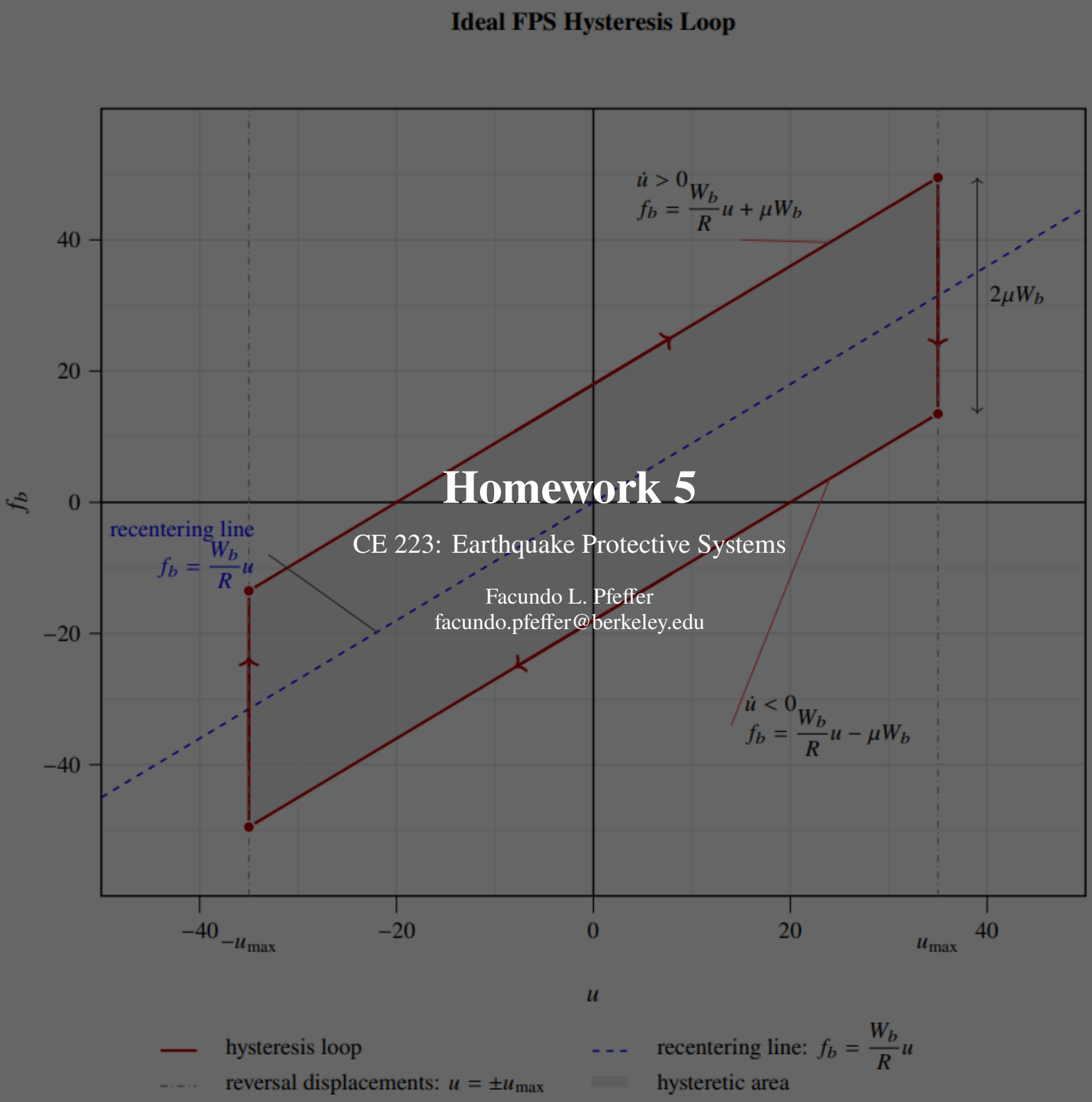
Self grade: 2/2



Homework 5

CE 223: Earthquake Protective Systems

Facundo L. Pfeffer
facundo.pfeffer@berkeley.edu

Contents

0 Theoretical Preface

The objective for this section is to introduce the fundamentals of the theory behind the derivations behind the practical examples that are solved later in Section 1 and Section 2. For further reading in this subject, the referenced material can be found in Section B.

0.1 Buckling of laminated elastomeric bearings

A laminated elastomeric bearing subjected to axial compression may buckle because lateral bending and shear deformation interact under the compressive load. A convenient formulation is obtained by combining two ingredients:

1. an effective flexural rigidity for the laminated rubber-steel assembly, obtained from the mechanics of bonded rubber layers[1, 2, 3], and
2. a homogenized shear-flexible beam model for the entire bearing.

Compression is taken as positive throughout. The coordinate x is measured along the bearing height, from the bottom end plate to the top end plate.

0.1.1 Effective flexural rigidity of a bonded rubber layer

Consider first a single circular rubber layer of radius R and thickness t, bonded to rigid plates. Let G denote the shear modulus of the elastomer. When the two plates undergo a small relative rotation φ, the hydrostatic pressure field $p(x_1, x_2)$ that develops in the rubber plane satisfies, under the incompressibility assumption,

$$\nabla^2 p = \frac{12G\varphi}{t^3}\, x_1 \qquad \text{in the loaded plan area,}$$

with boundary condition

$$p = 0 \qquad \text{at the free edge.}$$

For a circular layer it is convenient to use polar coordinates (r, θ), with

$$x_1 = r\cos\theta.$$

Then

$$\frac{\partial^2 p}{\partial r^2} + \frac{1}{r}\frac{\partial p}{\partial r} + \frac{1}{r^2}\frac{\partial^2 p}{\partial \theta^2} = \frac{12G\varphi}{t^3}\, r\cos\theta, \qquad p(R,\theta) = 0.$$

The solution with the required angular symmetry is

$$p(r,\theta) = -\frac{3G\varphi}{2t^3}\,(R^2 - r^2)\, r\cos\theta.$$

The bending moment produced by this pressure field about the centroidal axis perpendicular to x_1 is

$$M = -\int_A p\, x_1\, dA = -\int_0^{2\pi}\int_0^R p(r,\theta)\,(r\cos\theta)\, r\, dr\, d\theta = \frac{\pi G R^6}{8t^3}\,\varphi.$$

Introducing the curvature

$$\kappa = \frac{\varphi}{t},$$

the preceding result may be written as

$$M = \frac{\pi G R^6}{8t^2}\,\kappa.$$

This expression already has the form of a linear bending law. It is therefore natural to *define* the effective flexural rigidity of the bonded rubber layer by identifying the coefficient of χ in:

$$M = (EI)_{\text{eff}}^{\text{inc}}\,\chi. \tag{0.1}$$

Comparison with the preceding expression gives:

$$(EI)_{\text{eff}}^{\text{inc}} = \frac{\pi G R^6}{8t^2}.$$

It is now convenient to rewrite this result in terms of standard geometric quantities of the circular planform. The second moment of area of the loaded planform about a centroidal bending axis is:

$$I = \frac{\pi R^4}{4},$$

and the first shape factor of one rubber layer is:

$$S = \frac{\text{loaded area}}{\text{area free to bulge}} = \frac{\pi R^2}{2\pi R t} = \frac{R}{2t}.$$

Substitution of these two relations into the preceding expression gives:

$$(EI)_{\text{eff}}^{\text{inc}} = 2GIS^2.$$

Introducing also the effective compression modulus of the bonded rubber layer under the incompressibility assumption:

$$E_c = 6GS^2,$$

the same result may be written as:

$$(EI)_{\text{eff}}^{\text{inc}} = 2GIS^2 = \frac{1}{3}E_c I. \tag{0.2}$$

Equation **Eq. (0.2)** shows that the bending stiffness of the bonded rubber layer can be expressed either in terms of the shear modulus G and shape factor S, or equivalently in terms of the effective compression modulus E_c.

0.1.2 Remark on compressibility

If bulk compressibility is included, the pressure field no longer satisfies a Poisson equation but a modified Helmholtz equation:

$$\nabla^2 p - \lambda^2 p = \frac{\lambda^2 K \varphi}{t}\,x_1, \qquad \lambda^2 = \frac{12G}{Kt^2},$$

where K is the bulk modulus of the elastomer. In that case, the effective flexural rigidity is smaller than the incompressible value in **Eq. (0.2)**. Since the problem solved later provides only G, the incompressible expressions are the relevant ones.

0.1.3 Homogenized properties of the laminated bearing

Now consider a laminated bearing composed of multiple rubber layers separated by steel shims and bounded by end plates. Let

$$A = \text{loaded plan area of the bearing}, \qquad I = \text{second moment of that plan area},$$
$$h = \text{total bearing height}, \qquad t_r = \text{total rubber thickness}.$$

Because the shear and rotational compliances of the individual rubber layers combine in series through the laminate, the homogenized properties used in the beam model are

$$A_s = A\frac{h}{t_r}, \qquad EI_s = (EI)_{\text{eff}}\frac{h}{6t_r}.$$

For a circular bearing and incompressible bending response, $(EI)_{\text{eff}}$ is taken from **Eq. (0.2)**.

0.1.4 Beam kinematics

Let $v(x)$ denote the lateral displacement of the centroidal axis of the homogenized bearing, and let $\psi(x)$ denote the rotation of the cross-section. A prime denotes differentiation with respect to x.

The engineering shear strain is defined as

$$\gamma(x) = v'(x) - \psi(x),$$

and the bending curvature is

$$\kappa(x) = \psi'(x).$$

The corresponding constitutive relations are

$$V(x) = GA_s\,\gamma(x) = GA_s\big(v'(x) - \psi(x)\big),$$

$$M(x) = EI_s\,\kappa(x) = EI_s\,\psi'(x),$$

where $V(x)$ is the internal shear force and $M(x)$ is the internal bending moment.

0.1.5 Equilibrium under axial compression

Let P be the axial compressive load. To derive the buckling condition, a small initial lateral offset is retained. Let v_0 denote that offset, let H_0 be the horizontal reaction, and let M_0 be the end moment associated with the eccentricity. For small rotations, equilibrium gives

$$M - M_0 + P(v - v_0) - H_0 x = 0,$$

$$V + H_0 - P\psi = 0.$$

Substituting the constitutive relations:

$$EI_s\,\psi' + Pv = Pv_0 + M_0 + H_0 x,$$

$$GA_s\big(v' - \psi\big) - P\psi = -H_0.$$

From the second equation,

$$v' = \frac{GA_s + P}{GA_s}\psi - \frac{H_0}{GA_s}.$$

Differentiating the first equilibrium equation and eliminating ψ leads to a differential equation for v. Equivalently, eliminating v leads to a differential equation for ψ. Introducing

$$k^2 = \frac{P(GA_s + P)}{EI_s\, GA_s}, \qquad \beta = \frac{GA_s}{GA_s + P},$$

the general solution may be written as

$$v(x) = A_1 \cos(kx) + A_2 \sin(kx) + \frac{M_0}{P} + \frac{H_0}{P}x + v_0,$$

$$\psi(x) = k\beta A_2 \cos(kx) - k\beta A_1 \sin(kx) + \frac{H_0}{P}.$$

0.1.6 Buckling condition

For a bearing bounded by rigid end plates, the cross-section rotation vanishes at both ends:

$$\psi(0) = 0, \qquad \psi(h) = 0.$$

A small initial eccentricity δ is represented through equal end moments satisfying

$$2M_0 = P\delta, \qquad \frac{M_0}{P} = \frac{\delta}{2}.$$

Taking also

$$v(0) = 0, \qquad H_0 = 0,$$

the boundary conditions give

$$A_1 = -\frac{\delta}{2}, \qquad A_2 = 0.$$

Therefore,

$$\psi(h) = -k\beta A_1 \sin(kh) = 0.$$

A nontrivial buckled configuration requires

$$\sin(kh) = 0.$$

The first buckling mode corresponds to

$$kh = \pi.$$

Using the definition of k,

$$\frac{P(GA_s + P)}{EI_s\, GA_s} = \frac{\pi^2}{h^2}.$$

After rearrangement:

$$P^2 + P_S P - P_S P_E = 0, \qquad P_S = GA_s, \qquad P_E = \frac{\pi^2 EI_s}{h^2}. \tag{0.3}$$

The positive root of **Eq. (0.3)** gives the exact critical load within the homogenized beam model:

$$P_{\text{cr}} = -\frac{P_S}{2} + \sqrt{\frac{P_S^2}{4} + P_S P_E}. \tag{0.4}$$

When the shear load scale P_S is much smaller than the Euler-type load scale P_E, the positive root of **Eq. (0.4)** reduces to

$$P_{\text{cr}} \approx \sqrt{P_S P_E}. \tag{0.5}$$

Equation **Eq. (0.4)** is the exact expression requested in the problem statement, while **Eq. (0.5)** is its standard approximation.

0.2 Friction pendulum system and its bilinear regularization

0.2.1 Rigid-superstructure idealization

A base-isolated structure supported on identical friction pendulum system (FPS) bearings may be modeled, in one horizontal direction, as a single-degree-of-freedom system when the superstructure is assumed rigid in that direction. The relative displacement of the superstructure with respect to the ground is denoted by $u(t)$, with corresponding relative velocity $\dot{u}(t)$ and relative acceleration $\ddot{u}(t)$. The ground acceleration is denoted by $\ddot{u}_g(t)$. The total superstructure mass is M, and the corresponding total supported weight is:

$$W = Mg, \tag{0.6}$$

where g is the gravitational acceleration.

If n_b identical FPS bearings are used, the vertical load carried by each bearing is:

$$W_b = \frac{W}{n_b}. \tag{0.7}$$

Because the superstructure is assumed rigid, all bearings experience the same horizontal displacement $u(t)$, and the total restoring force is obtained by summing the contributions of the individual bearings.

0.2.2 Force-displacement relation of a single FPS bearing

The slider of an FPS bearing moves on a spherical surface of radius R. The angular position of the slider is denoted by θ, measured from the vertical. The horizontal force transmitted by one bearing is denoted by f_b, the normal force by N, and the friction coefficient by μ.

Under interia-less equilibrium of the slider[1], and taking the tangential direction along the sliding surface, the normal and tangential equilibrium equations are[2]:

$$N - W_b \cos\theta - f_b \sin\theta = 0,$$

$$f_b \cos\theta - W_b \sin\theta - \mu N \ \mathrm{sgn}(\dot{\theta}) = 0.$$

Elimination of N gives:

$$f_b\Big(\cos\theta - \mu\,\mathrm{sgn}(\dot{\theta})\sin\theta\Big) = W_b\Big(\sin\theta + \mu\,\mathrm{sgn}(\dot{\theta})\cos\theta\Big). \tag{0.8}$$

For the typical operating range of seismic isolation bearings, the rotation is small, so that:

$$\sin\theta \approx \theta, \qquad \cos\theta \approx 1.$$

Also, the horizontal displacement is related to the angular rotation by:

$$u \approx R\theta. \tag{0.9}$$

[1] The slider mass is assumed negligible, so its inertial contribution $m_s\mathbf{a}_s$ is neglected in Newton's second law. The slider is therefore treated as a massless internal element satisfying instantaneous force equilibrium, while the structure as a whole is still analyzed dynamically[1].

[2] $\mathrm{sgn}(\dot{\theta})$ is the operator thar retrieves 1 when $\theta \geq 0$ and -1 if $\theta < 0$.

Using these approximations in **Eq. (0.8)** gives:

$$f_b(u) = \frac{W_b}{R}\, u + \mu W_b \ \operatorname{sgn}(\dot{u}). \tag{0.10}$$

Equation **Eq. (0.10)** shows that the force developed by one FPS bearing contains two parts: a recentering contribution proportional to u, and a friction contribution of constant magnitude μW_b whose sign depends on the direction of motion. This can be visualized in **Fig. 0.1**.

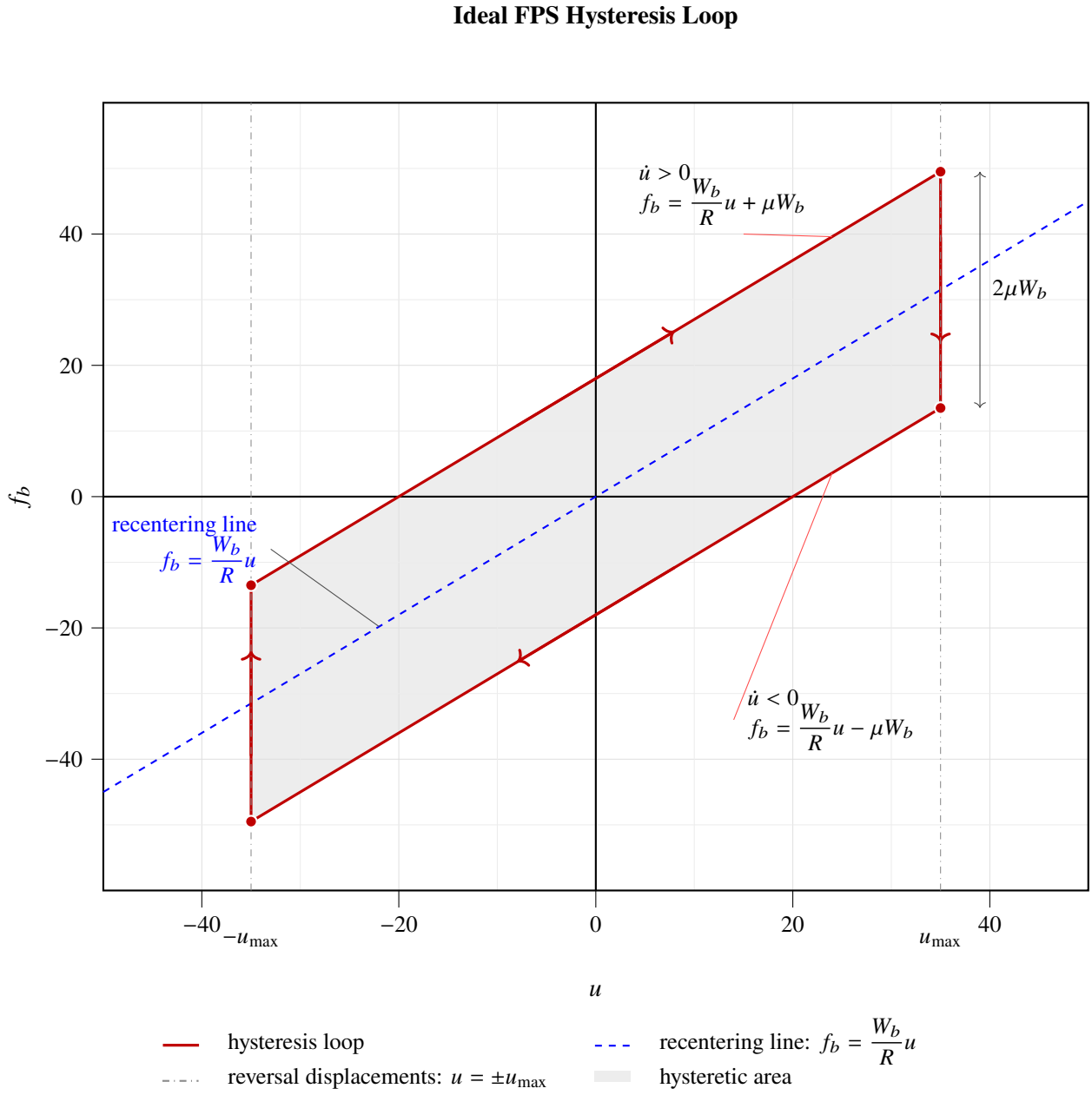


Figure 0.1: Ideal force-displacement law of a friction pendulum system bearing. The two branches have slope W_b/R and are vertically separated by $2\mu W_b$. At each displacement reversal, the sign of $\dot{u}$ changes and the response jumps from one branch to the other.

0.2.3 Total restoring force of the FPS isolation system

Since the n_b bearings act in parallel and all undergo the same displacement $u(t)$, the total restoring force is:

$$F_r = \sum_{i=1}^{n_b} f_b^{(i)}.$$

Using **Eq. (0.10)** together with **Eq. (0.7)** gives:

$$F_r(u, \dot{u}) = \frac{W}{R}\, u + \mu W\ \mathrm{sgn}(\dot{u}). \tag{0.11}$$

It is convenient to define:

$$K_p = \frac{W}{R} = \frac{Mg}{R}, \qquad Q = \mu W = \mu M g, \tag{0.12}$$

so that **Eq. (0.11)** may be written as:

$$F_r(u, \dot{u}) = K_p\, u + Q\ \mathrm{sgn}(\dot{u}). \tag{0.13}$$

The quantity K_p is the pendulum stiffness, while Q is the characteristic strength associated with friction. In the frictionless limit, the system behaves as a linear pendulum-type oscillator with circular frequency and period:

$$\omega_b = \sqrt{\frac{K_p}{M}} = \sqrt{\frac{g}{R}}, \qquad T_b = 2\pi\sqrt{\frac{R}{g}}. \tag{0.14}$$

0.2.4 Equation of motion of the isolated rigid superstructure

The horizontal relative equation of motion of the isolated rigid superstructure is:

$$M\ddot{u}(t) + F_r\big(u(t), \dot{u}(t)\big) = -M\ddot{u}_g(t). \tag{0.15}$$

Substitution of **Eq. (0.13)** into **Eq. (0.15)** gives the idealized FPS equation:

$$M\ddot{u}(t) + K_p\, u(t) + Q\ \mathrm{sgn}\big(\dot{u}(t)\big) = -M\ddot{u}_g(t). \tag{0.16}$$

The absolute acceleration of the rigid superstructure is:

$$u_t(t) = \ddot{u}(t) + \ddot{u}_g(t). \tag{0.17}$$

Since the superstructure is assumed rigid, **Eq. (0.17)** is also the floor absolute acceleration at every level above the isolation interface.

0.3 Bilinear regularization with kinematic hardening

The ideal FPS law **Eq. (0.13)** contains the discontinuous term $\text{sgn}(\dot{u})$. At each velocity reversal, this term changes sign abruptly, so the restoring force jumps from one branch of the hysteresis law to the other. For numerical response-history analysis, it is convenient to replace this ideal law by a rate-independent bilinear model with kinematic hardening and a very small yield displacement u_y. The purpose of this replacement is not to change the essential mechanics of the FPS, but to obtain a constitutive model that can be updated through a standard state-determination procedure at each time step[1].

The total displacement is decomposed into elastic and plastic parts:

$$u = u^e + u^p, \tag{0.18}$$

where u^e is the elastic displacement and u^p is the plastic displacement. Since only the elastic part contributes to the elastic restoring force, the constitutive relation is:

$$F = k\, u^e = k\big(u - u^p\big), \tag{0.19}$$

where k is the elastic stiffness of the regularized model.

A back force q is introduced to describe the translation of the yield surface during plastic loading. The yield function is:

$$f(F, q) = |F - q| - F_y \leq 0, \tag{0.20}$$

where F_y is the yield force. The quantity $F - q$ is the force measured relative to the translated yield surface. Yielding occurs when $|F - q| = F_y$.

The kinematic hardening law is taken in the form:

$$dq = H\, du^p, \tag{0.21}$$

where H is the kinematic hardening modulus. In one dimension, plastic evolution is written in terms of a nonnegative scalar increment $\Delta\gamma$, which has units of displacement and represents the magnitude of the plastic displacement increment. Accordingly,

$$\Delta u^p = \Delta\gamma\ \text{sgn}(F - q), \qquad \Delta q = H\,\Delta\gamma\ \text{sgn}(F - q), \qquad \Delta\gamma \geq 0. \tag{0.22}$$

Thus, in one dimension,

$$\Delta\gamma = |\Delta u^p|.$$

0.3.1 Post-yield tangent of the bilinear model

The post-yield tangent is obtained by considering monotonic plastic loading. Along such a loading branch, the sign of $F - q$ does not change, so the absolute value in the yield function may be differentiated without ambiguity. Since the stress point remains on the yield surface during plastic loading, the consistency condition requires:

$$df = d|F - q| = 0.$$

Because $\text{sgn}(F - q)$ is constant during monotonic plastic loading, this gives:

$$d(F - q) = 0,$$

or equivalently:

$$dF = dq.$$

Differentiating the constitutive relation **Eq. (0.19)** gives:

$$dF = k\left(du - du^p\right).$$

From the hardening law **Eq. (0.21)**, it follows that:

$$dq = H\, du^p.$$

Substitution into the consistency condition $dF = dq$ gives:

$$k\left(du - du^p\right) = H\, du^p.$$

Expanding and collecting the terms in du^p gives:

$$k\, du - k\, du^p = H\, du^p,$$

$$k\, du = (k + H)\, du^p.$$

Therefore:

$$du^p = \frac{k}{k + H}\, du.$$

Substituting this result back into $dF = k(du - du^p)$ gives:

$$dF = k\left(du - \frac{k}{k + H} du\right) = k\left(\frac{H}{k + H}\right) du.$$

Hence:

$$dF = \frac{kH}{k + H}\, du. \tag{0.23}$$

The post-yield tangent stiffness is therefore:

$$K_t = \frac{kH}{k + H}. \tag{0.24}$$

0.3.2 Matching the bilinear model to the FPS law

The regularized bilinear model is chosen so that, after yielding, it reproduces the ideal FPS law:

$$F_r(u, \dot{u}) = K_p\, u + Q\ \mathrm{sgn}(\dot{u}),$$

with:

$$K_p = \frac{W}{R}, \qquad Q = \mu W.$$

The first matching condition concerns the characteristic strength. The frictional part of the ideal FPS law has magnitude Q, so the yield force of the bilinear model is chosen as:

$$F_y = Q = \mu W. \tag{0.25}$$

The second matching condition concerns the initial elastic branch. The articulated slider is assumed to yield at displacement u_y, so the elastic stiffness must satisfy:

$$F_y = k\, u_y.$$

Therefore:

$$k = \frac{F_y}{u_y} = \frac{\mu W}{u_y}. \tag{0.26}$$

The third matching condition concerns the post-yield tangent. Once the bilinear model has yielded, its tangent stiffness must coincide with the pendulum stiffness of the FPS. Thus:

$$K_t = K_p = \frac{W}{R}. \tag{0.27}$$

Using **Eq. (0.24)** in **Eq. (0.27)** gives:

$$\frac{kH}{k+H} = K_p. \tag{0.28}$$

Multiplying by $k + H$ gives:

$$kH = K_p(k+H) = kK_p + HK_p.$$

Rearranging the terms in H gives:

$$H(k - K_p) = kK_p.$$

Therefore:

$$H = \frac{kK_p}{k - K_p}. \tag{0.29}$$

Accordingly, the regularized bilinear model that approximates the FPS is fully defined by:

$$F_y = \mu W, \qquad k = \frac{\mu W}{u_y}, \qquad K_p = \frac{W}{R}, \qquad H = \frac{kK_p}{k - K_p}.$$

When u_y is very small, k becomes very large, and the bilinear response approaches the ideal FPS law very closely.

0.3.3 State determination for the bilinear model

At time step t_{n+1}, the displacement u_{n+1} is assumed to be known from the global time-integration algorithm, and that the converged internal variables from the previous step are u_n^p and q_n. The goal is to determine the updated values F_{n+1}, u_{n+1}^p, and q_{n+1}.

Trial state The first step is to assume temporarily that the increment is elastic. Under that assumption, the plastic displacement and back force remain equal to their values from the previous step. The trial force is therefore:

$$F_{n+1}^{\text{tr}} = k\left(u_{n+1} - u_n^p\right). \tag{0.30}$$

The corresponding trial shifted force is:

$$\zeta_{n+1}^{\text{tr}} = F_{n+1}^{\text{tr}} - q_n.$$

The trial value of the yield function is then:

$$f_{n+1}^{\text{tr}} = \left|\zeta_{n+1}^{\text{tr}}\right| - F_y. \tag{0.31}$$

The interpretation is immediate:

- if $f_{n+1}^{\text{tr}} \leq 0$, the trial state lies inside or on the yield surface, so the step is elastic - if $f_{n+1}^{\text{tr}} > 0$, the trial state lies outside the yield surface, so a plastic correction is required

Elastic step If:

$$f_{n+1}^{\text{tr}} \leq 0,$$

the step is elastic, and no correction is needed. Hence:

$$F_{n+1} = F_{n+1}^{\text{tr}}, \qquad u_{n+1}^{p} = u_{n}^{p}, \qquad q_{n+1} = q_{n}. \tag{0.32}$$

Plastic correction If:

$$f_{n+1}^{\text{tr}} > 0,$$

the trial state must be returned to the yield surface. Define:

$$s_{n+1} = \operatorname{sgn}\left(\zeta_{n+1}^{\text{tr}}\right). \tag{0.33}$$

The scalar s_{n+1} gives the direction of the correction.

The corrected force, plastic displacement, and back force are written as:

$$F_{n+1} = F_{n+1}^{\text{tr}} - k\,\Delta\gamma_{n+1}\,s_{n+1}, \tag{0.34}$$

$$u_{n+1}^{p} = u_{n}^{p} + \Delta\gamma_{n+1}\,s_{n+1}, \tag{0.35}$$

$$q_{n+1} = q_{n} + H\,\Delta\gamma_{n+1}\,s_{n+1}. \tag{0.36}$$

The unknown quantity is now $\Delta\gamma_{n+1}$.

To determine it, subtract **Eq. (0.36)** from **Eq. (0.34)**. This gives:

$$F_{n+1} - q_{n+1} = \left(F_{n+1}^{\text{tr}} - q_{n}\right) - (k+H)\Delta\gamma_{n+1}\,s_{n+1}.$$

Since $\zeta_{n+1} = F_{n+1} - q_{n+1}$ and $\zeta_{n+1}^{\text{tr}} = F_{n+1}^{\text{tr}} - q_{n}$, the previous equation becomes:

$$\zeta_{n+1} = \zeta_{n+1}^{\text{tr}} - (k+H)\Delta\gamma_{n+1}\,s_{n+1}.$$

After correction, the state must lie exactly on the yield surface. Therefore:

$$|\zeta_{n+1}| = F_{y}.$$

Moreover, in one-dimensional return mapping, the correction does not change the sign of the shifted force, so:

$$\operatorname{sgn}(\zeta_{n+1}) = \operatorname{sgn}(\zeta_{n+1}^{\text{tr}}) = s_{n+1}.$$

Multiplying the expression for ζ_{n+1} by s_{n+1} gives:

$$|\zeta_{n+1}| = |\zeta_{n+1}^{\text{tr}}| - (k+H)\Delta\gamma_{n+1}.$$

Using $|\zeta_{n+1}| = F_{y}$, it follows that:

$$F_{y} = |\zeta_{n+1}^{\text{tr}}| - (k+H)\Delta\gamma_{n+1}.$$

Solving for $\Delta\gamma_{n+1}$ gives:

$$\Delta\gamma_{n+1} = \frac{|\zeta_{n+1}^{\text{tr}}| - F_{y}}{k+H} = \frac{f_{n+1}^{\text{tr}}}{k+H}. \tag{0.37}$$

Substituting **Eq. (0.37)** into the correction formulas yields the explicit plastic update:

$$F_{n+1} = F_{n+1}^{\text{tr}} - \frac{k}{k+H}\,f_{n+1}^{\text{tr}}\,\operatorname{sgn}\left(\zeta_{n+1}^{\text{tr}}\right), \tag{0.38}$$

$$u_{n+1}^{p} = u_{n}^{p} + \frac{1}{k+H}\,f_{n+1}^{\text{tr}}\,\operatorname{sgn}\left(\zeta_{n+1}^{\text{tr}}\right), \tag{0.39}$$

$$q_{n+1} = q_{n} + \frac{H}{k+H}\,f_{n+1}^{\text{tr}}\,\operatorname{sgn}\left(\zeta_{n+1}^{\text{tr}}\right). \tag{0.40}$$

Algorithmic tangent The algorithmic tangent required in iterative equilibrium solution is obtained directly from the two cases above. For an elastic step, the tangent is the elastic stiffness k. For a plastic step, the tangent is the post-yield tangent $kH/(k+H)$. Hence:

$$K_{\text{alg}} = \begin{cases} k, & f_{n+1}^{\text{tr}} \leq 0, \\ \dfrac{kH}{k+H}, & f_{n+1}^{\text{tr}} > 0. \end{cases} \tag{0.41}$$

0.3.4 Resulting nonlinear equation of motion for the regularized FPS model

Once the ideal FPS law is replaced by its bilinear regularization, the governing equation of motion retains the form:

$$M\ddot{u}(t) + F(t) = -M\ddot{u}_g(t), \tag{0.42}$$

where $F(t)$ is now obtained from the state-determination procedure described above, rather than directly from the ideal expression **Eq. (0.13)**.

The total, or absolute, floor acceleration is defined as:

$$u_t(t) = \ddot{u}(t) + \ddot{u}_g(t). \tag{0.43}$$

Because the superstructure is assumed rigid, this same quantity applies at all floors above the isolation level.

0.3.5 Newmark time-integration procedure for the regularized FPS model

Equation **Eq. (0.42)** is solved in the time domain by means of the constant-average-acceleration Newmark method, for which

$$\beta = \frac{1}{4}, \qquad \gamma = \frac{1}{2}.$$

Because the restoring force depends on internal variables through the bilinear constitutive update, equilibrium at each time step must be enforced iteratively.

Let

$$t_n = n\Delta t,$$

and suppose that, at the end of step n, the converged quantities

$$u_n, \qquad \dot{u}_n, \qquad \ddot{u}_n, \qquad u_n^p, \qquad q_n$$

are known. The objective at time t_{n+1} is to determine

$$u_{n+1}, \qquad \dot{u}_{n+1}, \qquad \ddot{u}_{n+1}, \qquad u_{n+1}^p, \qquad q_{n+1},$$

together with the restoring force F_{n+1}.

General dynamic equation and specialization to the present model. For a general single-degree-of-freedom system with an equation of motion of the form of **Eq. (0.42)**, the energy dissipation is represented by the hysteretic restoring force itself through the bilinear constitutive law. Accordingly,

$$c = 0,$$

and the governing equation reduces to **Eq. (0.42)**, which is the equation that is discretized by the Newmark method in the present implementation.

General Newmark relations. The Newmark method approximates the displacement and velocity at t_{n+1} in terms of the acceleration at the end of the step. The general formulas are[4]:

$$u_{n+1} = u_n + \Delta t\, \dot{u}_n + \Delta t^2 \left[\left(\frac{1}{2} - \beta \right) \ddot{u}_n + \beta \ddot{u}_{n+1} \right], \tag{0.44}$$

$$\dot{u}_{n+1} = \dot{u}_n + \Delta t \left[(1-\gamma)\ddot{u}_n + \gamma \ddot{u}_{n+1} \right]. \tag{0.45}$$

Equation **Eq. (0.44)** is solved for $\ddot{u}_{n+1}$, since the global iteration is carried out in terms of the unknown displacement u_{n+1}. This gives:

$$\ddot{u}_{n+1} = a_0 \left(u_{n+1} - u_n \right) - a_2 \dot{u}_n - a_3 \ddot{u}_n, \tag{0.46}$$

with

$$a_0 = \frac{1}{\beta \Delta t^2}, \qquad a_2 = \frac{1}{\beta \Delta t}, \qquad a_3 = \frac{1}{2\beta} - 1. \tag{0.47}$$

Substituting **Eq. (0.46)** into **Eq. (0.45)** gives the corresponding velocity update:

$$\dot{u}_{n+1} = a_1 \left(u_{n+1} - u_n \right) + \left(1 - \frac{\gamma}{\beta} \right) \dot{u}_n + \Delta t \left(1 - \frac{\gamma}{2\beta} \right) \ddot{u}_n, \tag{0.48}$$

where

$$a_1 = \frac{\gamma}{\beta \Delta t}. \tag{0.49}$$

Equations **Eq. (0.46)** and **Eq. (0.48)** show explicitly that, once a trial value of u_{n+1} is selected, both $\ddot{u}_{n+1}$ and $\dot{u}_{n+1}$ are determined.

Step equilibrium equation. At time t_{n+1}, dynamic equilibrium requires:

$$R_{n+1}(u_{n+1}) = M \ddot{u}_{n+1}(u_{n+1}) + F_{n+1}(u_{n+1}; u_n^p, q_n) + M \ddot{u}_{g,n+1} = 0. \tag{0.50}$$

The residual is written as a function of u_{n+1} because $\ddot{u}_{n+1}$ is given by **Eq. (0.46)**, while the restoring force F_{n+1} is obtained from the local bilinear constitutive update using the previous internal state (u_n^p, q_n).

Local constitutive update within a global iteration. Consider the i-th global equilibrium iteration at time t_{n+1}, and let $u_{n+1}^{(i)}$ be the current displacement iterate. The first step is the elastic predictor:

$$F_{n+1}^{\mathrm{tr},(i)} = k \left(u_{n+1}^{(i)} - u_n^p \right), \qquad \zeta_{n+1}^{\mathrm{tr},(i)} = F_{n+1}^{\mathrm{tr},(i)} - q_n, \tag{0.51}$$

followed by evaluation of the trial yield function:

$$f_{n+1}^{\mathrm{tr},(i)} = \left| \zeta_{n+1}^{\mathrm{tr},(i)} \right| - F_y. \tag{0.52}$$

If

$$f_{n+1}^{\mathrm{tr},(i)} \leq 0,$$

the trial state is admissible, the step is locally elastic, and:

$$F_{n+1}^{(i)} = F_{n+1}^{\mathrm{tr},(i)}, \qquad u_{n+1}^{p,(i)} = u_n^p, \qquad q_{n+1}^{(i)} = q_n, \qquad K_{\mathrm{alg}}^{(i)} = k.$$

If

$$f_{n+1}^{\mathrm{tr},(i)} > 0,$$

the trial state lies outside the yield surface and plastic correction is required. To avoid confusion with the Newmark parameter γ, the scalar magnitude of the plastic correction is denoted here by $\Delta\gamma_{\mathrm{pl}}^{(i)}$. It is obtained from:

$$\Delta\gamma_{\mathrm{pl}}^{(i)} = \frac{f_{n+1}^{\mathrm{tr},(i)}}{k+H}. \tag{0.53}$$

The corrected restoring force and internal variables are then obtained from **Eq. (0.38)**, **Eq. (0.39)**, and **Eq. (0.40)**, replacing the increment in those expressions by $\Delta\gamma_{\mathrm{pl}}^{(i)}$. The corresponding algorithmic tangent is:

$$K_{\mathrm{alg}}^{(i)} = \frac{kH}{k+H}. \tag{0.54}$$

Linearization of the step equilibrium equation. The Newton correction requires the derivative of the residual with respect to the displacement iterate. From **Eq. (0.46)**,

$$\frac{\partial \ddot{u}_{n+1}}{\partial u_{n+1}} = a_0.$$

From **Eq. (0.48)**,

$$\frac{\partial \dot{u}_{n+1}}{\partial u_{n+1}} = a_1.$$

Therefore, for the general damped equation $M\ddot{u} + c\dot{u} + F = -M\ddot{u}_g$, the linearized tangent would be:

$$\frac{\partial R_{n+1}}{\partial u_{n+1}} = Ma_0 + ca_1 + K_{\mathrm{alg}}.$$

In the present formulation, however, $c = 0$. Hence the damping contribution vanishes, and the global tangent reduces to:

$$K_{\mathrm{tan}}^{(i)} = Ma_0 + K_{\mathrm{alg}}^{(i)}. \tag{0.55}$$

This is the origin of the expression used in the Newton iteration.

Global equilibrium iteration at t_{n+1}. Once the local constitutive update has been carried out, the residual is evaluated from **Eq. (0.50)**. The displacement iterate is then corrected according to:

$$\Delta u^{(i)} = -\frac{R_{n+1}^{(i)}}{K_{\mathrm{tan}}^{(i)}}, \qquad u_{n+1}^{(i+1)} = u_{n+1}^{(i)} + \Delta u^{(i)}. \tag{0.56}$$

The iterations are continued until both the residual and the displacement correction are below the prescribed tolerances:

$$|R_{n+1}^{(i)}| \leq \varepsilon_R, \qquad |\Delta u^{(i)}| \leq \varepsilon_u.$$

Commit of step $n+1$. After convergence, the converged displacement is accepted:

$$u_{n+1} = u_{n+1}^{(i+1)}.$$

The corresponding acceleration is then computed from **Eq. (0.46)**, and the velocity is computed from **Eq. (0.48)**. The converged internal variables and restoring force are then committed:

$$u_{n+1}^{p}, \qquad q_{n+1}, \qquad F_{n+1}.$$

Finally, the absolute floor acceleration is obtained from the discrete form of **Eq. (0.43)**:

$$u_{t,n+1} = \ddot{u}_{n+1} + \ddot{u}_{g,n+1}. \tag{0.57}$$

Remark. Because the superstructure is assumed rigid, the single response quantity $u_t(t)$ obtained from **Eq. (0.57)** is the absolute floor acceleration at every level above the isolation interface.

0.3.6 Absolute floor spectra for nonstructural components

Consider a linear nonstructural component mounted on the rigid superstructure. Let $z(t)$ denote its displacement relative to the supporting floor, let T_p denote its natural period, let $\omega_p = 2\pi/T_p$ denote its natural circular frequency, and let ζ_p denote its damping ratio. Since the support input is the floor absolute acceleration $u_t(t)$, the relative equation of motion of the component is:

$$\ddot{z}(t) + 2\zeta_p\omega_p\dot{z}(t) + \omega_p^2 z(t) = -u_t(t). \tag{0.58}$$

The corresponding absolute acceleration of the component is:

$$a_{p,\text{abs}}(t) = \ddot{z}(t) + u_t(t). \tag{0.59}$$

For a fixed value of ζ_p, the absolute floor spectrum is obtained by solving **Eq. (0.58)** for many values of T_p and recording, for each one, the peak value of $|a_{p,\text{abs}}(t)|$. Since the superstructure is rigid, the same floor spectrum applies at every floor above the isolation level.

1 Buckling formulas to the circular laminated bearing

Assignment 1

Consider a steel-laminated elastomeric bearing in which the elastomer has a shear modulus of 1.0 MPa. The bearing is 600 mm in diameter, has two 25mm-thick end plates, 10×10mm-thick rubber layers and 9×2mm thick steel shims. Calculate the critical load of the bearing using the exact formula and the approximate formula and express the difference as a percentage

The problem is solved by direct application of the formulas derived in the preface for laminated elastomeric bearings. The calculation proceeds by evaluating the geometric quantities of the circular planform, the effective flexural rigidity of one bonded rubber layer through **Eq. (0.2)**, and then the two characteristic load scales entering the exact and approximate buckling expressions.

1.1 Given data

The elastomer shear modulus and the bearing diameter are:

$$G = 1.0 \text{ MPa} = 1.0 \text{ N/mm}^2, \qquad D = 600 \text{ mm}.$$

Hence, the radius of the circular planform is:

$$R = \frac{D}{2} = 300 \text{ mm}.$$

The laminate is composed of:

$$10 \text{ rubber layers of thickness } 10 \text{ mm},$$

$$9 \text{ steel shims of thickness } 2 \text{ mm},$$

$$2 \text{ end plates of thickness } 25 \text{ mm}.$$

The total rubber thickness is:

$$t_r = 10(10) = 100 \text{ mm}.$$

The height h entering the buckling formulas does not include the end plates. Therefore:

$$h = 10(10) + 9(2) = 118 \text{ mm}.$$

1.2 Planform area and second moment of area

Because the bearing has circular planform, the loaded area is:

$$A = \frac{\pi D^2}{4} = \frac{\pi(600)^2}{4} = 90{,}000\pi = 2.82743 \times 10^5 \text{ mm}^2.$$

The second moment of area about any centroidal bending axis is:

$$I = \frac{\pi D^4}{64} = \frac{\pi(600)^4}{64} = 8.1 \times 10^8 \pi = 6.36173 \times 10^9 \text{ mm}^4.$$

1.3 Shape factor and effective compression modulus

Each rubber layer has thickness

$$t = 10 \text{ mm},$$

so the shape factor is:

$$S = \frac{R}{2t} = \frac{300}{2(10)} = 15.$$

The effective compression modulus of one bonded rubber layer is:

$$E_c = 6GS^2.$$

Substituting the numerical values gives:

$$E_c = 6(1)(15)^2 = 1350 \text{ MPa}.$$

From **Eq. (0.2)**, the effective flexural rigidity of one bonded rubber layer is:

$$(EI)_{\text{eff}}^{\text{inc}} = \frac{1}{3} E_c I.$$

Hence:

$$(EI)_{\text{eff}}^{\text{inc}} = \frac{1}{3}(1350)(6.36173 \times 10^9) = 2.86278 \times 10^{12} \text{ N mm}^2.$$

1.4 Characteristic load scales

The characteristic shear and Euler-type load scales are[1, 5]:

$$P_S = GA\frac{h}{t_r}, \qquad P_E = \frac{\pi^2}{h^2}\frac{1}{3}E_c I\frac{h}{t_r}.$$

The shear load scale is therefore:

$$P_S = (1)(2.82743 \times 10^5)\frac{118}{100} = 3.33637 \times 10^5 \text{ N}.$$

The Euler-type load scale is:

$$P_E = \frac{\pi^2}{(118)^2}\frac{1}{3}(1350)(6.36173 \times 10^9)\frac{118}{100} = 2.39445 \times 10^9 \text{ N}.$$

Thus,

$$P_E \gg P_S,$$

which justifies the approximate expression.

1.5 Exact critical load

The exact critical load is obtained from:

$$P_{\text{crit}} = \frac{-P_S + \sqrt{P_S^2 + 4P_S P_E}}{2}.$$

Substituting the numerical values gives:

$$P_{\text{crit,exact}} = \frac{-3.33637 \times 10^5 + \sqrt{(3.33637 \times 10^5)^2 + 4(3.33637 \times 10^5)(2.39445 \times 10^9)}}{2}.$$

Therefore:

$$P_{\text{crit,exact}} = 2.80981 \times 10^7 \text{ N} = 28.10 \text{ MN}.$$

1.6 Approximate critical load

Since $P_E \gg P_S$, the approximation becomes:

$$P_{\text{crit}} \approx \sqrt{P_S P_E}.$$

Hence:

$$P_{\text{crit,approx}} = \sqrt{(3.33637 \times 10^5)(2.39445 \times 10^9)} = 2.82644 \times 10^7 \text{ N} = 28.26 \text{ MN}.$$

1.7 Percentage difference

The percentage difference between the approximate and exact values, measured relative to the exact result, is:

$$\frac{P_{\text{crit,approx}} - P_{\text{crit,exact}}}{P_{\text{crit,exact}}} \times 100 = \frac{28.26 - 28.10}{28.10} \times 100 = 0.59\%.$$

Thus, the approximate expression slightly overestimates the exact critical load.

1.8 Final results

The two requested values are:

$$\boxed{P_{\text{crit,exact}} = 28.10 \text{ MN}}$$

$$\boxed{P_{\text{crit,approx}} = 28.26 \text{ MN}}$$

The relative difference is:

$$\boxed{\text{Percentage difference} = 0.59\%}$$

2 Linear Viscous Dissipation Models Comparison

Assignment 2

A three-story reinforced concrete building (plan and elevation shown below) is supported on 15 identical friction pendulum system (FPS) bearings. The superstructure of total mass

$$M = 1.47 \times 10^6 \text{ kg}$$

can be assumed rigid. The FPS isolators have a dish radius

$$R = 1.0 \text{ m},$$

a friction coefficient

$$\mu = 0.03,$$

and the articulated slider has a yield displacement

$$u_y = 0.03 \text{ mm}.$$

(a) Using the rate-independent bilinear plasticity model discussed in lecture to describe the behavior of isolators, conduct time history analysis of the structure subjected to the Kobe and Sylmar ground motions[3]. Make one full-page figure for each ground motion, showing five subplots (from top to bottom): the displacement history, the velocity history, the force history, the absolute acceleration history, and the ground acceleration. Indicate the maximum responses on the graphs. Make another plot of force versus displacement for the duration of the motion.

(b) Use an equivalent viscously damped linear system to compute the response due to the Kobe and Sylmar motions. Show your results in the same fashion as part (a). Compare the peak responses to those from (a) and comment.

(c) One of the primary purposes of seismic isolation is to reduce the demands on nonstructural components (NSCs). Here, the interest is in whether the equivalent linear systems can capture those demands accurately. NSCs are often modeled as linear viscously damped oscillators with period T_p and damping ratio ζ_p, mounted on the structure. A decoupled analysis approach is taken where the structure is analyzed first (assuming that the NSCs are not there) to compute the floor absolute acceleration response. This is, in turn, used as input to compute the absolute acceleration of the NSC with period T_p and damping ratio ζ_p. If this is repeated for many values of T_p, while keeping ζ_p fixed, an absolute floor acceleration spectrum (or, for short, floor spectrum) can be generated. Use absolute floor acceleration time histories from (a) (considered to be the "exact" solution) and (b) to generate "floor spectra" for $\zeta_p = 2\%$ damping. Recall that the superstructure is assumed rigid, so the floor spectra will be the same at all levels above the isolation. Also, note the floor spectra should be generated with absolute acceleration as input, not relative. What is observed? Since nonstructural components tend to be stiff, focus attention on frequencies above, say, 2 Hz.

[3]For a discussion on how to conduct nonlinear response history analysis of SDOF systems with Newmark's method, see Chopra §5.7[4], "Nonlinear Systems: Newmark's Method." Follow the procedure therein (see Table 5.7.1) together with the state determination (in step 3.6) approach presented in class[1].

2.1 Nonlinear response-history analysis with the bilinear FPS model

Part (a) is carried out by applying the nonlinear FPS formulation developed in the preface. Specifically, the structural response is obtained from the regularized equation of motion **Eq. (0.42)**, where the restoring force is governed by the bilinear model calibrated through **Eq. (0.25)**, **Eq. (0.26)**, and **Eq. (0.29)**. At every time step, the restoring force and internal variables are updated through the trial-state and correction equations **Eq. (0.31)**, **Eq. (0.32)**, **Eq. (0.38)**, **Eq. (0.39)**, and **Eq. (0.40)**, and the time integration is performed with the Newmark scheme summarized in Section 0.3.5.

For the two ground motions analyzed[6, 7], the reported figure collects the histories of displacement, velocity, restoring force, absolute acceleration, and ground acceleration, while a separate force-displacement plot is used to show the hysteretic behavior of the isolation system. The plotted absolute acceleration corresponds to the quantity defined in **Eq. (0.43)**.

2.1.1 Kobe ground motion

Figure **Fig. 2.1** presents the nonlinear response histories for the Kobe record. The maximum values of the response quantities are indicated directly on the plots.

2.1.2 Sylmar ground motion

Figure **Fig. 2.2**, in turn, presents the corresponding nonlinear response histories for the Sylmar record.

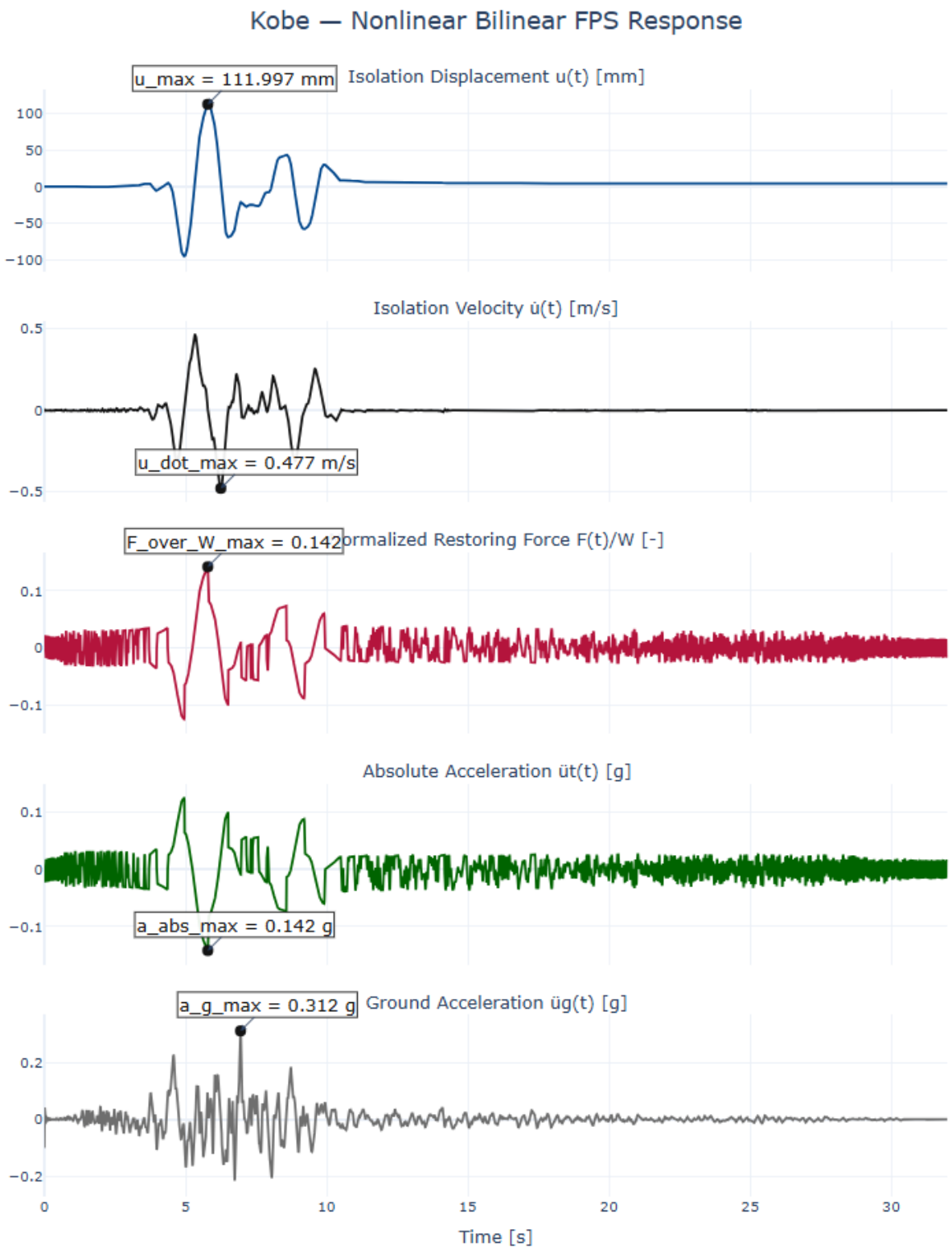


Figure 2.1: Nonlinear response-history results for the Kobe ground motion using the bilinear FPS model. From top to bottom: displacement, velocity, restoring force, absolute acceleration, and ground acceleration. Available online for dynamic visualization.

Figure 2.2: Nonlinear response-history results for the Sylmar ground motion using the bilinear FPS model. From top to bottom: displacement, velocity, restoring force, absolute acceleration, and ground acceleration. Available online for dynamic visualization.

2.1.3 Force-displacement response

The force-displacement plots provide a direct view of the hysteretic behavior predicted by the bilinear model. In particular, they make visible the yielding and post-yield recentering behavior that cannot be represented by a purely linear model.

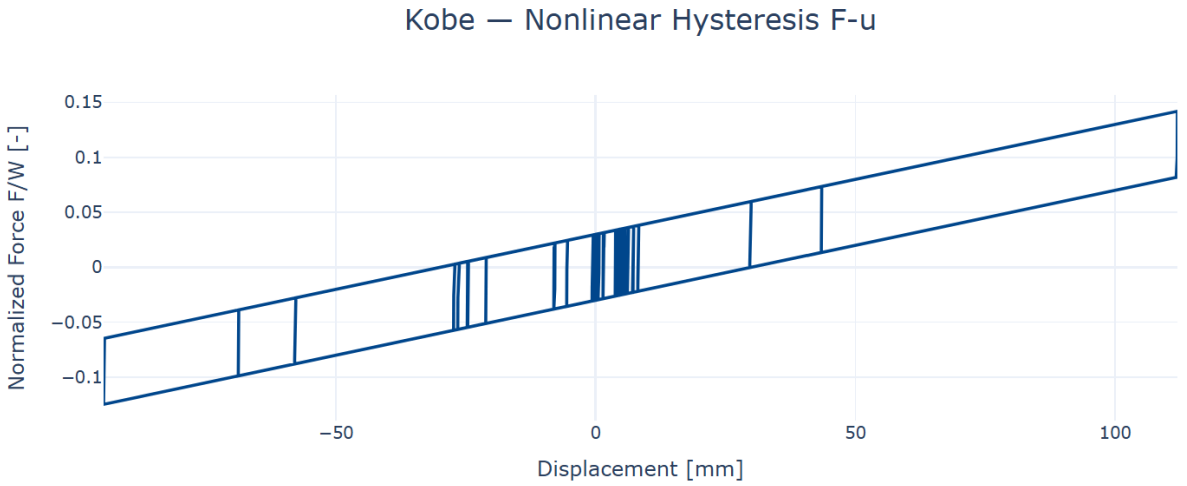


Figure 2.3: Nonlinear force-displacement response for the Kobe ground motion. Available online for dynamic visualization.

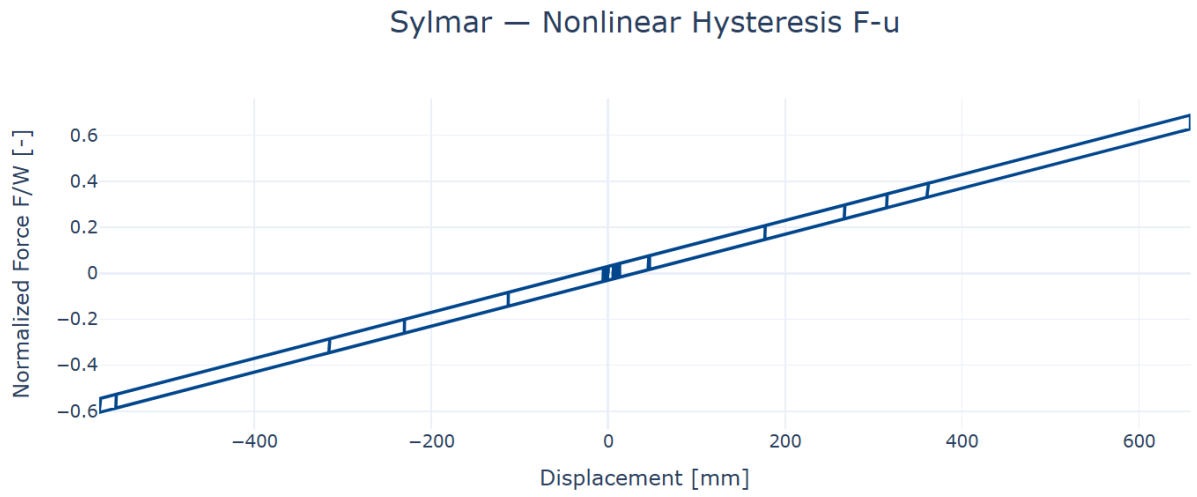


Figure 2.4: Nonlinear force-displacement response for the Sylmar ground motion. Available online for dynamic visualization.

2.2 Equivalent viscously damped linear analysis

Part (b) is carried out by replacing the bilinear hysteretic isolator model used in part (a) with the equivalent viscously damped linear representation introduced in the preface. In that approximation, the FPS isolation system is modeled by an effective linear stiffness together with an equivalent viscous damping chosen so that the global response of the isolation layer is matched in an average sense. The corresponding linear equation of motion is then solved in the time domain for the Kobe and Sylmar records using the same response-history framework adopted in part (a), so that the two sets of results may be compared directly. For each ground motion, one full-page figure is reported showing the histories of displacement $u(t)$, velocity $\dot{u}(t)$, restoring force $F(t)$, absolute acceleration $u_t(t)$, and ground acceleration $\ddot{u}_g(t)$. The absolute acceleration is again computed from **Eq. (0.43)**. A separate force-displacement plot is also included for each record in order to contrast the smooth elliptical response of the equivalent linear model with the hysteretic loops obtained from the bilinear formulation in part (a).

2.2.1 Kobe ground motion

Figure **Fig. 2.5** shows the response histories of the equivalent linear model under the Kobe record.

2.2.2 Sylmar ground motion

Figure **Fig. 2.6** shows the response histories of the equivalent linear model under the Sylmar record.

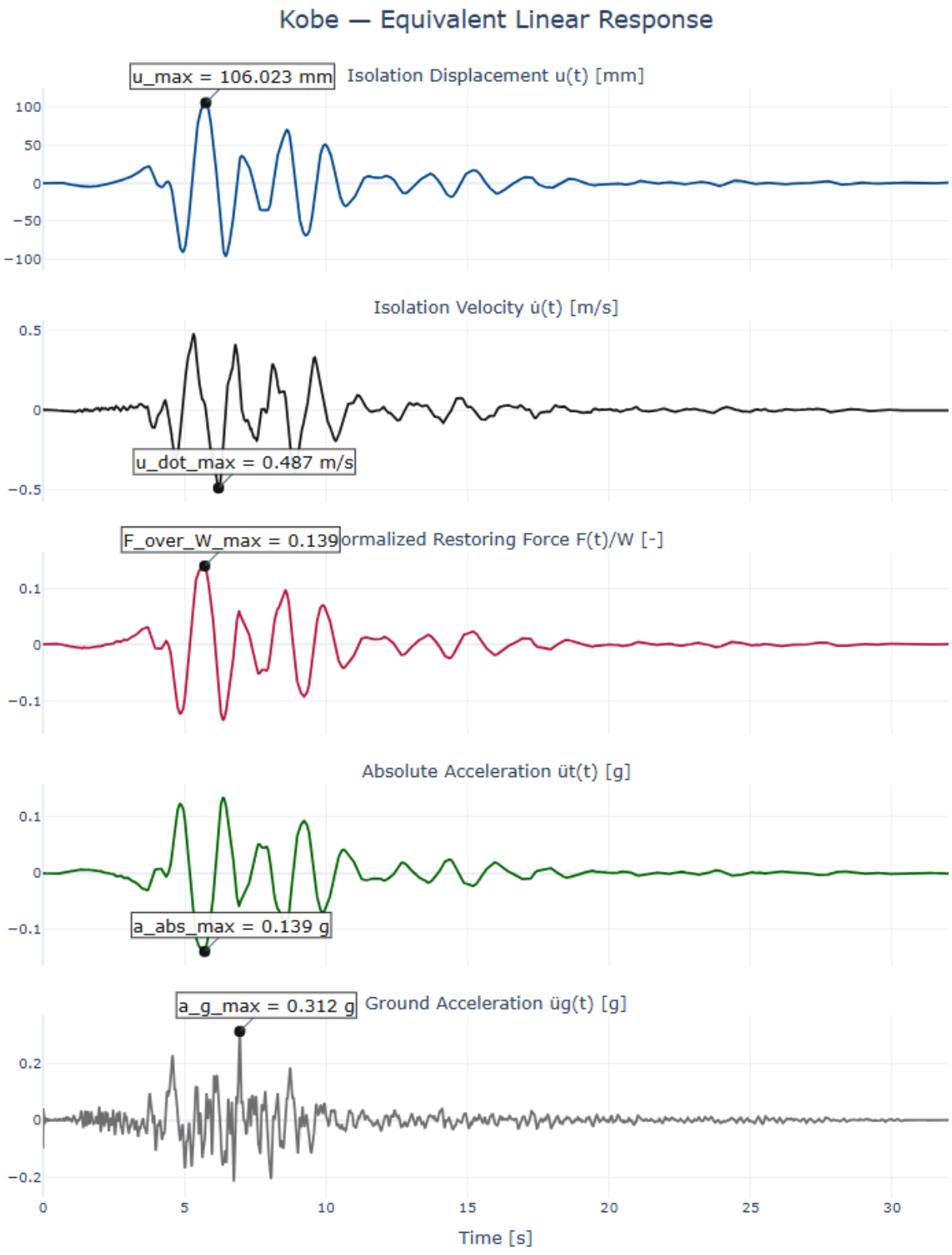


Figure 2.5: Equivalent linear response-history results for the Kobe ground motion. From top to bottom: displacement, velocity, restoring force, absolute acceleration, and ground acceleration. Available online for dynamic visualization.

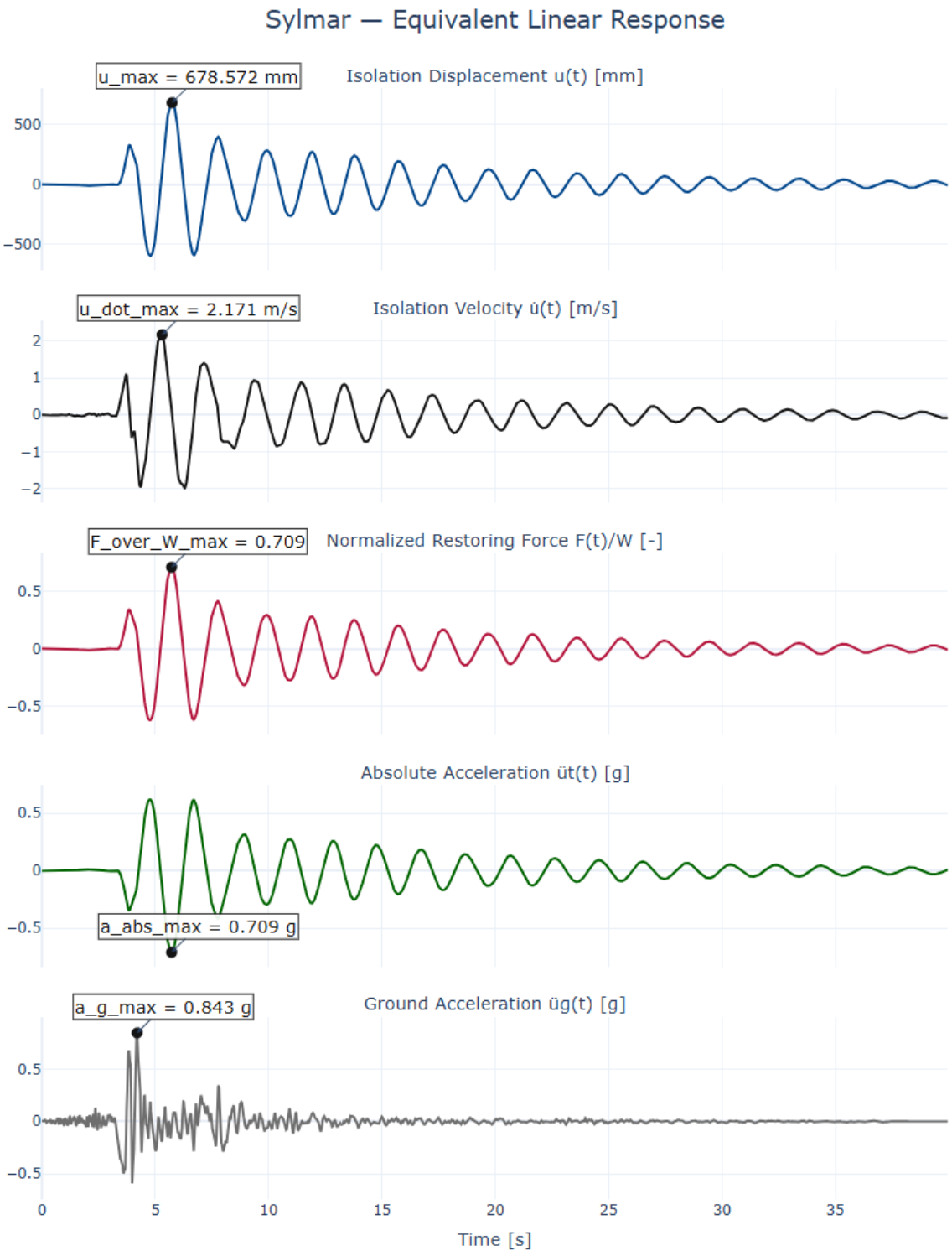


Figure 2.6: Equivalent linear response-history results for the Sylmar ground motion. From top to bottom: displacement, velocity, restoring force, absolute acceleration, and ground acceleration. Available online for dynamic visualization.

2.2.3 Force-displacement response

The corresponding force-displacement plots are shown below. Unlike the bilinear model, the equivalent linear model does not reproduce the sharp hysteretic transitions associated with sliding and re-centering. Its response is instead governed by an effective linear stiffness and an equivalent viscous dissipation mechanism.

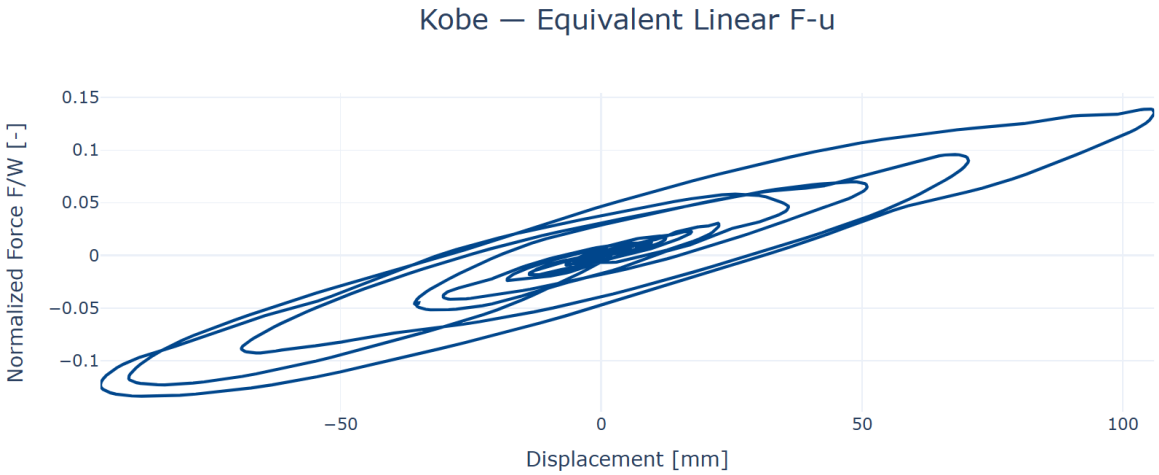


Figure 2.7: Force-displacement response of the equivalent linear model for the Kobe ground motion. Available online for dynamic visualization.

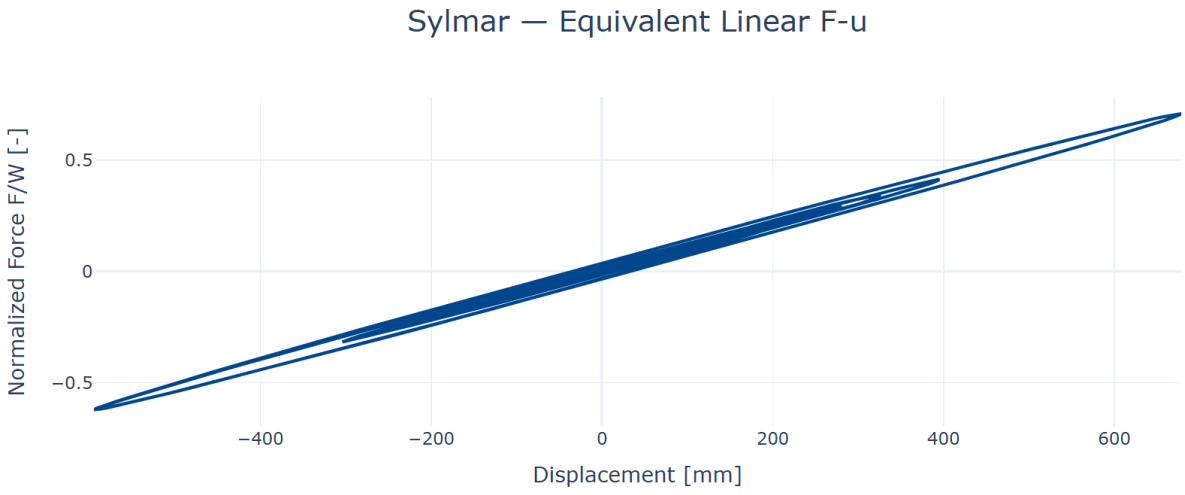


Figure 2.8: Force-displacement response of the equivalent linear model for the Sylmar ground motion. Available online for dynamic visualization.

2.2.4 Comparison of the two models

Comparison of **Fig. 2.1**–**Fig. 2.4** with **Fig. 2.5**–**Fig. 2.8** shows that the equivalent linear model can reproduce the global response only in an averaged sense, with the peak quantities compared in **Fig. 2.9** showing an acceptable behavior for analyzing the global response of the structure.

The nonlinear bilinear model by itself captures the actual hysteretic evolution of the FPS bearings and therefore

provides a more faithful representation of the isolation response. Any comparison of peak displacement, velocity, restoring force, and absolute acceleration should therefore be interpreted with the nonlinear solution taken as a better reference.

Peak Response Comparison

Motion	Model	Peak |u| [mm]	Peak |u̇| [m/s]	Peak |F| [MN]	Peak |üt| [g]
Kobe	Nonlinear	111.997	0.477	2.047	0.142
Kobe	Equivalent Linear	106.023	0.487	2.007	0.139
Sylmar	Nonlinear	658.157	2.085	9.920	0.688
Sylmar	Equivalent Linear	678.572	2.171	10.226	0.709

Figure 2.9: Comparison of peak quanitites.

2.3 Floor spectra and implications for nonstructural components

Part (c) is addressed by using the absolute floor acceleration histories obtained in parts (a) and (b) as support input for linear nonstructural components, exactly as described in the preface. Since the superstructure is assumed rigid, the same floor acceleration history applies at every level above the isolation interface, and therefore only one floor spectrum is needed for each structural model and each ground motion. For a nonstructural component with period T_p, circular frequency $\omega_p = 2\pi/T_p$, and damping ratio $\zeta_p = 2\%$, the response is computed from the relative equation of motion **Eq. (0.58)**, while the quantity of interest is the absolute component acceleration defined by **Eq. (0.59)**. Repeating this calculation for a range of values of T_p produces the floor spectra shown below[4]. The comparison between the spectra generated from the nonlinear solution in part (a) and from the equivalent linear solution in part (b) is then used to assess whether the equivalent linear approximation can reproduce the acceleration demands transmitted to nonstructural components, particularly in the higher-frequency range above 2 Hz.

2.3.1 Kobe ground motion

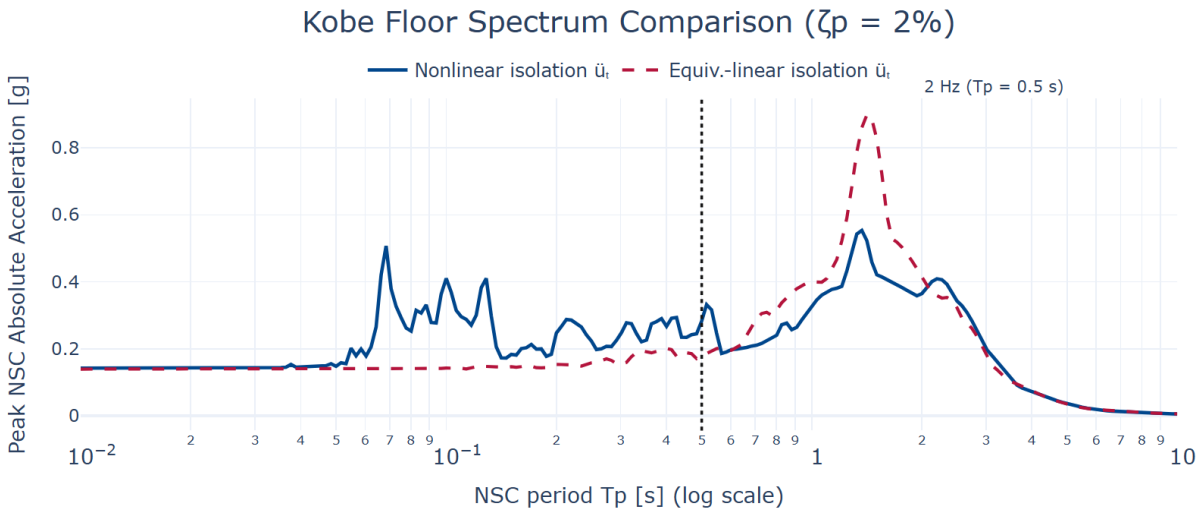


Figure 2.10: Absolute floor acceleration spectrum for the Kobe ground motion, generated from the nonlinear and equivalent linear structural responses. Available online for dynamic visualization.

2.3.2 Sylmar ground motion

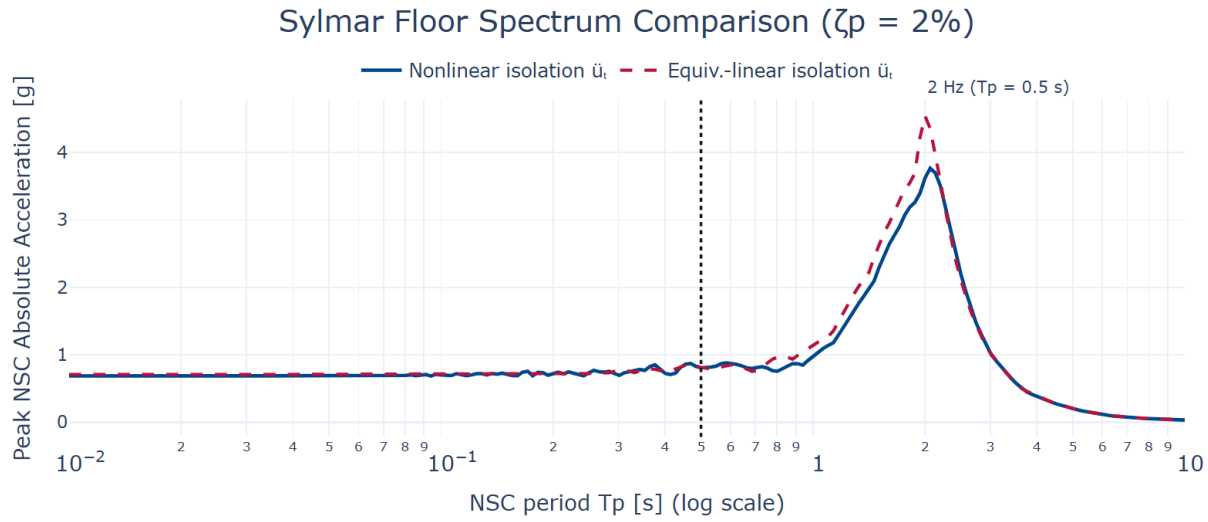


Figure 2.11: Absolute floor acceleration spectrum for the Sylmar ground motion, generated from the nonlinear and equivalent linear structural responses. Available online for dynamic visualization.

2.3.3 Discussion

The floor spectra highlight a key limitation of the equivalent linear approximation. Although an equivalent linear system may reproduce some global features of the isolated response, it does not necessarily reproduce the frequency content of the absolute floor acceleration with sufficient fidelity. This becomes especially important for nonstructural components, which are often relatively stiff and therefore sensitive to the higher-frequency range.

For the Kobe ground motion, the difference is particularly clear in **Fig. 2.10**. In the frequency range above 2 Hz, the equivalent linear model tends to overestimate the floor spectral ordinates relative to the nonlinear reference solution. Consequently, if the objective is to assess demands on acceleration-sensitive nonstructural components, the equivalent linear model may lead to misleadingly conservative predictions.

This observation is important because the performance of nonstructural components is governed not only by peak structural displacements, but also by the detailed acceleration content transmitted through the isolation system. The nonlinear bilinear model retains the actual hysteretic response of the FPS bearings and therefore provides a more reliable estimate of those demands.

A Appendix A: Implementation details of the code used for Section 2

This appendix summarizes the structure of the Python code used to generate the numerical results and figures presented in Section 2, which can be acceses through this GitHub link. Its purpose is not to reproduce the full source code, but to explain how the theoretical formulation developed in the preface is translated into a computational workflow.

More specifically, the code implements:

- the bilinear regularization of the FPS restoring force introduced in the preface through **Eq. (0.25)**, **Eq. (0.26)**, and **Eq. (0.29)**,
- the state-determination procedure given by **Eq. (0.31)**, **Eq. (0.32)**, **Eq. (0.38)**, **Eq. (0.39)**, and **Eq. (0.40)**,
- the nonlinear equation of motion **Eq. (0.42)**,
- the Newmark time-integration procedure summarized in Section 0.3.5,
- and the floor-spectrum calculation based on **Eq. (0.58)** and **Eq. (0.59)**.

Accordingly, the appendix should be read as a complement to the preface: the preface establishes the governing mechanics, while the present appendix explains how those equations are assembled and solved numerically.

A.1 Software organization of the FPS dashboard generator

The interactive dashboard is generated by a single Python module, *build_fps_dashboard.py*, located in the folder `CE223_EarthquakeProtectiveSystems/nonlinear_friction_pendulum/`. The code is organized as a compact object-oriented pipeline: tikz

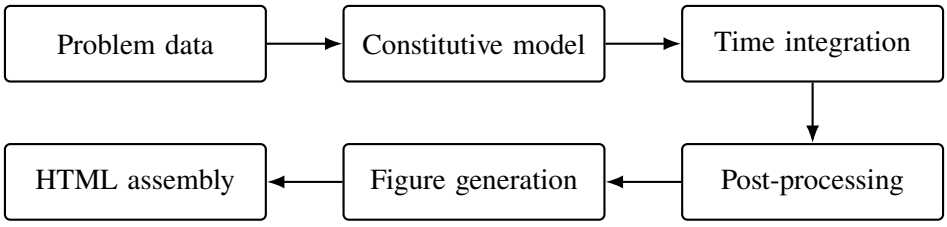


Figure A.1: Main software pipeline used for the FPS analysis workflow.

This organization mirrors the logical structure of the preface. The problem data define the FPS properties, the constitutive model implements the bilinear regularization, the time integrators solve the equations of motion, the post-processing stage generates the floor spectra, and the final layer assembles the visual report.

A.2 Module-level configuration

At the beginning of the module, a small set of constants and file paths is declared once. These include the gravitational acceleration, the locations of the Kobe and Sylmar input records, the output HTML path, and

the color palette used for plotting. Centralizing these definitions keeps the rest of the implementation free from repeated filesystem or formatting logic.

Python 3.13 Code

```
G_SI = 9.80665  # m/s^2
BASE_DIR = Path(__file__).resolve().parent
CE223_DIR = BASE_DIR.parent
HIGHLIGHTED_HTML_DIR = CE223_DIR / "highlighted_htmls"
KOBE_PATH = CE223_DIR / "input_ground_motion" / "RSN1108_KOBE_KBU090.AT2"
SYLMAR_PATH = BASE_DIR / "SYLMAR360.txt"
OUTPUT_HTML = HIGHLIGHTED_HTML_DIR / "CE223_FPS_Bilinear_Kobe_Sylmar.html"
```

A.3 Immutable data containers

The code uses dataclasses to store the physical inputs, derived FPS parameters, response histories, floor-spectrum results, and constitutive internal variables. This makes the implementation easier to read because each object has a clear mechanical meaning.

FpsProblemData stores the inputs appearing in the problem statement, namely M, R, μ, u_y, and n_b. From these, the total supported weight $W = Mg$ is derived. *BilinearFpsParameters* stores the bilinear parameters (K_p, Q, F_y, k, H) that are obtained from the matching relations introduced in the preface. *GroundMotionRecord* stores the ground acceleration array together with the time step Δt. *TimeHistoryResult* is the common output structure for the time-history solvers, containing u, $\dot{u}$, $\ddot{u}$, u_t, and F. *FloorSpectrumResult* stores the period grid and the corresponding peak spectral ordinates. Finally, *BilinearState* stores the two internal variables of the constitutive update, namely u^p and q.

Python 3.13 Code

```
@dataclass(frozen=True)
class FpsProblemData:
    total_mass: float = 1.47e6
    n_bearings: int = 15
    radius: float = 1.0
    friction_coefficient: float = 0.03
    yield_displacement_mm: float = 0.03

@dataclass
class TimeHistoryResult:
    time: np.ndarray
    ground_acceleration: np.ndarray
    displacement: np.ndarray
    velocity: np.ndarray
    relative_acceleration: np.ndarray
    absolute_acceleration: np.ndarray
    restoring_force: np.ndarray
```

A.4 Ground motion input

The class *GroundMotionLoader* is responsible for reading the Kobe and Sylmar records. The PEER-style header is parsed in order to recover the sampling interval Δt, and the acceleration values, originally stored in units of g, are converted to SI units. This stage produces the discrete input $\ddot{u}_g(t)$ required by the structural equation of motion **Eq. (0.42)**.

Python 3.13 Code

```
class GroundMotionLoader:
    @staticmethod
    def load_peer_file(path: Path) -> GroundMotionRecord:
        dt = GroundMotionLoader._parse_dt(path)
        acc_g = np.loadtxt(path, comments="%")
        acc_g = np.asarray(acc_g, dtype=float).ravel()
        return GroundMotionRecord(name=path.name, dt=dt,
                                  acceleration_mps2=acc_g * G_SI)
```

A.5 Assembly of the bilinear FPS parameters

The class *FpsParameterBuilder* converts the physical problem data into the bilinear parameters used in the constitutive model. This is the direct computational implementation of the matching relations developed in the preface. In particular,

$$F_y = \mu W, \qquad k = \frac{F_y}{u_y}, \qquad K_p = \frac{W}{R}, \qquad H = \frac{kK_p}{k - K_p},$$

which correspond to **Eq. (0.25)**, **Eq. (0.26)**, **Eq. (0.27)**, and **Eq. (0.29)**. Thus, the code constructs a bilinear model whose characteristic strength and post-yield tangent reproduce those of the ideal FPS law.

Python 3.13 Code

```
class FpsParameterBuilder:
    @staticmethod
    def from_problem_data(data: FpsProblemData) -> BilinearFpsParameters:
        weight = data.weight
        kp = weight / data.radius
        q = data.friction_coefficient * weight
        fy = q
        k = fy / data.yield_displacement
        h = (k * kp) / (k - kp)
        return BilinearFpsParameters(...)
```

A.6 Constitutive update and return mapping

The class *BilinearConstitutiveModel* implements the state-determination procedure introduced in the preface. Given a trial displacement u_{n+1} and the previously converged state (u_n^p, q_n), it first forms the elastic predictor:

$$F_{n+1}^{\text{tr}} = k(u_{n+1} - u_n^p), \qquad \zeta_{n+1}^{\text{tr}} = F_{n+1}^{\text{tr}} - q_n,$$

and evaluates the trial yield function

$$f_{n+1}^{\text{tr}} = \left|\zeta_{n+1}^{\text{tr}}\right| - F_y,$$

as in **Eq. (0.31)**.

If the trial state satisfies $f_{n+1}^{\text{tr}} \leq 0$, the update is elastic and the algorithmic tangent is k, as in **Eq. (0.32)**. Otherwise, the code computes the plastic correction increment

$$\Delta\gamma = \frac{f_{n+1}^{\text{tr}}}{k + H},$$

and updates the restoring force and internal variables according to **Eq. (0.38)**, **Eq. (0.39)**, and **Eq. (0.40)**. The corresponding algorithmic tangent is then

$$K_{\text{alg}} = \frac{kH}{k + H},$$

as in **Eq. (0.41)**.

Python 3.13 Code

```
class BilinearConstitutiveModel:
    def update(self, displacement: float, previous_state: BilinearState):
        trial_force = k * (displacement - previous_state.plastic_displacement)
        shifted_trial_force = trial_force - previous_state.back_force
        trial_yield_cond = abs(shifted_trial_force) - fy
        if trial_yield_cond <= 0.0:
            return trial_force, BilinearState(...), k
        gamma_increment = trial_yield_cond / (k + h)
        ...
        return force, BilinearState(...), (k * h) / (k + h)
```

A.7 Nonlinear Newmark solver for Part (a)

The class *NonlinearNewmarkSolver* solves the nonlinear structural problem from Part (a). It implements the equation of motion

$$M\ddot{u} + F(u; u^p, q) = -M\ddot{u}_g,$$

namely **Eq. (0.42)**, together with the Newmark time-integration procedure described in Section 0.3.5.

At each time step, the solver performs nested equilibrium iterations on u_{n+1}. For every displacement guess, it does four things:

first, it calls the constitutive update to obtain F_{n+1}, the internal variables, and K_{alg};

second, it computes the acceleration from the Newmark kinematic relation corresponding to **Eq. (0.46)**;

third, it evaluates the dynamic residual

$$R_{n+1} = M\ddot{u}_{n+1} + F_{n+1} + M\ddot{u}_{g,n+1};$$

and fourth, it applies a scalar Newton correction based on the tangent

$$K_{\text{tan}} = Ma_0 + K_{\text{alg}},$$

exactly as established in Section 0.3.5.

After convergence, the code stores the displacement, velocity, relative acceleration, absolute acceleration, and restoring force. The absolute acceleration is computed from

$$u_t(t) = \ddot{u}(t) + \ddot{u}_g(t),$$

which is the discrete counterpart of **Eq. (0.43)**.

Python 3.13 Code

```
for i in range(1, n):
    u_guess = u[i - 1]
    for _ in range(max_iterations):
        force_i, state_i, k_alg = self.model.update(u_guess, trial_state_input)
        acc_i = a0 * (u_guess - u[i - 1]) - a2 * v[i - 1] - a3 * a[i - 1]
        residual = m * acc_i + force_i + m * ug[i]
        tangent = m * a0 + k_alg
        delta_u = -residual / tangent
        u_guess += delta_u
        ...
    u[i] = u_guess
    abs_a[i] = a[i] + ug[i]
```

A.8 Equivalent linear iteration for Part (b)

The class *EquivalentLinearSolver* implements the equivalent linear model used in Part (b). In contrast with the nonlinear solver, the isolator response is not updated through internal variables. Instead, the code uses an outer fixed-point iteration to identify an effective linear stiffness and an equivalent viscous damping ratio based on the current estimate of the peak displacement.

Given an estimate of $u_{\max}$, the code sets

$$k_{\text{eff}} = K_p + \frac{Q}{u_{\max}},$$

and then computes an equivalent viscous damping ratio through an energy-matching relation. From this damping ratio, an effective viscous coefficient c_{eff} is constructed and the corresponding linear equation of motion is solved. The resulting maximum displacement is then used to update $u_{\max}$, and the process is repeated until the equivalent linear properties stabilize.

Thus, this part of the implementation corresponds to the linearized approximation discussed in the main text, and provides the response histories used for comparison with the nonlinear bilinear solution.

Python 3.13 Code

```
for _ in range(iterations):
    k_eff = kp + q / max(umax, 1e-6)
    zeta = (2.0 * q) / (math.pi * k_eff * max(umax, 1e-6))
    c_eff = 2.0 * zeta * math.sqrt(m * k_eff)
    result = LinearNewmarkSolver.solve_sdof_base_excitation(record, m, c_eff, k_eff
        )
    umax = max(result.peak_displacement, 1e-6)
```

A.9 Linear Newmark solver for base excitation

The class *LinearNewmarkSolver* solves linear single-degree-of-freedom systems under base excitation. It is used in two places: first, in Part (b) for the equivalent linear structural model; second, in Part (c) for the nonstructural component oscillators used to generate the floor spectra.

Because the system is linear, the effective stiffness can be formed once at the beginning of the analysis, and the standard Newmark update can then be applied directly at every time step. In this sense, the linear solver plays the same role as the nonlinear solver, but without the nested constitutive update and without internal-variable evolution.

Python 3.13 Code

```
k_eff = stiffness + a0 * mass + a1 * damping
for i in range(1, n):
    p_i = -mass * ug[i]
    p_eff = p_i + mass * (...) + damping * (...)
    u[i] = p_eff / k_eff
    abs_a[i] = a[i] + ug[i]
```

A.10 Floor-spectrum calculation for Part (c)

The class *FloorSpectrumCalculator* implements the NSC analysis described in the preface. The input to this stage is the absolute floor acceleration

$$u_t(t) = \ddot{u}(t) + \ddot{u}_g(t),$$

which comes from either the nonlinear structural response of Part (a) or the equivalent linear structural response of Part (b). This is precisely the support input required by **Eq. (0.58)**.

For each oscillator period T_p, the code constructs the corresponding natural circular frequency

$$\omega_p = \frac{2\pi}{T_p},$$

solves the relative NSC equation of motion for the prescribed damping ratio ζ_p, and then reconstructs the absolute component acceleration from

$$a_{p,\mathrm{abs}}(t) = \ddot{z}(t) + u_t(t),$$

as in **Eq. (0.59)**. The peak absolute acceleration is stored as the floor-spectrum ordinate for that period. Repeating this over the full period grid produces the floor spectra reported in Section 2.

Python 3.13 Code

```
for i, period in enumerate(periods):
    omega = 2.0 * math.pi / period
    result = LinearNewmarkSolver.solve_sdof_base_excitation(...)
    component_abs_acc = result.relative_acceleration + support_abs_acceleration
    peaks[i] = np.max(np.abs(component_abs_acc))
```

A.11 Visualization and HTML report assembly

The final stage of the code is devoted to presentation rather than mechanics. The class *FigureFactory* creates the Plotly figures[8], including the stacked time-history plots, the force-displacement loops, and the floor spectra. The class *HtmlReportBuilder* then assembles those figures into a single HTML document, embeds the required mathematical rendering support, and writes the final report to disk.

The entry point *main* coordinates the full pipeline: it reads the two ground-motion records, constructs the FPS parameters, runs the nonlinear and equivalent linear analyses, generates the floor spectra, and writes the final dashboard.

Python 3.13 Code

```
def main() -> None:
    problem_data = FpsProblemData()
    parameters = FpsParameterBuilder.from_problem_data(problem_data)
    constitutive = BilinearConstitutiveModel(parameters)
    ...
    report = HtmlReportBuilder(parameters, problem_data).build(...)
    OUTPUT_HTML.write_text(report, encoding="utf-8")
```

A.12 Closing remark

The implementation is therefore fully consistent with the theoretical development of the preface. The code does not introduce an alternative model; rather, it provides a computational realization of the bilinear FPS formulation, the Newmark solution procedure, the equivalent linear comparison model, and the floor-spectrum analysis used in Section 2.

B References and Further Reading

[1] D. Konstantinidis, "Lectures for cee223: Earthquake protective systems." Course lectures, University of California, Berkeley, 2026. Spring 2026 semester SEMM MS Program.

[2] L. Anand and S. Govindjee, *Continuum Mechanics of Solids*. Oxford University Press, 07 2020.

[3] S. Govindjee, "Lectures for cee231: Solid mechanics." Course lectures, University of California, Berkeley, 2025. Fall 2025 semester SEMM MS Program.

[4] A. K. Chopra, *Dynamics of Structures: Theory and Applications to Earthquake Engineering*. Boston: Pearson Education, 4th, global edition ed., 2014.

[5] J. M. Kelly and D. A. Konstantinidis, *Mechanics of Rubber Bearings for Seismic and Vibration Isolation*. John Wiley & Sons, Ltd., 1 ed., 2011.

[6] Pacific Earthquake Engineering Research Center (PEER), "Peer nga-west2 record rsn1108: Kobe, japan (1995-01-16), kobe university, component kbu090." Ground-motion time-history record, PEER Ground Motion Database (NGA-West2), 1995. File used in this study: RSN1108_KOBE_KBU090.AT2; acceleration units in g; NPTS = 3200; DT = 0.010 s.

[7] California Department of Conservation, Division of Mines and Geology (CDMG), "Northridge earthquake record: Sylmar – hospital, 360 component, station 24514." Strong-motion accelerogram (commonly distributed through PEER/COSMOS archives), 1994. File used in this study: SYLMAR360.txt; event date 1994-01-17; acceleration units in g; NPTS = 2000; DT = 0.020 s.

[8] P. T. Inc., "Plotly: Collaborative data science." https://plotly.com/python/, 2015. Accessed: 2025-10-04.