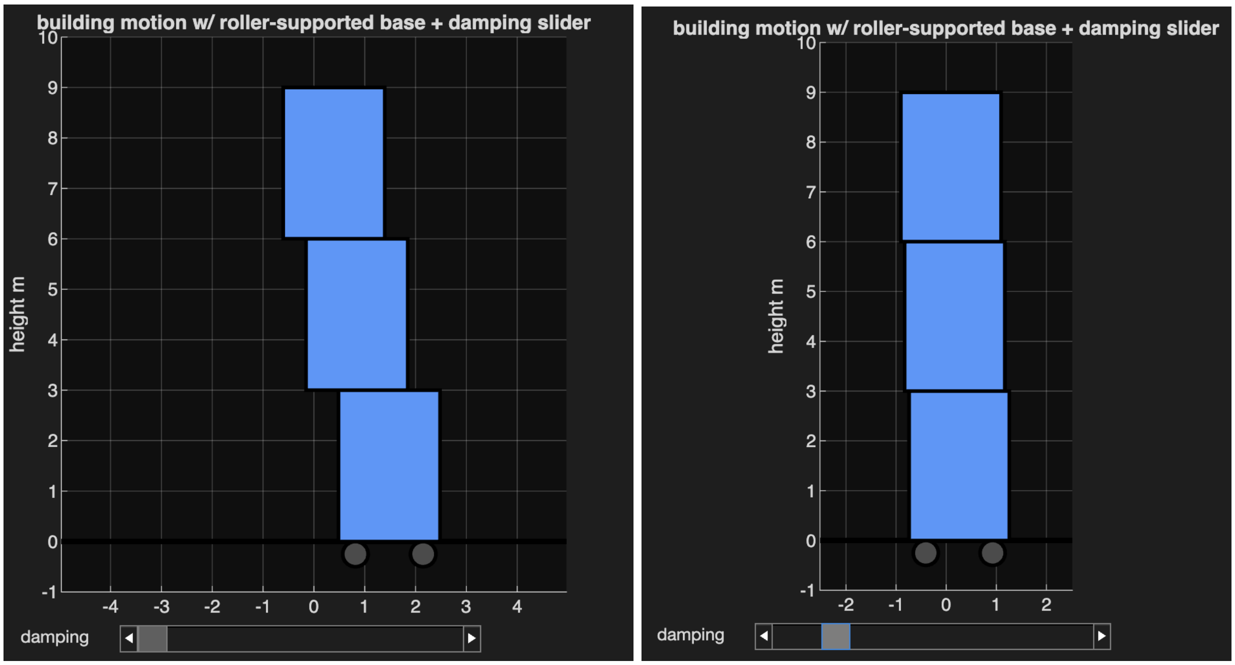

Figure 6: Above is a more generalized version of the script we used to generate the visualization based on data. This script instead took the number of floors, along with other specifications, with a function describing the displacement of each floor over time, plotting that into the visualization graph. Here, purely for our understanding, we were able to implement a damping slider to observe how the response changed as we increased the damping.

Our visualizations of multi-story buildings subject to such motion helped us to see, in a barebones way, what the data and systems we were modeling may look like in real life. Particularly from our data-driven visualization, we corroborated the natural phenomenon of higher floors experiencing a larger response. We were also able to see how actuator placement reflected in data differences translated to actual building motion. Overall, we found this to be a good way to cross-check our math and research in a more physical way.

References

Suhardjo, J., et al. “Feedback-feedforward control of structures under seismic excitation.” *Structural Safety*, Volume 8, Issues 1-4, 1990, pp 69-89. https://doi.org/10.1016/0167-4730(90)90031-J.

Nagashima, Ichiro and Ryota Maseki. “Study on feed-forward control of base-isolated buildings using predicted propagation of seismic waves.” https://www.iitk.ac.in/nicee/wcee/article/WCEE2012_1671.pdf

Abdulateef, W.S., Hejazi, F. Adaptive Real Time Efficient Control Algorithm for Buildings Under Wind and Earthquake Forces. *Int J Civ Eng* 22, 2193–2231 (2024). https://doi.org/10.1007/s40999-024-00990-1

Zhang, Xiaozhe and Franklin Y. Cheng, "Control Force Estimation in Seismic Building Design," *Structures Congress 2010*, pp. 1510 - 1522, American Society of Civil Engineers, Jan 2010. https://doi.org/10.1061/41130(369)137

Tai WC and Ikenaga M. “A semi-active control system in coupled buildings with base-isolation and magnetorheological dampers using an adaptive neuro-fuzzy inference system.” *Front. Built Environ.* Volume 8, 2022. https://doi.org/10.3389/fbuil.2022.1057962