CAAM 336 · DIFFERENTIAL EQUATIONS

Homework 9 · Solutions

Posted Wednesday 12 November, 2014. Due 5pm Wednesday 19 November, 2014.

*Please write your name and **residential college** on your homework.*

1. [40 points: 10 points each]

(a) Consider the function $u_0(x) = \begin{cases} 1, & x \in [0, 1/3]; \\ 0, & x \in (1/3, 2/3); \\ 1, & x \in [2/3, 1]. \end{cases}$

Recall that the eigenvalues of the operator $L : C_N^2[0,1] \to C[0,1]$,

$$Lu = -u''$$

are $\lambda_n = n^2\pi^2$ for $n = 0, 1, \ldots$ with associated (normalized) eigenfunctions $\psi_0(x) = 1$ and

$$\psi_n(x) = \sqrt{2}\cos(n\pi x), \qquad n = 1, 2, \ldots.$$

We wish to write $u_0(x)$ as a series of the form

$$u_0(x) = \sum_{n=0}^{\infty} a_n(0)\psi_n(x),$$

where $a_n(0) = (u_0, \psi_n)$.

Compute these inner products $a_n(0) = (u_0, \psi_n)$ by hand and simplify as much as possible.

For $m = 0, 2, 4, 80$, plot the partial sums

$$u_{0,m}(x) = \sum_{n=0}^{m} a_n(0)\psi_n(x).$$

(You may superimpose these on one single, well-labeled plot if you like.)

(b) Write down a series solution to the homogeneous heat equation

$$u_t(x,t) = u_{xx}(x,t), \qquad 0 < x < 1, \quad t \geq 0$$

with Neumann boundary conditions

$$u_x(0,t) = u_x(1,t) = 0$$

and initial condition $u(x, 0) = u_0(x)$.

Create a plot showing the solution at times $t = 0, 0.002, 0.05, 0.1$.

You will need to truncate your infinite series to show this plot.

Discuss how the number of terms you use in this infinite series affects the accuracy of your plots.

(c) Describe the behavior of your solution as $t \to \infty$.

(To do so, write down a formula for the solution in the limit $t \to \infty$.)

(d) How would you expect the solution to the inhomogeneous heat equation

$$u_t(x,t) = u_{xx} + 1, \qquad 0 < x < 1, \quad t \geq 0$$

with Neumann boundary conditions

$$u_x(0,t) = u_x(1,t) = 0$$

to behave as $t \to \infty$?

Solution.

(a) To expand $u_0(x)$ in the form

$$u_0(x) = \sum_{n=0}^{\infty} a_n(0)\psi_n(x),$$

we must compute the coefficients $a_n(0)$. For $n = 0$ we compute

$$a_0(0) = \int_0^1 u_0(x) \cdot 1 \, \mathrm{d}x = \int_0^{1/3} 1 \, \mathrm{d}x + \int_{2/3}^1 1 \, \mathrm{d}x = 2/3.$$

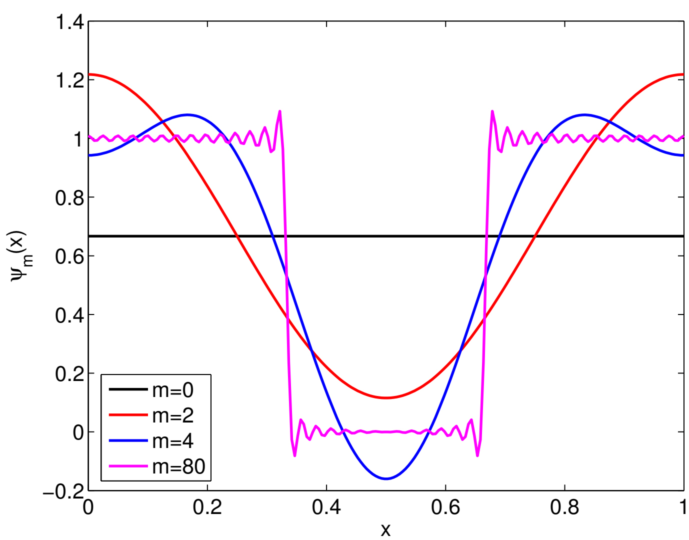
For $n > 0$ we have

$$\begin{aligned} a_n(0) &= \sqrt{2}\int_0^1 u_0(x)\cos(n\pi x)\,dx \\ &= \sqrt{2}\Big(\int_0^{1/3}\cos(n\pi x)\,dx + \int_{2/3}^1 \cos(n\pi x)\,dx\Big) \\ &= \sqrt{2}\Big(\Big[\frac{\sin(n\pi x)}{n\pi}\Big]_0^{1/3} + \Big[\frac{\sin(n\pi x)}{n\pi}\Big]_{2/3}^1 \\ &= \frac{\sqrt{2}(\sin(n\pi/3) - \sin(2n\pi/3))}{n\pi}. \end{aligned}$$

[GRADERS: this last expression is sufficiently simplified to receive full credit.]
Note that $\sin(2n\pi/3) = 2\sin(n\pi/3)\cos(n\pi/3)$, and hence

$$\sin(n\pi/3) - \sin(2n\pi/3) = \sin(n\pi/3)(1 - 2\cos(n\pi/3)).$$

Thus we have $a_n(0) = 0$ in two cases: if n is a multiple of 3, or if $\cos(n\pi/3) = 1/2$. The former occurs when $n = 3, 6, 9, 12, 15, \ldots$, whiles the latter occurs when $n\pi/3 (\mathrm{mod}\ 2\pi) = \pi/3$ or $5\pi/3$, and hence $a_n(0) = 0$ when $n = 1 + 6p$ for integers $p \geq 0$ or $n = -1 + 6p$ for integers $p \geq 1$. Together, this implies that for all odd integers n, $a_n(0) = 0$. We end up with the partial sums shown in the following figure. (MATLAB code follows at the end of this solution.)



(b) We seek a series solution of the form

$$u(x,t) = \sum_{n=0}^{\infty} a_n(t)\psi_n(x).$$

Using standard techniques described in class, together with the fact the problem is homogeneous ($f(x,t) = 0$), we find that

$$a_n'(t) + \lambda_n a_n(t) = 0.$$

For $n = 0$ we have

$$a_0'(t) = 0,$$

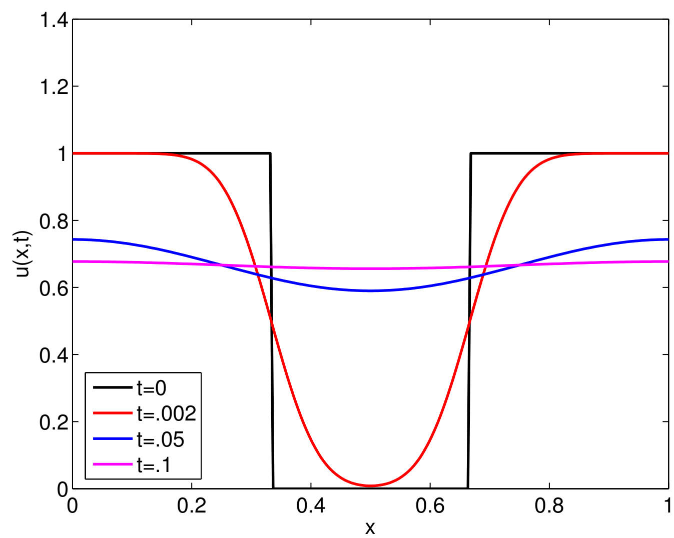
and hence $a_0(t)$ is constant, so we conclude $a_0(t) = a_0(0) = 2/3$. For $n \geq 1$ we have

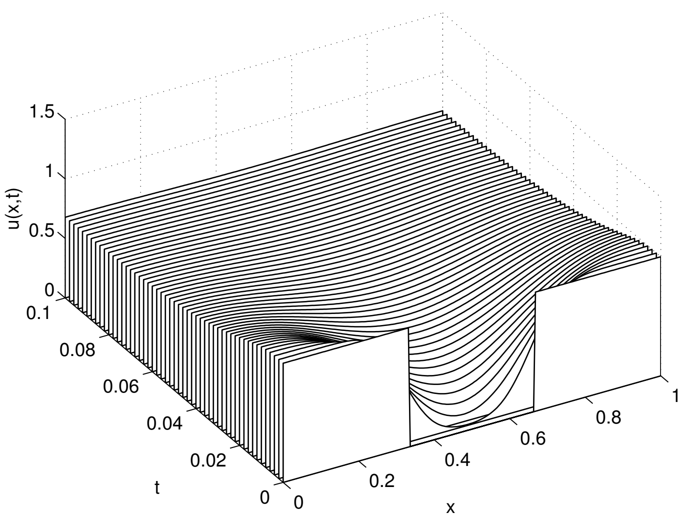
$$a_n(t) = e^{-\lambda_n t} a_n(0),$$

where $\lambda_n = n^2\pi^2$. In sum, we have

$$u(x,t) = 2/3 + \sum_{n=1}^{n} e^{-\lambda_n t} a_n(0)\big(\sqrt{2}\cos(n\pi x)\big).$$

Below we show this plot at the required times, based on taking the sum out to $N = 20$. While the number of terms in the series affects the accuracy of the solution in at early times, the importance of these extra terms decreases as $t \to \infty$.





(c) As is clear from the series formula in part (b) and from the figures, as $t \to \infty$, $u(x,t) \to 2/3$ for all $x \in [0,1]$.

(d) The existence of the limiting solution in part (c) does not contradict the fact that $\lambda_0 = 0$. There is no division by zero, as there is in the analogous steady-state problem $u_{xx} = f(x)$ with homogeneous Neumann conditions. The addition of the source term adds energy to the system, effectively increasing the rate of change of temperature with respect to time (u_t) by one unit. This corresponds to the physical situation of pumping more energy into a bar that is insulated at both ends—and hence energy cannot escape. Thus we expect the heat to grow as $t \to \infty$.

The above paragraph is satisfactory for full credit, but we can actually be quite a bit more precise. The eigenvalue $\lambda_0 = 0$ contributes a constant term to the solution of the PDE $u_t = u_{xx}$, and this constant will be nonzero provided $(u_0, \psi_0) = \int_0^1 u_0(x) \cdot 1\, dx \neq 0$. If u_0 has 'zero mean', i.e., $\int_0^1 u_0(x)\, dx = 0$, then the solution to the homogeneous problem will decay as $t \to \infty$; otherwise, as $t \to \infty$ the solution will approach the nonzero constant (u_0, ψ_0).

To write down the solution to the general inhomogeneous equation $u_t = u_{xx} + f$, we must expand

$$f(x,t) = \sum_{n=0}^{\infty} c_n(t)\psi_n(x).$$

The the coefficients $a_n(t)$ in the expansion of the solution

$$u(x,t) = \sum_{n=0}^{\infty} a_n(t)\psi_n(x)$$

obey the differential equation

$$a_n'(t) = -\lambda_n a_n(t) + c_n(t).$$

As seen in class, these ODEs have the solutions

$$a_n(t) = e^{-\lambda_n t} a_n(0) + \int_0^t e^{-\lambda_n (t-\tau)} c_n(\tau)\, d\tau.$$

The $a_0(t)$ case is particularly interesting: $a_0(t) = a_0(0) + \int_0^t c_0(\tau)\, d\tau$. Hence we cannot possibly have a steady state solution if $c_0(\tau)$ is bounded away from zero for all $\tau > 0$.

In the case of $f(x,t) = 1$, we have $c_0(t) = 1$ and $c_n(t) = 0$ for $n > 0$, so that

$$a_0(t) = a_0(0) + \int_0^t 1\, d\tau = a_0(0) + t;$$

and for $n > 0$,

$$a_n(t) = e^{-\lambda_n t} a_n(0),$$

thus giving the solution

$$u(x,t) = a_0(0) + t + \sum_{n=1}^{\infty} e^{-\lambda_n t} a_n(0)\psi_n(x).$$

```
% Plot the expansion of the initial data, psi(x)

 x = linspace(0,1,200);
 col = 'krbm';
 figure(1), clf
 fm = zeros(size(x));
 for n=0:2:80
    if n==0, an0 = 2/3;                          % psi(x) = 1 for x in [0,1/3], [2/3,1];
    else, an0 = sqrt(2)*(sin(n*pi/3)-sin(2*n*pi/3))/(n*pi);% psi(x) = 0 otherwise.
    end
```

```
      if n==0, fm = an0*ones(size(fm));
      else, fm = fm + an0*sqrt(2)*cos(n*pi*x);
      end
      if ismember(n,[ 0 2 4 80]),
         plot(x, fm, '-','linewidth',2,'color',col(1)), hold on, col = col(2:end);
      end
   end
   legend('m=0', 'm=2','m=4','m=80',3)
   set(gca,'fontsize',16)
   xlabel('x'), ylabel('\psi_m(x)')
   print -depsc2 heateqn1

% Compute the solution at at various times.

   psi = (x <= 1/3) | (x >= 2/3);    %  initial condition
   U = [psi];
   col = 'krbmc';
   figure(2), clf
   plot(x, psi, 'linewidth',2,'color',col(1)), hold on, col = col(2:end);
   t = .002:.002:0.1;
   tprint = [.002 .05 0.1];
   for j=1:length(t)
      for n=0:2:20
         if n==0,
            an0    = 2/3;
            lambda = 0;
            uj     = exp(-lambda*t(j))*an0*ones(size(x));
         else
            an0 = sqrt(2)*(sin(n*pi/3)-sin(2*n*pi/3))/(n*pi);
            lambda = n^2*pi^2;
            uj = uj + exp(-lambda*t(j))*an0*(sqrt(2)*cos(n*pi*x));
         end
      end
      U = [U;uj];
      if ismember(t(j),tprint),
         plot(x, uj, '-','linewidth',2,'color',col(1)), hold on, col = col(2:end);
      end
   end
   legend('t=0','t=.002','t=.05','t=.1',3)
   set(gca,'fontsize',16)
   xlabel('x'), ylabel('u(x,t)')
   print -depsc2 heateqn2

   figure(3), clf
   plt = waterfall(x,[0 t],U);
   set(plt,'edgecolor','k')       % make the lines black

   view(-30,50)
   set(gca,'fontsize',14)
   xlabel('x'), ylabel('t'), zlabel('u(x,t)')
   print -depsc2 heateqn3
```

---

2. [30 points: 10 points each]
   Consider the *fourth order* partial differential equation

$$u_t(x,t) = u_{xx}(x,t) - u_{xxxx}(x,t)$$

with so-called *hinged* boundary conditions

$$u(0,t) = u_{xx}(0,t) = u(1,t) = u_{xx}(1,t) = 0$$

and initial condition (that should satisfy the boundary conditions)

$$u(x,0) = u_0(x).$$

(This equation is related to a model that arises in the study of thin films.)

To solve this PDE, we introduce the linear operator $L : C_H^4[0,1] \to C[0,1]$, where

$$Lu = -u'' + u''''$$

and

$$C_H^4[0,1] = \{u \in C^4[0,1], u(0) = u''(0) = u(1) = u''(1) = 0\}$$

is the set of C^4 functions that satisfy the hinged boundary conditions.

(a) The operator L has eigenfunctions

$$\psi_n(x) = \sqrt{2}\sin(n\pi x), \qquad n = 1, 2, \ldots.$$

Use this fact to compute a formula for the eigenvalues λ_n, $n = 1, 2, \ldots$.

(b) Suppose the initial condition $u_0(x)$ is expanded in the form

$$u_0(x) = \sum_{n=1}^{\infty} a_n(0)\psi_n(x).$$

Briefly describe how one can write the solution to the PDE $u_t = u_{xx} - u_{xxxx}$ as an infinite sum.

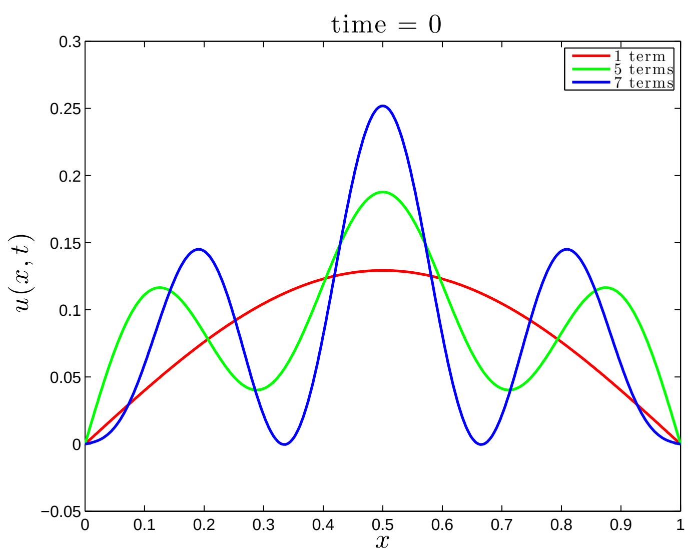
(c) Suppose the initial data is given by

$$u_0(x) = (x - x^2)\sin(3\pi x)^2,$$

with associated coefficients

$$a_n(0) = \begin{cases} \dfrac{432\sqrt{2}(n^4 - 18n^2 + 216)}{(36n - n^3)^3\pi^3}, & n \text{ odd;} \\ 0, & n \text{ even.} \end{cases}$$

Write a program (you may modify your earlier codes) to compute the solution you describe in part (b) up to seven terms in the infinite sum. At each time $t = 0$; 10^{-5}; 2×10^{-5}; 4×10^{-5}, produce a plot comparing the sum of the first 1, 5, and 7 terms of the series. For example, at time $t = 0$, your plot should appear as shown below. (Alternatively, you can produce attractive 3-dimensional plots over the time interval $t \in [0, 4 \times 10^{-5}]$ using 1, 5, and 7 terms in the series.)



Solution.

(a) Given the eigenfunctions ψ_n, we simply apply L to ψ_n to compute $\lambda_n\psi_n$:

$$\begin{aligned} L\psi_n(x) &= -\psi_n''(x) + \psi_n''''(x) \\ &= -\frac{d^2}{dx^2}(\sqrt{2}\sin(n\pi x)) + \frac{d^4}{dx^4}(\sqrt{2}\sin(n\pi x)) \\ &= n^2\pi^2\sqrt{2}\sin(n\pi x) + n^4\pi^4\sqrt{2}\sin(n\pi x) \\ &= (n^2\pi^2 + n^4\pi^4)(\sqrt{2}\sin(n\pi x) \\ &= \lambda_n\psi_n(x). \end{aligned}$$

Thus, we identify $\lambda_n = n^2\pi^2 + n^4\pi^4$ for $n = 1, 2, \ldots$.

(b) Following the prodedure outlined in class, we look for a solution of the form

$$u(x,t) = \sum_{n=1}^{\infty} a_n(t)\psi_n(x).$$

Substituting this equation into the differential equation, we obtain

$$\begin{aligned} \sum_{n=1}^{\infty} a_n'(t)\psi_n(x) &= \sum_{n=1}^{\infty} a_n(t)(\psi_n''(x) - \psi_n''''(x)) \\ &= \sum_{n=1}^{\infty} -\lambda_n a_n(t)\psi_n(x). \end{aligned}$$

Taking an inner product of both sides with ψ_k and using the orthonormality of the eigenfunctions, we obtain the scalar differential equations

$$a_k'(t) = -\lambda_k a_k(t),$$

which has the solution

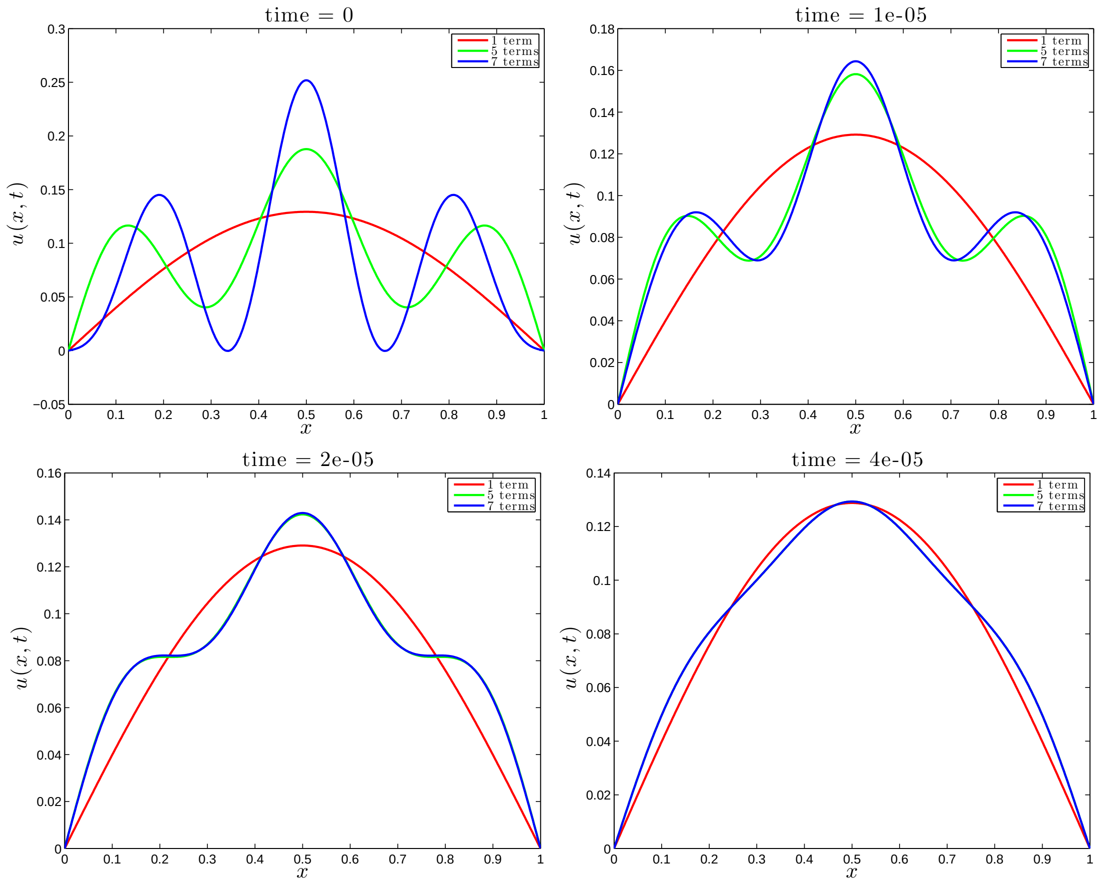
$$a_k(t) = e^{-\lambda_k} a_k(0).$$

Thus, the solution can be written in the series

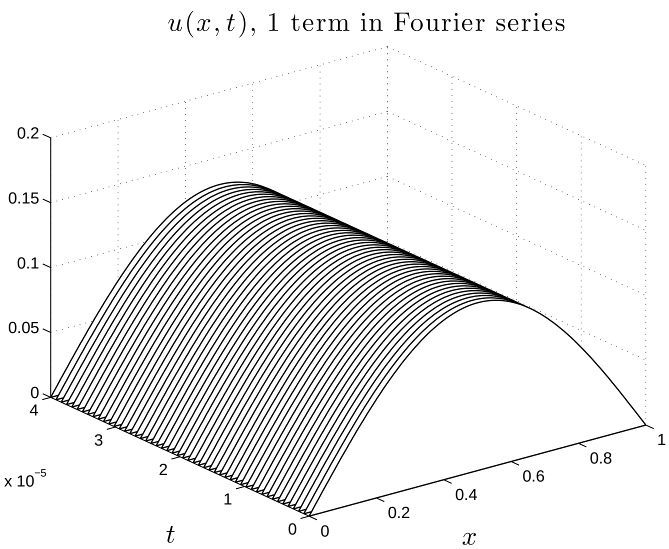
$$\begin{aligned} u(x,t) &= \sum_{n=1}^{\infty} e^{-\lambda_n t} a_n(0)\psi_n(x) \\ &= \sum_{n=1}^{\infty} \sqrt{2}e^{-(n^2\pi^2+n^4\pi^4)t} a_n(0)\sin(n\pi x). \end{aligned}$$

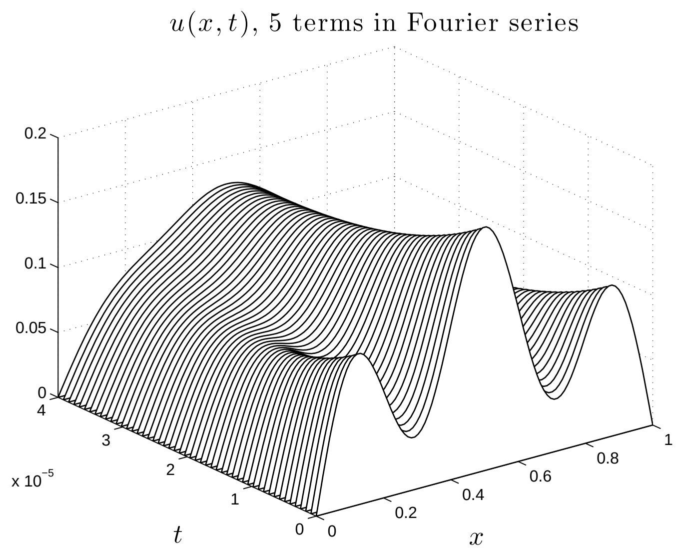
[GRADERS: students need only write down one of these series solutions for $u(x,t)$; they need not include the derivation.]

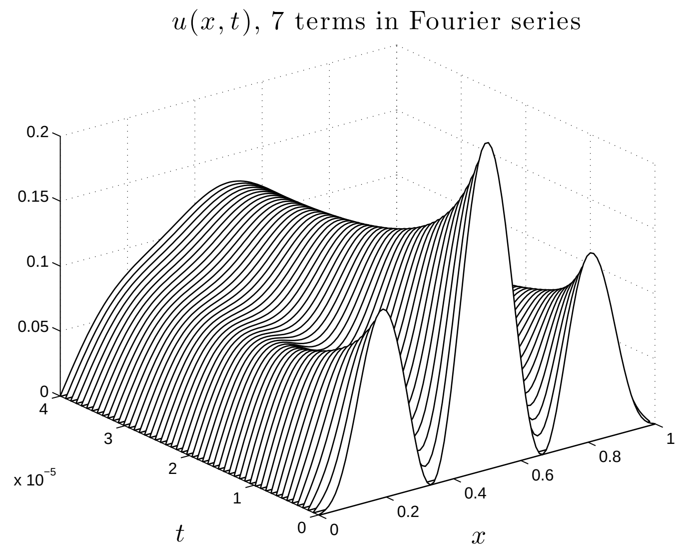
(c) Plots for the four requested times are shown below.



Alternatively, students may produce three-dimensional plots over the same time span for 1, 5, and 7 terms in the Fourier series.







One can produce these plots with the following code.

```
tvec = [0 .00001 .00002 .00004];
x = linspace(0,1,500);
an0 = inline('sqrt(2)*432*(n^4-18*n^2+216)/((36*n-n^3)^3*pi^3)');
lam = inline('n^2*pi^2 + n^4*pi^4');
col = 'rgb';
str = 'abcd';
for j=1:length(tvec)
   figure(1), clf
   t = tvec(j);
   u = zeros(size(x));
   for n=1:2:7
       u = u+exp(-lam(n)*t)*an0(n)*(sqrt(2)*sin(n*pi*x));
       [tf,loc] = ismember(n,[1 5 7]);
       if tf,
          plot(x,u,'-','color',col(loc),'linewidth',2), hold on
       end
   end
   legend('1 term','5 terms', '7 terms')
   xlabel('x','fontsize',20)
   ylabel('u(x,t)','fontsize',20)
   title(sprintf('time = %g',t),'fontsize',20)
   eval(sprintf('print -depsc2 fourth_%s',str(j)))
   pause(.1)
end

% surface plot
tvec = linspace(0, .00004, 50);
x    = linspace(0, 1, 100);
U = zeros(length(tvec),length(x),3);
for j=1:length(tvec)
   t = tvec(j);
   u = zeros(size(x));
   for n=1:2:7
      u = u+exp(-lam(n)*t)*an0(n)*(sqrt(2)*sin(n*pi*x));
      [tf,loc] = ismember(n,[1 5 7]);
      if tf, U(j,:,loc) = u; end
   end
end
figure(1), clf
plt=waterfall(x,tvec,U(:,:,1));
set(plt,'edgecolor','k')
axis([0 1 0 .00004 0 .2])
xlabel('x','fontsize',20), ylabel('t','fontsize',20)
```

```
title('u(x,t), 1 term in Fourier series','fontsize',20)
print -depsc2 fourth_wf1

figure(1), clf
plt=waterfall(x,tvec,U(:,:,2));
set(plt,'edgecolor','k')
axis([0 1 0 .00004 0 .2])
xlabel('x','fontsize',20), ylabel('t','fontsize',20)
title('u(x,t), 5 terms in Fourier series','fontsize',20)
print -depsc2 fourth_wf5

figure(1), clf
plt=waterfall(x,tvec,U(:,:,3));
set(plt,'edgecolor','k')
axis([0 1 0 .00004 0 .2])
xlabel('x','fontsize',20), ylabel('t','fontsize',20)
title('u(x,t), 7 terms in Fourier series','fontsize',20)
print -depsc2 fourth_wf7
```

---

3. [30 points: 15 points each]
We wish to approximate the solution to the heat equation

$$u_t(x,t) = u_{xx}(x,t) + 100tx, \qquad 0 \le x \le 1,\ t \ge 0$$

with homogeneous Dirichlet boundary conditions

$$u(0,t) = u(1,t) = 0$$

and initial condition

$$u(x,0) = 0$$

using the finite element method (method of lines). Let $N \ge 1$, $h = 1/(N+1)$, and $x_k = kh$ for $k = 0, \ldots, N+1$. We shall construct approximations using the hat functions

$$\phi_k(x) = \begin{cases} (x - x_{k-1})/h, & x \in [x_{k-1}, x_k); \\ (x_{k+1} - x)/h, & x \in [x_k, x_{k+1}); \\ 0, & \text{otherwise.} \end{cases}$$

The approximate solution shall have the form

$$u_N(x,t) = \sum_{k=1}^{N} a_k(t)\phi_k(x).$$

(a) Write down the system of ordinary differential equations that determines the coefficients $a_k(t)$, $k = 1, \ldots, N$. Specify the entries in the mass and stiffness matrices and the load vector. (You may use results from previous homeworks and class as convenient.)

(b) Write a MATLAB code that uses the backward Euler method to solve for the coefficients $a_k(t)$. Plot your approximate solution $u_N(x,t)$ at time $t = 1$. Choose N and Δt so that your solution appears to be accurate. Verify this accuracy by superimposing on your plot the computed solution at $t = 1$ obtained by using space and time steps that are ten times smaller.

Solution.

(a) As derived in class, the system of ordinary differential equations that governs the behavior of the coefficients $a_k(t)$ is given by

$$\mathbf{M}\frac{d\mathbf{a}}{dt}(t) + \mathbf{K}\mathbf{a}(t) = \mathbf{f}(t),$$

where $\mathbf{M}$ and $\mathbf{K}$ are the *mass* and *stiffness* matrices whose (j,k) entries are given by (ϕ_j,ϕ_k) and $a(\phi_j,\phi_k)$. We also have When the basis functions $\{\phi_j\}_{j=1}^N$ are hat functions, we have

$$\mathbf{M}_{j,k} = (\phi_j,\phi_k) = \begin{cases} 2h/3 & \text{if } j=k; \\ h/6 & \text{if } |j-k|=1; \\ 0 & \text{otherwise.} \end{cases},$$

as computed in Problem 3 of Problem Set 6, and

$$\mathbf{K}_{j,k} = a(\phi_j,\phi_k) = \begin{cases} 2/h & \text{if } j=k; \\ -1/h & \text{if } |j-k|=1; \\ 0 & \text{otherwise.} \end{cases},$$

as computed in class.

(b) The backward Euler method for this differential equation takes the form

$$\mathbf{a}_{k+1} = (\mathbf{I} + \Delta t\mathbf{M}^{-1}\mathbf{K})^{-1}(\mathbf{a}_k + \Delta t\mathbf{M}^{-1}\mathbf{f}(t_{k+1})),$$

which can be rearranged in the more computationally appealing form

$$\mathbf{a}_{k+1} = (\mathbf{M} + \Delta t\mathbf{K})^{-1}(\mathbf{M}\mathbf{a}_k + \Delta t\mathbf{f}(t_{k+1})).$$

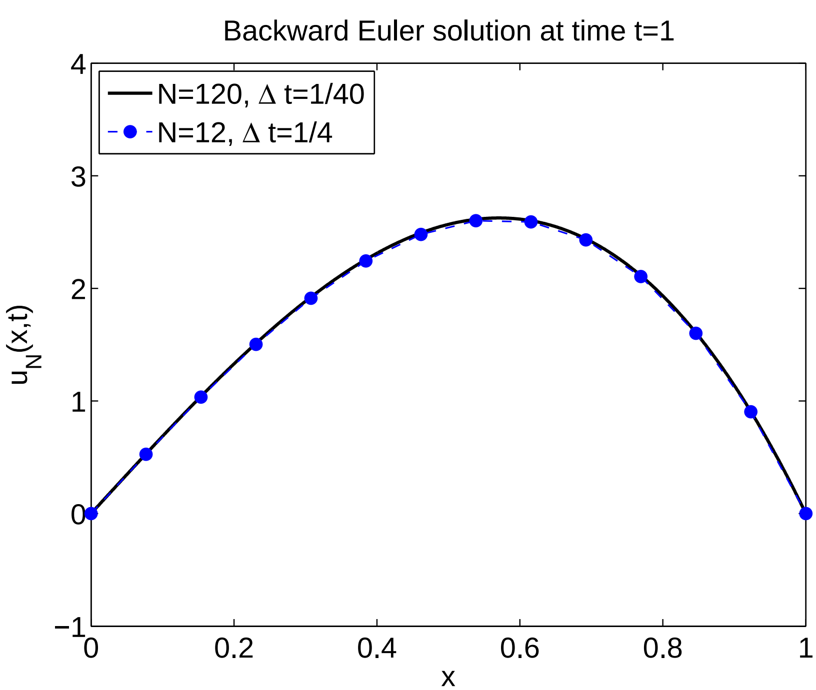
For this $f(x,t) = 100tx$, we have for fixed t that

$$\mathbf{f}_j(t) = (f(\cdot,t),\phi_j) = 100t\int_0^1 x\phi_j(x)\,dx,$$

and (as can be readily computed with Mathematica)

$$\int_0^1 x\phi_j(x)\,dx = \int_{x_{j-1}}^{x_j} \frac{x(x-x_{j-1})}{h}\,dx + \int_{x_j}^{x_{j+1}} \frac{x(x_{j+1}-x)}{h}\,dx = \Big(\frac{hx_j}{2} - \frac{h^2}{6}\Big) + \Big(\frac{hx_j}{2} + \frac{h^2}{6}\Big) = x_jh = jh^2.$$

The backward Euler method produces remarkably resilient solutions to this problem. The plot below compares the solution obtained the $N=8$ and $\Delta t = 1/2$ (dots) to that computed with $N=80$ and $\Delta t = 1/20$.



MATLAB code to solve the problem for a specified N and Δt is given below.

```
 N = input('enter N:  ');
 dt = input('enter dt: ');

 h = 1/(N+1);
 x = h*[1:N];

 M = (h/6)*(4*eye(N)+diag(ones(N-1,1),1)+diag(ones(N-1,1),-1));
 K = (1/h)*(2*eye(N)-diag(ones(N-1,1),1)-diag(ones(N-1,1),-1));

 tfinal = 1;
 xx = linspace(0,1,1000)';     % finely spaced points between 0 and 1.

 k = [1:N]';
 f = 100*h^2*k;       % (f,phi_k) = 100*exp(-t)*h^2*k:
                      % compute it once, but add the "exp(-t)" part at each step
 a0 = zeros(N,1);     % initial condition

 akp1 = a0;           % a_{k+1} will be computed in this variable
 L = chol(M)';
 for k=0:tfinal/dt-1
    t = (k+1)*dt;
    akp1 = (M+dt*K)\(M*akp1+dt*(exp(-t)*f));  % compute solution at t_{k+1}
 end

% plot the solution on a fine grid.
 uN = zeros(size(xx));
 for j=1:N
   uN = uN + akp1(j)*hat(xx,j,N);
 end
 figure(1), clf
 plot(xx,uN,'b-','linewidth',2)
 axis([0 1 -1.25 7])
 set(gca,'fontsize',18)
 xlabel('x')
 ylabel('u_N(x,t)')
 title(sprintf('time = %f', t))
```