Math PhD 2025: Physical Modeling Exam
Musical Stiff Strings

Michele Ducceschi

Directions:

- The submission should be a zip archive containing all the MATLAB scripts and functions, and the answer sheet (this can be a scan of a hand-written sheet. If you are going to hand-write, please use clear handwriting.)
- Absolutely no collaboration between the students is allowed! But please do get in touch with the lecturer if you have doubts.

And now, the exam:
The vibration of a thin cable is described by the D'Alembert equation, as seen in class and as explained in the lecture notes. The vanishing of the cross section is an important underlying hyopthesis in deriving the model. In some cases, such as in musical instruments, this hypothesis does not hold, and one must account for finite-thickness effects in the model. These can be easily borrowed from the theory of bars, as seen in class. Strings of this kind are referred to as *stiff* (where the stiffness is here intended as a rigidity impeding the string to flex under the application of a flexural force).

A model for such strings is therefore given as:

$$\rho A \frac{\partial^2 y}{\partial t^2} = T_0 \frac{\partial^2 y}{\partial x^2} - EI \frac{\partial^4 y}{\partial x^4}. \tag{1}$$

For now, losses are not considered, and the system is zero-input. Here, $y = y(x,t) : \mathcal{D} \times \mathbb{R}_0^+ \to \mathbb{R}$ is the flexural (i.e., vertical) displacement of the cable; $x \in \mathcal{D} := [0, L]$ is the spatial coordinate defined on the closed interval; $t \geq 0$ is time. Constants appear as: ρ, the volume density of the cable; $A = \pi r^2$, the area of the cross section, with r being the radius; T_0, the applied tension at the cable's ends; E, Young's modulus; I, the area moment of inertia. We are considering cylindrical cables only, for which $I = \pi r^4/4$. Reference values for these parameters may be used as:

- $L = 0.67$ m
- $\rho = 8000 \text{ kg} \cdot \text{m}^{-3}$
- $r = 0.35$ mm
- $T_0 = 100$ N
- $E = 2 \cdot 10^{11}$ Pa

Answer Sheet Question 1. Divide both members of (1) by ρA, and obtain an equation of the form:

$$\frac{\partial^2 y}{\partial t^2} = c_T^2 \frac{\partial^2 y}{\partial x^2} - c_L^2 \kappa^2 \frac{\partial^4 y}{\partial x^4}, \tag{2}$$

depending on three constants: c_T is the "D'Alembert" wave speed; c_L is the speed of sound for the longitudinal wave in bars; κ is the radius of gyration. Give the expressions for all the three constants. Which constant depends on *geometric parameters* only? Which constant depends on *material parameters* only? Which constant can be made smaller or larger depending on the applied tension?

It is now time to look at the boundary conditions. For that, you will use *energy analysis*.

Answer Sheet Question 2. Multiply both sides of (2) by $\frac{\partial y}{\partial t}$, and integrate the resulting equation between $x = 0$ and $x = L$. Perform all the necessary integration by parts, and show that the following energy balance holds:

$$\frac{dH}{dt} = \mathcal{B}\Big|_0^L, \tag{3}$$

with:

$$H = \int_0^L \left(\frac{1}{2}\left(\frac{\partial y}{\partial t}\right)^2 + \frac{c_T^2}{2}\left(\frac{\partial y}{\partial x}\right)^2 + \frac{c_L^2\kappa^2}{2}\left(\frac{\partial^2 y}{\partial x^2}\right)^2 \right) \mathrm{d}x, \tag{4}$$

and where $\mathcal{B}$ is a boundary term to be evaluated at the endpoints of the string. From the form of H, give the forms of the kinetic energy and the potential energy. Do these forms have units of energy (i.e. Joules)? If not, multiply the expressions by an appropriate constant so that the energy components have units of energy. Furthermore, give a general expression for $\mathcal{B}$. Show that the boundary term is written as:

$$\mathcal{B} := \frac{\partial y}{\partial t}\mathcal{F} + \frac{\partial^2 y}{\partial t \partial x}\mathcal{M}, \tag{5}$$

where $\mathcal{F}$ and $\mathcal{M}$ are the boundary *force* and *moment*. From there, extract the following boundary conditions:

$$\text{clamped: } y = 0,\ \frac{\partial y}{\partial x} = 0; \qquad \text{simply-supported: } y = 0,\ \mathcal{M} = 0; \qquad \text{free: } \mathcal{F} = 0,\ \mathcal{M} = 0. \tag{6}$$

All throughout, give expressions for $\mathcal{F}$ and $\mathcal{M}$.

Equation (2) is now discretised using finite differences. Start with the spatial component. Consider a number $M := L/\Delta_x$ of grid intervals, with grid spacing Δ_x. Then, define the state vector $\mathbf{y}(t) := [y_0(t), y_1(t), ..., y_M(t)]^\intercal$. A semi-discrete version of (2) is given as:

$$\ddot{\mathbf{y}}(t) = \left(\underbrace{c_T^2 \mathbf{D}_{xx} - c_L^2 \kappa^2 \mathbf{D}_{xxxx}}_{\mathbf{B}} \right) \mathbf{y}(t) := \mathbf{B}\mathbf{y}(t) \tag{7}$$

where $\mathbf{D}_{xx}$, $\mathbf{D}_{xxxx}$ are suitable approximation of the laplacian and biharmonic operators, respectively. The operator **B** incorporates the linear combination of the laplacian and the biharmonic, and we will consider it from now on.

Answer Sheet Question 3. Discretise the continuous boundary conditions (8) using *centred* difference operators. For the left endpoint, these are given explictly as:

$$\text{clamped: } y_0 = 0,\ \delta_{x\cdot} y_0 = 0; \qquad \text{simply-supported: } y_0 = 0,\ \delta_{xx} y_0 = 0; \qquad \text{free: } ? = 0,\ ? = 0. \tag{8}$$

Give suitable expressions for the free boundary conditions at the left end-point. (Note that these are not uniquely defined. Many options are possible, but if you can, use second-order accurate conditions ...). After doing this, compute the explicit expression of the linear operator **B** near the endpoints, under {simply-supported, simply-supported} boundary conditions are the left and right endpoints. *Hint: Remember that, when one endpoint is* fixed*, there is no need to store or update the endpoint, effectively shortening the length of* **y**. *For example, for the fully clamped case, the length of* **y** *is* $M-1$ *since* $y_0 = y_M = 0$. *These points need not be stored or updated. One must still specialise the operator* **B** *at* y_1 *and* y_{M-1}. When $M = 5$, write the explicit expressions of the operator **B** with the appropriate size. Use the symbolic expressions for $c_T, c_L, \kappa, \Delta_x$, and not their numerical values.

Matlab Task 1. Write a Matlab *function*, called `B_build.m`, to compute the numerical expression of the operator **B** under simply-supported boundary conditions. The function should take the following numerical input parameters: `M,L,c_T,c_L,kappa`, and return the matrix `B` with the appropriate dimensions. Your function should be appropriately commented.

Bounds on the eigenvalues of the matrix are important to get to a stability condition for the time-stepping scheme, as will be shown below.

Matlab Task 2. Write a Matlab *script*, called `eigentest.m`, to compute the numerical eigenvalues of the matrix **B** as constructed from the task above. Test out the following cases: $M = 10, 30, 50, 100$. The user selectable inputs should be: `L,c_T,c_L,kappa`. The script should:

- have a for loop with a counter ranging from 1 to 4 for the four cases $M = 10, 30, 50, 100$
- for each case, compute *all* the eigenvalues of **B**
- sort all the eigenvalues in ascending order
- plot the sorted eigenvalues in four subplots, to be stacked vertically, where the x axis is the eigenvalue number and the y axis is the eigenvalue. Each subplot should also contain a *horizontal line* at $-\frac{4c_T^2}{\Delta_x^2} - \frac{16c_L^2\kappa^2}{\Delta_x^4}$
- print on screen the variables `M`, `largest_eigen`, `smallest_eigen` corresponding to: number of subintervals, largest eigenvalue, smallest eigenvalue.

You can now discretise in time the string equation in time. For that, the vector $\mathbf{y}(t)$ defined in continuous time is approximated via $\mathbf{y}^n$, at the time $t_n = nk$, where $n \in \mathbb{N}$ is the time index, and where $k = 1/f_s$ is the time step, computed as the inverse of the sample rate f_s. In what follows, use $f_s = 48 \cdot 10^3$ (audio rate). Then, use:

$$\delta_{tt}\mathbf{y}^n = \mathbf{B}\mathbf{y}^n. \tag{9}$$

Answer Sheet Question 4. Write out the update for scheme (9). Is it an *implicit* (that is, requiring the inversion of a linear system) or an *explicit* scheme? Using the results from the previous block, derive a *stability condition* for scheme (9). To that end, left-multiply the scheme by $\delta_{t\cdot}(\mathbf{y}^n)^\intercal$, where $^\intercal$ is the transpose operator. It is possible to show, using vector equivalents of the discrete-time energy identities seen in class, that the following energy conservation property exists in the discrete case:

$$\delta_{t+}\mathfrak{h}^{n-1/2} = 0, \qquad \text{with } \mathfrak{h}^{n-1/2} := \frac{1}{2}(\delta_{t-}\mathbf{y}^n)^\intercal(\delta_{t-}\mathbf{y}^n) - \frac{1}{2}(e_{t-}\mathbf{y}^n)^\intercal\mathbf{B}\,\mathbf{y}^n \tag{10}$$

Note that this is true only if the matrix $\mathbf{B}$ is symmetric for the chosen boundary conditions. Check this! Then, use the following identity:

$$(e_{t-}\mathbf{y}^n)^\intercal\mathbf{B}\,\mathbf{y}^n = (\mu_{t-}\mathbf{y}^n)^\intercal\mathbf{B}\mu_{t-}\mathbf{y}^n - \frac{k^2}{4}(\delta_{t-}\mathbf{y}^n)^\intercal\mathbf{B}\delta_{t-}\mathbf{y}^n \tag{11}$$

and note that the energy components (kinetic and potential) will be non-negative if and only if the eigenvalues of the matrix $\mathbf{I} + \frac{k^2}{4}\mathbf{B}$ are non-negative. Then, using the results from the previous block, conclude that this conditions is:

$$1 - \frac{k^2}{4}\left(\frac{4c_T^2}{\Delta_x^2} + \frac{16c_L^2\kappa^2}{\Delta_x^4}\right) \geq 0. \tag{12}$$

Solve this inequality for Δ_x: this is in the form $\Delta_x \geq \Delta_{x,min}(k)$ and give the number M of grid subintervals for the string with the parameters given on the first page, $\Delta_x = \Delta_{x,min}$, using the sample rate given above.

Answer Sheet Question 5. Using an eigenvalue procedure, find six values for c_T such that the fundamental frequency of the string is: 82 Hz, 110 Hz, 147 Hz, 196 Hz, 247 Hz, and 330 Hz. These are the standard frequencies for "open" guitar strings. Rememer that the fundamental radian frequency ω_1 is given approximately by:

$$\omega_1^2 \approx -\lambda_{min}(\mathbf{B}), \tag{13}$$

where λ_{min} is the eigenvalue with the smallest *absolute value* of the matrix $\mathbf{B}$. You should find the values of c_T by trial and error, adjusting its value until the condition above is approximately met. In the answer sheet, just list the six values of c_T.

You can now add losses and forcing in the system. Starting with the continuous equation, a model for the lossy, forced string is given as:

$$\frac{\partial^2 y}{\partial t^2} = c_T^2\frac{\partial^2 y}{\partial x^2} - c_L^2\kappa^2\frac{\partial^4 y}{\partial x^4} - 2\sigma_0\frac{\partial y}{\partial t} + \delta(x - x_f)f(t), \tag{14}$$

where σ_0 is a loss parameter, $\delta(x - x_f)$ is a Dirac delta applied at x_f, and where $f(t)$ is an external forcing. The external forcing is in the form:

$$f(t) = r_c(t)(1 + bn(t)), \tag{15}$$

where $0 \leq b \leq 1$, and where $n(t)$ is a random sequence in $[-1, 1]$. Furthermore,

$$r_c(t) = \frac{F_0}{2}\left(1 - \cos\frac{2\pi t}{t_w}\right) \tag{16}$$

for $0 \leq t \leq t_w$, and is zero otherwise. The input function depends on two parameters, t_w and F_0, controlling, respectively, the contact duration in time and the largest forcing amplitude.

Answer Sheet Question 6. Using as your template the scheme given in the Tutorial uploaded to GitHub, write a finite difference scheme discretising (14). What are the units of $f(t)$? As per usual, many options exist for discretising (14). Probably, the easiest option is discretising the temporal losses using the centred time operator $\delta_{t\cdot}$. This choice will not affect the stability condition already computed for the conservative case (*why*?).

Matlab Task 3. Using the same template, write a matlab script called `openguitarstring.m` generating the sound of the six guitar strings. The preamble should look like this:

```
%-------------------------------------------------------------------------
%                      Physical Modelling
%                    Dr Michele Ducceschi
%                    University of Bologna
%                         09-06-2025
%-------------------------------------------------------------------------



clear all
close all
clc

%-------------------------------------------------------------------------
% custom parameters
fs      = 48000 ;         %-- sample rate [Hz]
T       = 5 ;             %-- total sim time [s]
L       = 0.67 ;          %-- string length [m]
cL      = ... ;           %-- wave speed [m/s]
kappa   = ... ;           %-- radius of gyration [m]
stringN = 1 ;             %-- string to pluck (1,2,3,4,5,6)
```

```
%-- forcing parameters (modulated raised cosine)
%-- these are example values, user can change these
tw       = 0.01 ;        %-- contact duration [s]
F0       = 0.25e4 ;      %-- amplitude
xf       = 0.93 ;        %-- input location [frac]
b        = 0.5 ;         %-- noise modulation

%-- output parameters
%-- these are example values, user can change these
xL    = 0.34 ;           %-- left channel
xR    = 0.37 ;           %-- right channel

%-- loss parameters
%-- these are example values, user can change these
sig0   = 0.3 ;


%-- play sound
PlaySound   = 1 ;        %-- 1 play sound, 2 dont play sound
%--------------------------------------------------------------------
```

The script, as suggested, should allow the user to pluck any of the six strings. After the preamble and the custom parameters, you should have a second section building the loop matrices, calling the appropriate function (`B_build.m`). Make sure you select the grid spacing as $\Delta_{x,min}$ from the stability condition given by (12). Then, you should create the main loop, and finally play or not play the sound according to `PlaySound`: this is just a flag allowing the user to prevent Matlab from playing a sound if not needed. If you fancy, try coding up an improved version with an added loss term in the form of a frequency-dependent loss, such that the total loss is:

$$\mathcal{R}\left(\frac{\partial y}{\partial t}\right) := \left(2\sigma_0 - 2\sigma_1 \frac{\partial^2}{\partial x^2}\right)\frac{\partial y}{\partial t} \tag{17}$$

Good Luck!