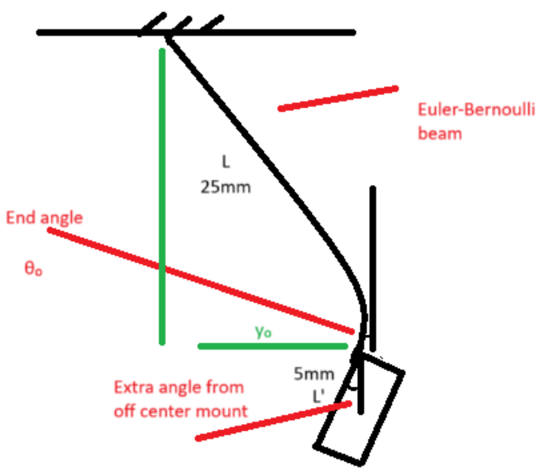
29 July 2025

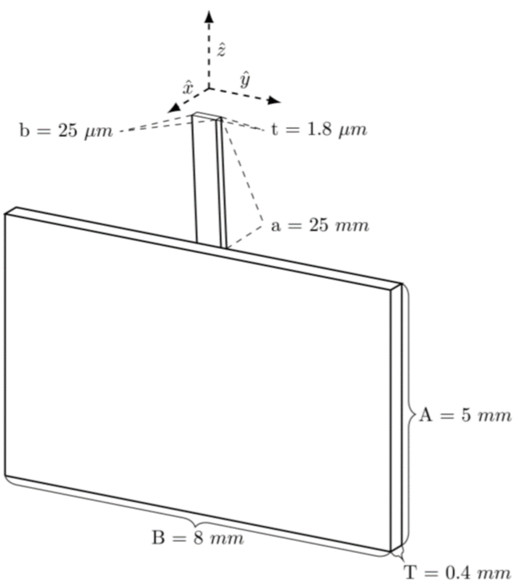
SUBJECT: Finite-size bob for uniform cross section flexure-based compound pendulum.

Refs: Anelasiticity in flexure strips revisited (Speake),Pendulum suspended from a flexure (Schlamminger)

Variables:

Name	Units	Desc.
L	m	length of flexure
L'	m	distance COM to mounting
$\theta_{\text{end shape}}$	rad	angle from vertical of the line from mounting to weight COM
y_0	m	horizontal displacement
θ_0	rad	end angle
E	Pa	elastic modulus
I	m^4	second area moment flexure cross section aligned to bending
m	kg	mass of end weight
W	N	gravitational force on end mass
$\alpha = \sqrt{\frac{W}{EI}}$	m$^{-}1$	scale constant for flexure geometry
F	N	force of the end of the flexure
τ	N m	torque of the end of the flexure
I_{cm}	kg m^2	bob moment of inertia
C	mixed	matrix describing linear relation between τ, F and y_0, θ_0





SUBJECT: Finite-size bob for uniform cross section flexure-based compound pendulum.

1. **Dynamics**: Speake provides the end torque and forces on the Flexure for quasi-static case, since we are exciting modes at 3 Hz we are in the quasi-static case. The cock angle of the end bob is small so there is no need to worry it will disrupt linearity or change the linear coefficients significantly.

For the elastic Euler-Bernoulli flexure Speake gives the geometry $y(x)$ for parameters τ, F or θ_0, y_0 equally well. We can calculate the elastic potential energy and the gravitational potential energy using this geometry. Speake's paper outlines such calculations in full. Schlamminger's writeup appears to do this correctly yielding the following.

$$V_{\text{point mass}} = \frac{1}{2W}(\frac{F}{\alpha}(-F\exp(-2L\alpha)+F(-1+L\alpha)+2\alpha\tau+2(F-\alpha\tau)\exp(-L\alpha))+\alpha\tau^2\tanh(L\alpha))$$

To avoid algebra errors we use computer algebra system Maxima (code in appendix) to calculate the analytic potential energy of the flexure in terms of its end angle θ_0 and end displacement y_0 by integrating the vertical displacement and elastic potential energy, then substituting in y_0 and θ_0 coordinates according to Speake's compliance matrix. We find $V = Ay_0^2 + B\theta_0^2 + C\theta_0 y_0$ since the angle is small. Once we find this we write it in translational center of mass coordinates, $V(y_0', \theta_0)$ for the end mass by substituting $y_0' = y_0 + \theta_0 L'/2$. Then to include the extension of the COM from the end of the flexure we include extra gravitational potential energy $V_{\text{finite size}} = V_{\text{point mass}} + \frac{1}{2}\theta_0^2 W \frac{L'}{2}$ for the point mass mounted on a flexure. With this we have found pure force terms of the Lagrangian.

The kinetic energy is for our purposes $T = \frac{1}{2}m(\frac{dy_0'}{dt})^2 + \frac{1}{2}(I_{\text{cm}} + m\frac{L'^2}{4})(\frac{d\theta_0}{dt})^2$. Since the rotation is not about the CM there is a cross term, it is proportional to $\phi\frac{d\theta_0}{dt}\frac{dy_0'}{dt}$, ϕ being the characteristic asymmetry angle of the paddle geometry in the plane of rotation. For the pliable flexure axis motion ϕ is small and so is the cross term. When the paddle is wide (stiff flexure axis motion) ϕ is negligible since the paddle is mirror symmetric making the cross term zero. There is also the vertical motion, but this has negligible contribution to the kinetic energy. The vertical displacement is $O(^2)$ and so does appear in the potential energy, but this makes the vertical kinetic energy contribution $O(^4)$.

$r_{\text{eff}}\theta \approx y$ characterizes the eigenmode by its lever arm. An ideal pivot mounted stiff-arm pendulum would have one mode with $r_{\text{eff}} = L$ in the small-angle regime.

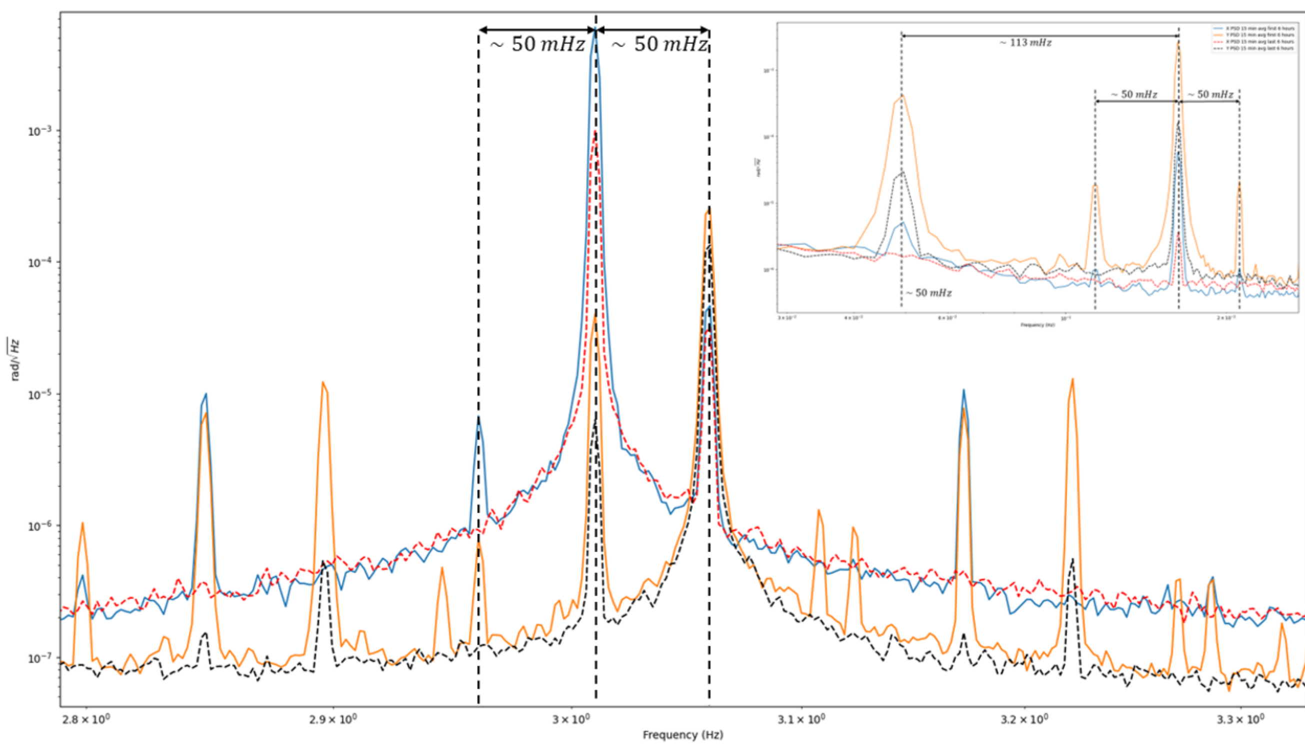
With a quadratic Lagrangian in hand we find the characteristic frequencies by eigenvalue decomposition of the equations of motion. We used maxima computer algebra to find A, B, C since the flexure is of uniform cross section. If the flexure were not uniform we could use a numerical simulation and fit parameters A, B, C.

We provide the analytically calculated (with computer algebra) fundamental frequencies for the simple paddle pendulum above.

SUBJECT: Finite-size bob for uniform cross section flexure-based compound pendulum.

Bending dir. of pendulum:	Stiff	Pliable
m	0.03728 g	0.03728 g
W	0.00037 N	0.00037 N
L	25 mm	25 mm
L'	5 mm	5 mm
I_{end}	5.1×10^{-10} kg m^2	3.1×10^{-10} kg m^2
E	280 GPa	280 GPa
I	2.3×10^{-21} m^4	1.2×10^{-23} m^4
α	747 m^{-1}	10368 m^{-1}
$r_{\text{eff}-}$	0.0232 m	0.0245 m
$\frac{1}{2\pi}\omega_-$	3.06 Hz	3.01 Hz
$\frac{1}{2\pi}\sqrt{\frac{g}{L+\frac{L'}{2}}}$	3.00 Hz	3.00 Hz
$r_{\text{eff}+}$	-5.9×10^{-4} m	-3.4×10^{-4} m
$\frac{1}{2\pi}\omega_+$	9.17 Hz	9.35 Hz
$\frac{1}{2\pi}\sqrt{\frac{3g}{2L'}}$	8.62 Hz	8.62 Hz

Quad Detector: 2025/05/23 to 27 data




```
/* stiff flexure frequencies */
/* cannonical coordinate y and flexure end point y' are not the y*/
/*Lets find the shape parameters of flexure in terms of cannonical
   coordinates*/
F: W^2/((L)*a-2)*(a/W*(y - t * Lp/2)-t/W);
T: W^2/((L)*a - 2)*(-(y - t * Lp/2)/W+(L/W-1/a/W)*t);

/*Shape curve of flexure*/
Y: F/a/W*(tanh(a*L)*(cosh(a*x)-1)+a*x-sinh(a*x))+T/W/cosh(a*L)*(cosh
   (a*x) - 1);
```

SUBJECT: Finite-size bob for uniform cross section flexure-based compound pendulum.

```
dydx: diff(Y,x);
d2ydx2:diff(dydx,x);
/*Calculate energies of the elastic flexure in terms of cannonical
   coordinates*/
Vg: 1/2*t**2*Lp/2*W + 1/2*W*ratsimp(integrate((dydx)^2,x,0,L));
positive;
Vel: 1/2 * E * I * ratsimp(integrate((d2ydx2)^2,x,0,L));
positive;
V: ratsimp(exponentialize(Vel + Vg));

/* find RHS lagrangian*/
dVdt: diff(V, t);
dVdy: diff(V, y);


/* Simplify them */
dVdt_simp: combine(ratsimp(expand(dVdt)));
dVdy_simp: combine(ratsimp(expand(dVdy)));

/* Define numeric values */
m: 0.03728e-3; /* kg */
Lp: 0.005;     /* m */
W: 0.00037;    /* N */
Icm: 2.766e-10; /* kg·m² */
E: 280e9;      /* Pa */
I: 2.3e-21;    /*  m */
L: bfloat(0.025);      /* m */
/* We use Iend because theta moves the COM y which gives extra KE
   with thetadot which is exactly as shifting*/
Iend: (Lp/2)^2 * m + Icm;
/* Define a = alpha = sqrt(W/(E*I)) */
a: bfloat(sqrt(W/(E*I)));

/* Evaluate the simplified expressions with all numerical values */
dVdy_eval: bfloat(ev(ev(dVdy_simp,a = a), L = L));

dVdt_eval: bfloat(ev(ev(dVdt_simp,a = a), L = L));


/* Print float weightings of t and y */
display(dVdy_eval, dVdt_eval);

dVdt_t: bfloat(ev(dVdt_simp, [t=1, y=0, a=a, L=L]));
dVdt_y: bfloat(ev(dVdt_simp, [t=0, y=1, a=a, L=L]));

dVdy_t: bfloat(ev(dVdy_simp, [t=1, y=0, a=a, L=L]));
dVdy_y: bfloat(ev(dVdy_simp, [t=0, y=1, a=a, L=L]));

Force_Matrix: matrix(
    [dVdy_y, dVdy_t],
    [dVdt_y, dVdt_t]
```

```
);

M: matrix(
  [m,0],
  [0, Iend      ]
);

/* Invert M */
M_inv: invert(M);

Kmat: -1 * M_inv . Force_Matrix;

/* Compute eigenvalues */
K_eigenvals: eigenvalues(Kmat);

/* Extract the list of eigenvalues */
lambda_list: K_eigenvals[1]; /* eigenvalues() returns [ [e1,e2], [v1
   ,v2] ] */

/* Convert eigenvalues to frequencies in Hz */
frequencies_hz: map(lambda ([], bfloat(sqrt (-) / (2 * %pi))),
   lambda_list);

/* Print frequencies in Hz */
display(frequencies_hz);

/* Compute eigenvectors too */
K_eigs: eigenvectors(Kmat);

/* Extract eigenvectors - they are in K_eigs[2] as columns */
v1: K_eigs[2][1][1];
v2: K_eigs[2][2][1];


reff_plus: bfloat(v1[1] / v1[2]);
reff_minus:  bfloat(v2[1] / v2[2]);

/* Display the effective lever arms in meters */
display(reff_minus, reff_plus);
```

Listing 1: Maxima pendulum frequency calculation