Introduction to Numerical Simulation (Fall 2003)
Problem Set #1 - Solutions

1 Problem 1.1

a) Following traditional nodal analysis, we write the following system for nodal voltages:

$$\begin{cases} \dfrac{V_1}{R_1} + \dfrac{V_1 - V_2}{R_3} = 0 \\ \dfrac{V_2 - V_1}{R_3} + \dfrac{V_2}{R_2} = -I_s \end{cases} \tag{1}$$

Now we can rewrite this in the matrix form:

$$\begin{bmatrix} \frac{1}{R_1} + \frac{1}{R_3} & -\frac{1}{R_3} \\ -\frac{1}{R_3} & \frac{1}{R_2} + \frac{1}{R_3} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -I_s \end{bmatrix} \tag{2}$$

b) First, we need to assign currents to the branches of the circuit, one of the choices which we pick is shown on fig. 1.

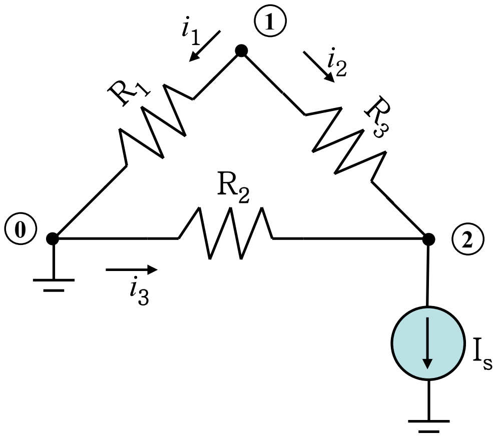

Figure 1: Our choice of current flows

Following the stamping procedure described in class, we assemble the matrices A and α:

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & -1 & -1 \end{bmatrix} \tag{3}$$

Here rows denote nodes 0 and 1, columns corresponds to branches.

$$\alpha = \begin{bmatrix} \frac{1}{R_1} & 0 & 0 \\ 0 & \frac{1}{R_3} & 0 \\ 0 & 0 & \frac{1}{R_2} \end{bmatrix} \tag{4}$$

Now we follow the lecture 2 and form the linear system of equations:

$$\begin{bmatrix} I & -\alpha A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} I \\ V \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -\frac{1}{R_1} & 0 \\ 0 & 1 & 0 & -\frac{1}{R_3} & \frac{1}{R_3} \\ 0 & 0 & 1 & 0 & \frac{1}{R_2} \\ 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -I_s \end{bmatrix} \tag{5}$$

First three equations describe constitutive relations for currents and voltages through resistors, and the last pair of equations describe conservation of current.

2 Problem 1.2

The main idea behind the method is

1. inserting a voltage source between a node and ground gives the voltage at the node, and eliminates one unknown from the system of equations,
2. because the number of variables is reduced by one, there is a redundant equation which has to be removed.

To implement the idea, the matrix-vector product is interpreted as a linear combination of columns in the matrix:

$$\begin{aligned} Gx &= [\mathbf{G_1} \dots \mathbf{G_i} \dots \mathbf{G_n}][x_1 \dots x_i \dots x_n]^T \\ &= \mathbf{G_1} x_1 + \cdots + \mathbf{G_i} x_i + \cdots + \mathbf{G_n} x_n. \end{aligned}$$

If x_i is know, $Gx = b$ can be written as

$$\begin{aligned} &\mathbf{G_1} x_1 + \cdots + \mathbf{G_i} 0 + \cdots + \mathbf{G_n} x_n = b - \mathbf{G_i} x_i \\ &\Leftrightarrow [\mathbf{G_1} \dots 0 \dots \mathbf{G_n}]\mathbf{x} = b - \mathbf{G_1} x_i. \end{aligned} \tag{6}$$

Then, we zero out the ith row because the KCL relation at the ith node is no longer necessary, and put 1 in the diagonal entry and modify the ith RHS entry to the value of the voltage source. This will make x_i equal to the assigned voltage.

One way to implement this is to add the following code to the end of *loadMatrix.m.*

```
for i=1:size(sourcenodes,2)
   if (sourcenodes(i)~=0)
      G(i,:)=0;
     b=b-sourcenodes(i)*G(:,i);
```

```
    G(:,i)=0;
    G(i,i)=1;
    b(i)=sourcenodes(i);
  end
end
```

The convention used for voltage sources and current sources in a schematic file is

- ix (arrow starts) (arrow ends) (value),
- vx (node with +) (node with -) (value).

3 Problem 1.3

a) Let's approximate the given differential equation and draw some analogy to circuits. There are more than one way to do this, and we only introduce one possibility.

First, discretize the heat conducting bar into sections as given in Fig. 2, and assume the following:

1. the temperature is constant within each control volume (divied by dashed lines),
2. the heat flow is constant in each section.

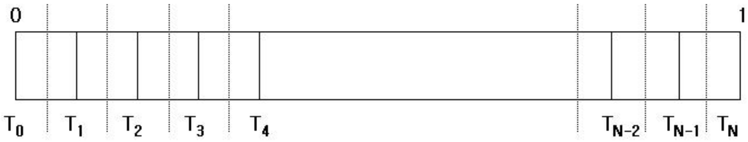

Figure 2: Discrete representation of a heat conducting bar.

Based on these assumptions, the differential equation can be approximated as

$$\begin{aligned}\frac{T_{i+1}-T_i}{\triangle x}-\frac{T_i-T_{i-1}}{\triangle x} &= \frac{\kappa_a \triangle x}{\kappa_m}(T_i-T_0)-\frac{\triangle x H_i}{\kappa_m}\\ &= \frac{T_i}{\kappa_m/(\kappa_a \triangle x)}-(\frac{\triangle x H_i}{\kappa_m}+\frac{\kappa_a\ dxT_0}{\kappa_m}).\end{aligned} \tag{7}$$

Considering temperatures as voltages in a circuit, the above equation, which models the conservation law, can be considered as a KCL relation at a node. Then the corresponding circuit looks like the one in Figs. 3, and the values for the elements are

$$\begin{aligned}R_1 &\equiv \triangle x,\\ R_2 &\equiv \frac{\kappa_m}{\kappa_a \triangle x},\\ I_1 &\equiv \frac{\triangle x H_i}{\kappa_m}+\frac{\kappa_a \triangle x T_0}{\kappa_m}.\end{aligned} \tag{8}$$

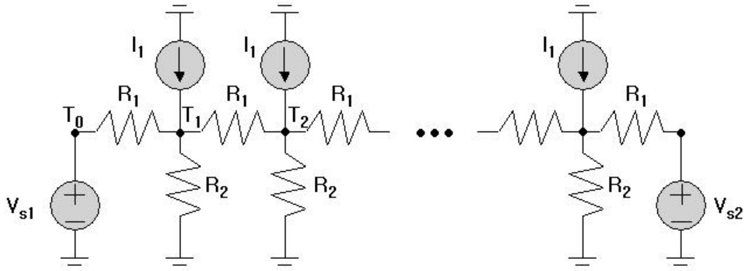

Figure 3: Circuit representation of the heat conducting bar problem.

Due to the analogy that your argument is based on, temperature corresponds to voltage, and the Dirichlet boundary condition should be modeled by a voltage source. Hence we have voltage sources at the ends.

Another way to discretize the problem, which seems to be more intuitive, is

$$\frac{T_{i+1}-T_i}{\triangle x/\kappa_m}-\frac{T_i-T_{i-1}}{\triangle x/\kappa_m}=\frac{T_i-T_0}{1/(\kappa_a\triangle x)}-\triangle x H_i, \tag{9}$$

which corresponds to Figs. 4 with

$$\begin{aligned} R_1 &\equiv \frac{\triangle x}{\kappa_m} \\ R_2 &\equiv \frac{1}{\triangle x\kappa_a} \\ V_{T0} &\equiv T_0 \\ I_1 &\equiv -\triangle x H_i \end{aligned} \tag{10}$$

The temperature distribution is in Figs. 5, and the script used to generate the circuit is

```
numseg=100;
ka=0.001;
km=0.1;
h=50;
T0=350;

dx=1/numseg;
numnode=numseg+1;

r1=dx/km;
r2=1/dx/ka;
i1=dx*h;
```

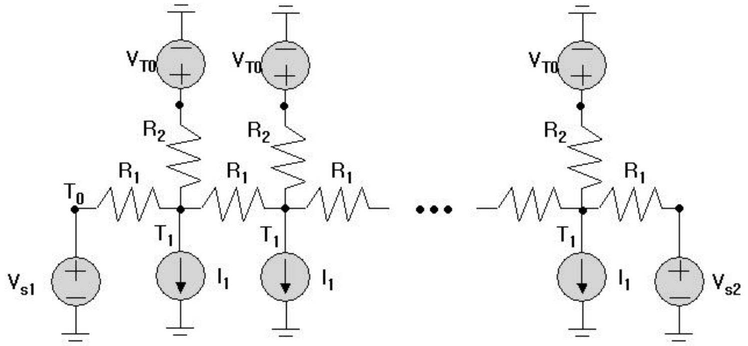

Figure 4: Circuit representation of the heat conducting bar problem.

```
fid=fopen('thermal.ckt','w');

fprintf(fid, 'v1 1 0 300\n');
fprintf(fid, 'v\%d \%d 0 300\n', numnode, numnode);
for i=2:numnode-1
   fprintf(fid, 'r\%d \%d \%d \%f\n', i-1, i-1, i, r1);
   fprintf(fid, 'r\%d \%d \%d \%f\n', i+numnode-2, i, i+numnode-1, r2);
   fprintf(fid, 'v\%d \%d \%d \%f\n', i, i+numnode-1, 0, T0);
   fprintf(fid, 'i\%d \%d \%d \%f\n', i, 0, i, i1);
end
   fprintf(fid, 'r\%d \%d \%d \%f\n', numnode-1, numnode-1, numnode, r1);

fclose(fid)
```

b) Because the heat flux, or the change in the temperature, corresponds to currents in the circuit analogy, Neumann boundary conditions are modeled by current sources. But, zero current source means that the branch is an open circuit, and simply disconnecting the the last two nodes from ground is enough instead of usingzero current sources.

The temperature distribution is given is Figs. 6. Now, we try to justify quantitatively why we get constant. One simple way to think of it is that all the nodes are in the same condition: connected to the same current sources and the same resistors. Or, refer to Figs. 3. If we assume that T_2 is higher than T_1, there should be some current flowing through R_1. But, since there is no path from T_0 to ground and no additional current can flow though the current source, all the current should go through R_2 which increase T_1. And this process continues until the T_1 is the same as T_2.

c) As one can see from Figs. 6, the temperature increases as κ_a decreases. From Figs. 4, one can see that small κ_a corresponds to large R_2, which means that the temperature is more sensitive to the change in the current flowing through R_2.

If κ_a becomes 0, R_2 is infinity which corresponding to an open circuit, and there is no

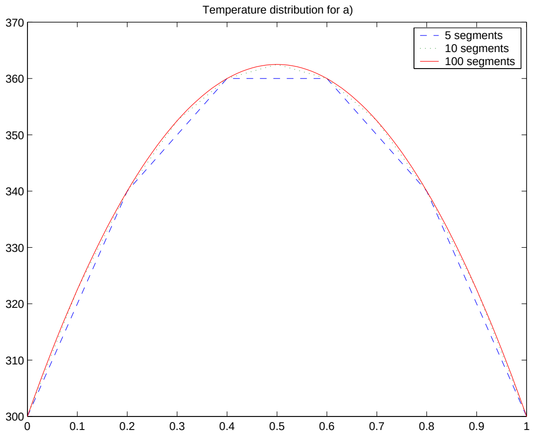

Figure 5: Temperature distribution on the bar for problem 1.3 a).

way for the current to escape from the bar. In other words, the conservation law cannot be met and there is no solution.

4 Problem 1.4

a) First of all, using the assumptions of this problem we can easily calculate the eigenfunctions analytically, just by plugging $T(x) = sin(\alpha x)$ you will get $H(x)$ which causes this distribution, to be sine function as well. The choice for α is goverened by our boundary conditions, giving us a discrete (though still infinite) spectrum for this eigenvalue problem. Only one solution, however, is physically realizable, because the value of T may be only positive (note, however, that if we leave the T_0 to be some nonzero temperature, we could set new variable $\hat{T} = T - T_0$ and we would get the same equation for $\hat{T}$ as for T in case of zero ambient temperature, but since T_0 is nonzero, $\hat{T}$ may be negative, therefore for this problem we would have had infinitely many physical solutions!).

Now we can use our simulator and create a circuit according to our conditions, as in fig. 8.

After we get the matrix using the *loadMatrix.m* function, we need to issue the following MATLAB command:

```
[V,D] = eig(G);
```

The columns of the matrix D will now contain eigenvectors of G, i.e. the temperature/heat distributions which satisfy

$$Gv_i = \hat{\lambda} v_i \tag{11}$$

In terminology of the problem 1.4 a), $\lambda = \frac{1}{\hat{\lambda}}$. The eigenvector $\hat{\lambda_i}$, which corresponds to the eigenvalue v_i is simply a diagonal element of matrix D: $\hat{\lambda_i} = d_i i$. This is what we need.

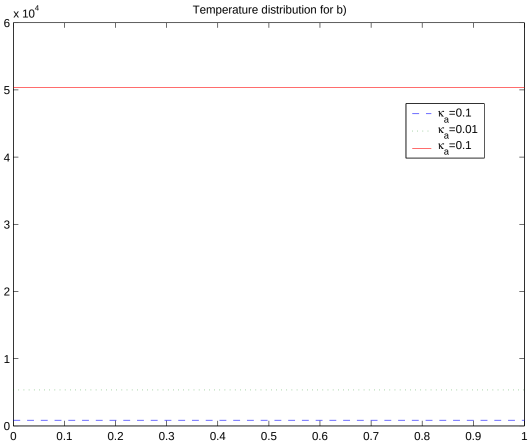

Figure 6: Temperature distribution on the bar for problem 1.3 b).

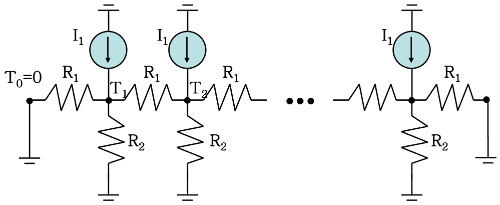

Figure 7: The circuit which implements discretization of the bar for the eigenvalue problem 1.4.

However, the eigenvectors (i.e. temperature distributions) are defined up to multiplication on a scalar. Therefore we need to find a vector from V, which corresponds to the lowest mode (which has the same sign for all nodes). In our experience this was simply the first column of matrix V. Then you need to issue the following command:

```
the_distribution = V(:,1)./max(V(:,1));
```

This normalizes our heat distribution to be unity at the maximum. We scaled our distribution.

b) For the discretized problem we will have exactly n solutions with different eigenvalues. They all correspond to the modes of initial system, however a general logic can suggest that the higher order modes will be represented not exactly, because higher modes will have length of oscillations comparable to the discretization step, whereas our discretized model may approximate the system in the limit where all characteristic lengths are much larger than Δx. In fact, the distribution for the part a) simply overlaps with sine function (I did not plot the sine solution, because simply there is no difference).

c) From the differential equation we can see that setting $\kappa_a = 0$ doesn't change the eigenvalue problem, except the values of λ_i will change, the shapes of the temperature distributions will not change. This must be the case for the discretized PDE as well.

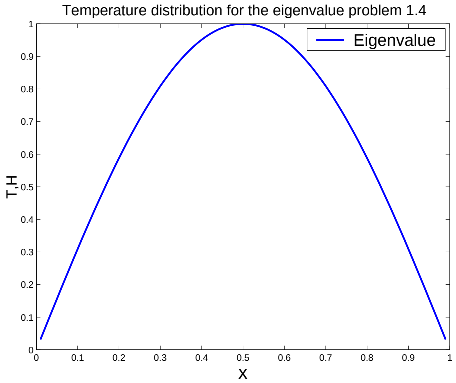

Figure 8: Temperature distribution on the bar for eigenvalue problem 1.4.