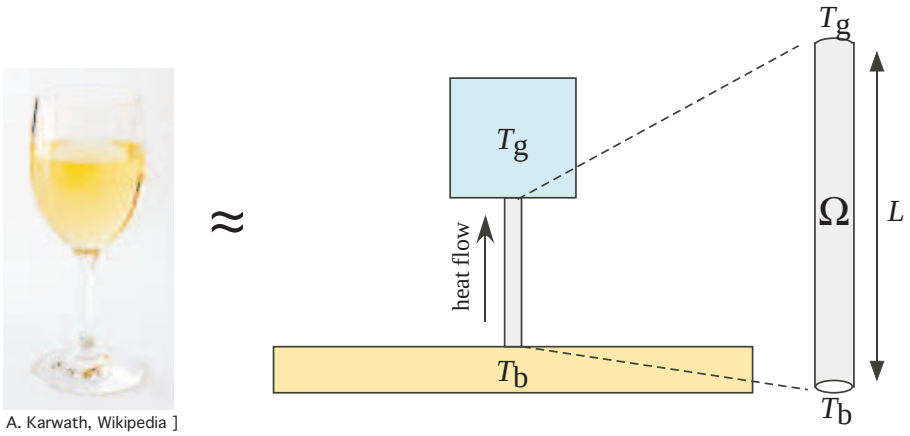

Figure 1: Schematic for problem 1: cooling of a wine glass with liquid at temperature T_g sitting on a base at temperature T_b, connected by a thin glass stem. We approximate the stem by a cylindrical domain Ω of length L with cross-sectional area a in which the top is temperature T_g and the bottom is temperature T_b, while the sides are insulated.

18.303 Problem Set 4

Due Wednesday, 5 October 2011.

Problem 1: Estimating cooling

Consider the problem of a wine glass filled with liquid at a temperature T_g, sitting on top of a base with temperature T_b, connected by a narrow stem, as depicted in fig. 1. You are going to try to estimate the rate of cooling via conduction through the stem (ignoring radiative cooling, convection in the air, etcetera). We will model the stem approximately as a 3d glass cylindrical domain Ω of length L ($z \in [0, L]$) and cross-sectional area a, in which the temperature $T(\mathbf{x}, t)$ satisfies the heat equation $\frac{\partial T}{\partial t} = \frac{\kappa}{c\rho}\nabla^2 T$ where $\kappa = 1.1$ W/m K is the thermal conductivity of the glass, $\rho = 3100$ kg/m^3 is the density, and $c = 800$ J/kg K is the specific heat capacity. Assume $L = 7$ cm and $a = 1$ cm^2. We will impose Dirichlet boundary conditions $T = T_g$ at the top ($z = L$) and $T = T_b$ at the bottom ($z = 0$), and insulating (Neumann) boundary conditions $\mathbf{n} \cdot \nabla T$ at the sides of Ω (where $\mathbf{n}$ is the outward surface normal vector).

(a) If we assume that the temperature T_g of the liquid in the glass changes much more slowly than the temperature equilibriates within the stem, then we can approximately assume steady state ($\partial T/\partial t = 0$) within the stem. Under this assumption, solve for $T(x, y, z)$ in the stem Ω. (Hint: recall pset 1, and change the boundary conditions in the z direction to $T = 0$ at the ends by writing $T = T_0 + \Delta T$ for some known function T_0.)

(b) In terms of T_g and T_b, compute the rate q of heat transport (in W) into the glass by $q = \iint \kappa \nabla T \cdot d\mathbf{S}$, where the integral is over the *top* surface of the wine stem Ω. Using this, compute the time for the liquid in the glass to warm from 8°C (ideal for a Riesling) to a ghastly 20°C assuming that $T_b = 25$°C (room temperature), assuming the temperature of the glass follows the equation $c_g \rho_g V_g \frac{dT_g}{dt} = q$, where $c_g = 4000$ J/kg K is the specific heat capacity of the liquid, $\rho_g = 1000$ kg/m^3 is its density, and $V_g = 0.00025$ m^3 (≈ 1 cup) is the volume.

(c) Was our steady-state assumption (that T in the stem reaches steady state much faster than T_g changes) justified? *Be quantitative.* (Hint: the eigenfunctions are separable into functions of xy multiplied by functions of z, and the eigenvalues give you a timescale to reach steady state. You can easily calculate the smallest-magnitude eigenvalue.)

Problem 2: Null spaces and conservation laws

In class, we showed that if $\hat{A}u = \frac{\partial u}{\partial t}$, and $v = N(A^*)$, then $\langle v, u \rangle$ is conserved quantity: $\frac{d}{dt}\langle v, u \rangle = 0$ for any solution u of the PDE. In class, we used the example of conservation of heat (or solute mass) for the heat equation (diffusion equation). Here, we will look at the consequences of a similar analysis for electromagnetism, and obtain conservation of charge density.

In vacuum with no currents, Maxwell's equations can be written $\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$, $\nabla \times \mathbf{B} = \frac{\partial \mathbf{E}}{\partial t}$, $\nabla \cdot \mathbf{E} = \rho_E$ (ρ_E = electric charge density) , $\nabla \cdot \mathbf{B} = \rho_M$ (ρ_M = magnetic monopole density, usually assumed to be zero). The first two equations express the *dynamics* of the electromagnetic fields, and can be written:

$$\hat{A}\mathbf{u} = \begin{pmatrix} & \nabla\times \\ -\nabla\times & \end{pmatrix} \begin{pmatrix} \mathbf{E} \\ \mathbf{B} \end{pmatrix} = \frac{\partial}{\partial t} \begin{pmatrix} \mathbf{E} \\ \mathbf{B} \end{pmatrix} = \frac{\partial \mathbf{u}}{\partial t},$$

where $\hat{A}$ acts on the 6-component vector field $\mathbf{u}$ containing *both* $\mathbf{E}$ and $\mathbf{B}$. Define the usual inner product $\langle \mathbf{u}, \mathbf{v} \rangle = \int_\Omega \bar{\mathbf{u}} \cdot \mathbf{v}$. Ω is a finite volume of space. Assume the following boundary conditions $\mathbf{n} \times \mathbf{E}|_{\partial\Omega} = 0$ and $\mathbf{n} \cdot \mathbf{B}|_{\partial\Omega} = 0$, where $\mathbf{n}$ is the outward normal vector (i.e., $\mathbf{E}$ is paralle to $\mathbf{n}$ and $\mathbf{B}$ is perpendicular).[1]

(a) Find $\hat{A}^*$ (with help from pset 3), and hence conclude that the eigenvalues λ of $\hat{A}$ must be ...? (Hint: recall pset 1.) Does this give oscillating electromagnetic-wavelike solutions?

(b) The charge densities ρ_E and ρ_M are given by:

$$\rho = \begin{pmatrix} \rho_E \\ \rho_M \end{pmatrix} = \begin{pmatrix} \nabla\cdot & \\ & \nabla\cdot \end{pmatrix} \begin{pmatrix} \mathbf{E} \\ \mathbf{B} \end{pmatrix} = \hat{C}\mathbf{u}.$$

Show that, assuming $\mathbf{u}$ satisfies Maxwell's equations $\hat{A}\mathbf{u} = \frac{\partial \mathbf{u}}{\partial t}$, that charge density is conserved: $\frac{\partial \rho}{\partial t} = 0$. (You should be able to show this by directly differentiating ρ; don't worry about null spaces yet.)

Another way of interpreting this is that if you have a fixed charge distribution and fields that satisfy $\nabla \cdot \mathbf{E} = \rho_E$ and $\nabla \cdot \mathbf{B} = \rho_M$, the dynamical equations by themselves won't screw up Gauss's law(s). (This does not mean that charge distributions cannot change: fields can push charges around, for example. However, we cannot get this to occur unless we include additional equations for the forces on the charges, in which case the problem becomes highly nonlinear.)

(c) Given that $N(\nabla\times) = \{\nabla\phi$ for any (sufficiently differentiable) scalar field $\phi\}$, describe $N(\hat{A}^*)$. (You may also need some boundary conditions on your scalar field(s).)

(d) From the fact that $\frac{\partial}{\partial t}\langle \mathbf{v}, \mathbf{u} \rangle = 0$ for any $\mathbf{v} \in N(\hat{A}^*)$ and any $\mathbf{u}$ satisfying $\hat{A}\mathbf{u} = \frac{\partial \mathbf{u}}{\partial t}$, (re)derive $\frac{\partial \rho}{\partial t} = 0$. (Hint: integrate by parts. Assume that if $\int \phi f = 0$ for all ϕ, then $f = 0$.[2])

Problem 3: Bunches of Bessels

In this problem, you will consider the eigenfunctions of the operator $\hat{A} = c\nabla^2$ in a 2d cylindrical domain Ω via separation of variables. In class, we attacked the $c = 1$ eigenproblem $\hat{A}u = \lambda u$ and obtained Bessel's equation and Bessel-function solutions. Although Bessel's equation has two solutions $J_m(kr)$ and $Y_m(kr)$ where $k = \sqrt{-\lambda/c}$ (the *Bessel functions*), the second solution (Y_m) blows up as $r \to 0$ and so for that problem we could only have $J_m(kr)$ solutions (although we still needed to solve a transcendental equation to obtain k).

[1] These are the boundary conditions for perfect electric conductors.

[2] Technically, this requires some further information about the space of functions f, as otherwise we could e.g. still have f nonzero at isolated points. Later in 18.303, we will make statements like this more precise in a different way by redefining the "=" in $f = 0$ to "weak equality."

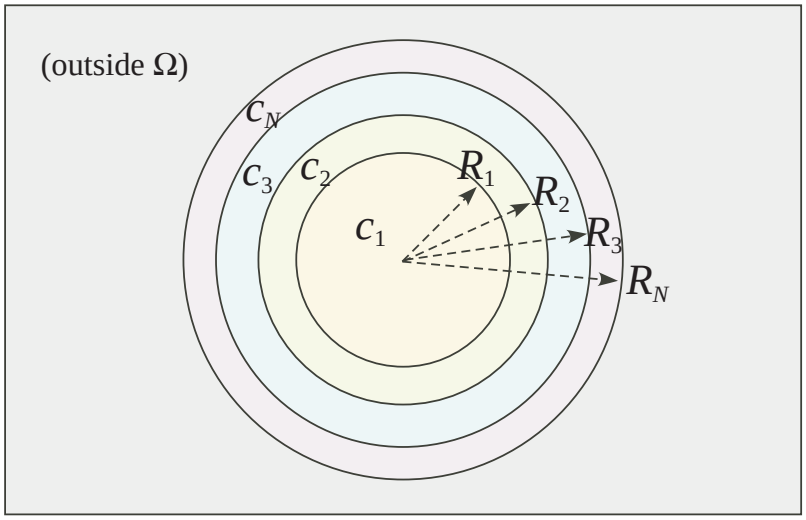

Figure 2: Schematic of cylindrical domain Ω of radius R_N containing N nested cylinders of different coefficients $c_1, \ldots, c_N$ for $\hat{A} = c(r)\nabla^2$.

In this problem, you will solve the case where c is *not* a constant. Instead, c is a piecewise-constant (real and > 0) function of r broken up into N concentric layers with interfaces at radii $R_1, \ldots, R_N$, as shown schematically in fig. 2. That is, in cylindrical (r, θ) coordinates, c is a function:

$$c(r,\theta) = \begin{cases} c_1 & 0 \le r \le R_1 \\ c_2 & R_1 < r \le R_2 \\ \vdots & \vdots \\ c_N & R_{N-1} < r \le R_N \end{cases} = c_n \text{ for } (r,\theta) \in \Omega_n,$$

where we denote each concentric layer by $\Omega_1, \ldots, \Omega_N$ for convenience. We will impose Dirichlet boundary conditions $u|_{\partial\Omega} = u|_{r=R_N} = 0$. Because this problem still has rotational symmetry, separation of variables still works: exactly as in class, we can write the eigenfunctions in the form $u(r,\theta) = \rho(r)\tau(\theta)$ where $\tau(\theta)$ is spanned by $\sin(m\theta)$ and $\cos(m\theta)$ for integers m and $\rho(r)$ is a function that satisfies Bessel's equation *in each constant-c layer*. That is, in each concentric layer Ω_n, we can write the eigenfunctions in the separable form:

$$u(r,\theta) = [\alpha_n J_m(k_n r) + \beta_n Y_m(k_n r)] \times [A\cos(m\theta) + B\sin(m\theta)],$$

where m is an integer and A and B are arbitrary constants (which are the *same* for all layers Ω_n), α_n and β_n are constants to be determined (which are *different* in each layer Ω_n), and $k_n = \sqrt{-\lambda/c_n}$ (in terms of the eigenvalue λ to be determined). Requiring u to be finite at $r = 0$ means that we must have $\boxed{\beta_1 = 0}$, while the other layers can (and generally will) have $\beta_n \neq 0$.

(a) In order for $\nabla^2 u$ to be finite, we must have u and $\partial u/\partial r$ continuous across each interface R_n. (This is why m and A and B had to be the same for all layers.) Using this fact, write down a matrix M_n (a so-called "transfer matrix") such that

$$M_n \begin{pmatrix} \alpha_n \\ \beta_n \end{pmatrix} = \begin{pmatrix} \alpha_{n+1} \\ \beta_{n+1} \end{pmatrix}$$

for any $n < N$. M_n should be in terms of J_m and Y_m and J'_m and Y'_m (and λ and the c's) at R_n. (You can leave M_n in an unsimplified form, e.g. write $M_n = C^{-1}D$ for some 2×2 matrices C and D.)

(b) Using the Dirichlet boundary conditions, write down equations for α_N and β_N in terms of

some two-component vector m_N:

$$m_N^T \begin{pmatrix} \alpha_N \\ \beta_N \end{pmatrix} = 0.$$

(c) Combine the previous two parts, and the fact that $\beta_1 = 0$, to obtain an equation

$$f_m(\lambda) = 0$$

for the eigenvalues λ at a given angular frequency m (λ which should be the *only* unknown in the equation). You can leave f_m in unsimplified form as a product of matrices or whatever.

(d) Now, write a Matlab function that implements your $f_m(\lambda)$ and plot it for $m = 0$ for the case of $N = 2$, with $R_1 = 0.5$, $R_2 = 1$, $c_1 = 5$, $c_2 = 1$. Note that Matlab provides the Bessel functions built-in: $J_m(x)$ is `besselj(m,x)` and $Y_m(x)$ is `bessely(m,x)`. You can use the identities $J_0'(x) = -J_1(x)$ and $Y_0'(x) = -Y_1(x)$ to compute their derivatives. Write your function by creating a file `f0.m` of the form:

```
function val = f0(lambdas)
val = zeros(size(lambdas));
for i = 1:length(lambdas)
    lambda = lambdas(i);
    val(i) = .....compute f_0(λ)....;
end
```

so that your function can operate an array of λ values at once (and return an array of f_0 values as its output). Note that if you wisely left your f_0 as a product of matrices or matrix inverses above, you can do so here as well: Matlab is perfectly happy to multiply matrices for you with `*`. You can create a matrix $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ in Matlab via `M=[a,b;c,d]`, and you can invert a matrix with `inv(M)`. Note also that you can create whatever local variables you want inside the loop; e.g. it might be useful to set `k1=sqrt(-lambda/5); k2=sqrt(-lambda/1)`.

Once you have defined your function and saved the file, plot it with `fplot(@f0, [-xxx,0])`, where "`xxx`" is some value big enough for you to see at least 3 zeros of f_0.

(e) Solve for the smallest-magnitude three eigenvalues λ for $m = 0$. You can use the Matlab `fzero` function to help you find zeros, once you know their approximate locations from your plot in the previous part. e.g. to find the zero closest to -1, you could use `fzero(@f0, -1)`.

(f) For the eigenvalues in the previous part, plot the corresponding eigenfunctions $u(r, \theta)$ versus r (noting that they are independent of θ for $m = 0$. Normalize them to $\alpha_1 = 1$. It might be easiest to create a little function u0(x) by writing a file u0.m of the form:

```
function val = u0(lambda, r)
val = zeros(size(r));
for i = 1:length(r)
    if r(i) < 0.5
        val(i) = ....compute u(r) in Ω_1 from α_1 = 1, β_1 = 0
    else
        val(i) = ....compute u(r) in Ω_2 from α_2, β_2 via M_1
    end
end
```

and then you can plot it via `fplot(@(r) u0(λ,r), [0,1])` for each of your λ values. (Check that u and $\partial u/\partial r$ are continuous!)