EXAMPLE 3.8-1: Transient Response of a Bent-beam Actuator

The steady-state behavior of a micro-scale, lithographically fabricated device referred to as a bent-beam actuator was investigated in EXAMPLE 1.7-1. This example examines the transient response of the bent-beam actuator. A V-shaped structure (the bent-beam in Figure 1) is suspended between two pillars. The entire beam is initially at T_{ini} = 20°C when, at time t = 0, a voltage difference is applied between the pillars causing current, I = 10 mA, to flow through the bent-beam structure. The volumetric generation of thermal energy associatd with ohmic heating leads to a thermally induced expansion of both legs that causes the apex of the bent-beam to move outwards.

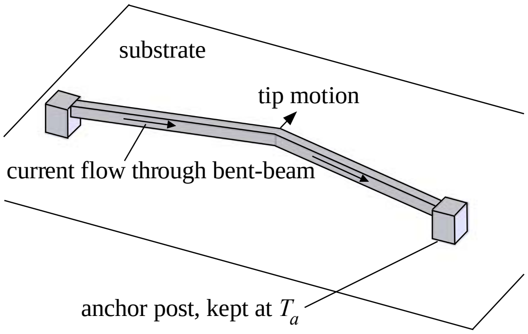

Figure 1: Bent-beam actuator

The bent beam actuator considered here is identical to that in EXAMPLE 1.7-1. The anchors of the bent-beam actuator are placed L_a = 1 mm apart and the beam structure has a cross-section of W = 10 μm by th = 5 μm. The slope of the beams (with respect to a line connecting the two pillars) is θ = 0.5 rad, as shown in Figure 2. The bent-beam material has conductivity k = 80 W/m-K, electrical resistivity ρ_e = 1x10^{-5} ohm-m, density ρ = 2300 kg/m^3, specific heat capacity c = 700 J/kg-K, and coefficient of thermal expansion CTE = 3.5x10^{-5} K^{-1}. Radiation from the beam surface can be neglected. All of the thermal energy that is generated in the beam is either convected to the surrounding air at temperature T_∞ = 20ºC with average heat transfer coefficient $\overline{h}$ = 100 W/m^2-K or transferred conductively to the pillars that are maintained at T_a = 20 ºC.

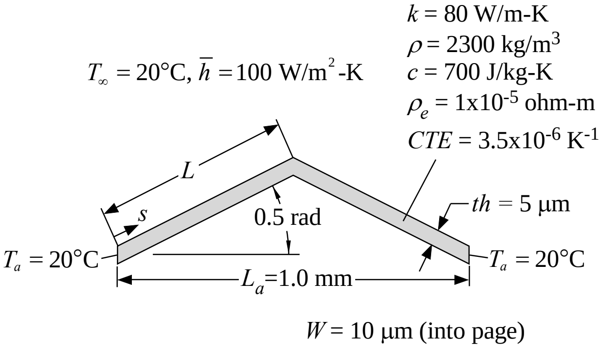

Figure 2: Dimensions and conditions associated with bent-beam actuator.

In EXAMPLE 1.7-1, an appropriate Biot number was used to show that the bent beam can be treated as an extended surface. Therefore, the temperature varies significantly only along the beam (i.e., in the s direction in Figure 2).

a.) Develop a 1-D, transient simulation that predicts the time response of the bent beam.

The known information is entered in EES:

```
"EXAMPLE 3.8-1: Transient Response of a Bent-beam Actuator"
$UnitSystem SI MASS RAD PA K J
$Tabstops 0.2 0.4 0.6 3.5 in

"Inputs"
L_a=1 [mm]*convert(mm,m)                                 "distance between anchors"
w=10 [micron]*convert(micron,m)                          "width of beam"
th=5 [micron]*convert(micron,m)                          "thickness of beam"
Current=0.010 [Amp]                                      "current"
theta=0.5 [rad]                                          "slope of beam"
T_a=converttemp(C,K,20 [C])                              "temperature of pillars"
T_ini=converttemp(C,K,20 [C])                            "initial temperature of the beam"
T_infinity=converttemp(C,K,20 [C])                       "temperature of air"
h_bar=100 [W/m^2-K]                                      "heat transfer coefficient"
k=80 [W/m-K]                                             "conductivity"
rho_e=1e-5 [ohm-m]                                       "electrical resistivity"
CTE=3.5e-6 [1/K]                                         "coefficient of thermal expansion"
c=700 [J/kg-K]                                           "specific heat capacity"
rho=2300 [kg/m^3]                                        "density"
```

A half-symmetry (around the apex of the V-shape) numerical model of the bent-beam is developed with nodes positioned as shown in Figure 3.

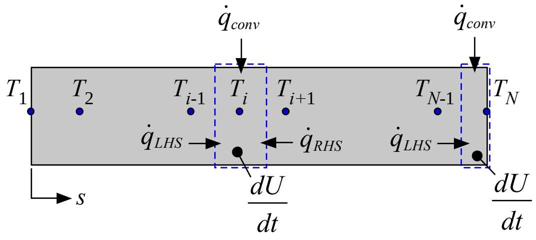

Figure 3: Nodes placed in a half-symmetry model of the bent-beam.

The length of each leg of the beam structure (L in Figure 2) is:

$$L = \frac{L_a}{2\cos\left(\theta\right)}$$

The position of each node is given by:

$$s_i = \frac{\left(i-1\right)}{\left(N-1\right)} L \quad \text{for } i = 1..N$$

and the distance between adjacent nodes is:

$$\Delta s = \frac{L}{\left(N-1\right)}$$

```
L=L_a/(2*cos(theta))                          "length of one side of the beam"
N=6 [-]                                       "number of nodes"
DELTAs=L/(N-1)                                "distance between adjacent nodes"
duplicate i=1,N
    s[i]=L*(i-1)/(N-1)                        "position of each node"
    s_mm[i]=s[i]*convert(m,mm)                "in mm"
end
```

The equations that govern the behavior of each node must be obtained using energy balances. The control volume for an arbitrary internal node *i* (shown in Figure 3) leads to the energy balance:

$$\dot{q}_{LHS} + \dot{q}_{RHS} + \dot{g} + \dot{q}_{conv} = \frac{dU}{dt}$$

Each of the terms must be approximated. The conduction terms are:

$$\dot{q}_{LHS} = k\frac{w\,th}{\Delta s}\left(T_{i-1} - T_i\right)$$

$$\dot{q}_{RHS} = k\frac{w\,th}{\Delta s}\left(T_{i+1} - T_i\right)$$

The convection term is written as:

$$\dot{q}_{conv} = 2\left(w + th\right)\Delta s\,\overline{h}\left(T_\infty - T_i\right)$$

The generation term is:

$$\dot{g} = \rho_e \frac{\Delta s}{w\,th} I^2$$

The energy storage term is:

$$\frac{dU}{dt} = w\;th\,\Delta s\,\rho\,c\frac{dT_i}{dt}$$

These equations are combined to obtain:

$$k\frac{w\,th}{\Delta s}\left(T_{i-1}-T_i\right)+k\frac{w\,th}{\Delta s}\left(T_{i+1}-T_i\right)+\rho_e\frac{\Delta s}{w\,th}I^2+2\left(w+th\right)\Delta s\,\overline{h}\left(T_\infty-T_i\right)=w\,th\,\Delta s\,\rho\,c\frac{dT_i}{dt}$$
$$\text{for } i=2...\left(N-1\right) \tag{1}$$

Equation (1) can be solved for the rate of temperature change for the internal nodes:

$$\frac{dT_i}{dt}=\frac{k}{\Delta s^2\,\rho\,c}\left(T_{i-1}+T_{i+1}-2T_i\right)+\frac{\rho_e}{w^2\,th^2\rho\,c}I^2+\frac{2\left(w+th\right)\overline{h}}{w\,th\,\rho\,c}\left(T_\infty-T_i\right) \tag{2}$$
$$\text{for } i=2...\left(N-1\right)$$

Node 1 is always maintained at T_a. Therefore, it is not necessary to carry out an energy balance on the control volume associated with node 1.

$$\frac{dT_1}{dt}=0 \tag{3}$$

Node N is the half-node at the apex of the beam (i.e., at $s = L$); the energy balance for the control volume associated with node N (see Figure 3) leads to:

$$\dot{q}_{LHS}+\dot{g}+\dot{q}_{conv}=\frac{dU}{dt}$$

Notice that there is no $\dot{q}_{RHS}$ because the right side of node N is adiabatic according to the assumption of symmetry. The terms are approximated according to:

$$\dot{q}_{LHS}=k\frac{w\,th}{\Delta s}\left(T_{N-1}-T_N\right)$$

$$\dot{q}_{conv}=\left(w+th\right)\Delta s\,\overline{h}\left(T_\infty-T_N\right)$$

$$\dot{g}=\rho_e\frac{\Delta s}{2\,w\,th}I^2$$

$$\frac{dU}{dt}=w\,th\frac{\Delta s}{2}\rho\,c\frac{dT_N}{dt}$$

Notice that the convection, generation, and storage terms have all changed by a factor of 2 because node N is a half-node. Combining these equations leads to:

$$k\frac{w\,th}{\Delta s}\left(T_{N-1}-T_N\right)+\rho_e\frac{\Delta s}{2\,w\,th}I^2+\left(w+th\right)\Delta s\,\overline{h}\left(T_\infty-T_N\right)=w\,th\frac{\Delta s}{2}\rho\,c\frac{dT_N}{dt}$$

or

$$\frac{dT_N}{dt} = \frac{2\,k}{\Delta s^2\,\rho\,c}\left(T_{N-1} - T_N\right) + \frac{\rho_e}{w^2\,th^2\,\rho\,c} I^2 + \frac{2\left(w + th\right)\overline{h}}{w\,th\,\rho\,c}\left(T_\infty - T_N\right) \tag{4}$$

Equations (2) through (4) must be integrated forward in time using one of the techniques discussed in Section 3.8.2. Any of the techniques will work; here, the `Integral` command in EES is used.

Before implementing the solution, it is useful to estimate approximately how long the start up process will take in order to determine the simulation time and provide a sanity check on the solution. Transient conduction processes are characterized by the diffusive time constant (τ_{diff}, which was discussed in Section 3.3.1 in the context of a semi-infinite body) and a lumped capacitance time constant (τ_{lumped}, which was discussed in Section 3.1.3 in the context of 0-D transient problems). The fact that the bent-beam actuator cannot be treated as either a semi-infinite body or a lumped capacitance does not reduce the relevance of these time constants. The diffusive time constant is related to the amount of time required for a conduction wave to move through the material. For this problem, it is interesting to know approximately how long is required for energy to be conducted from the apex to the pillar:

$$\tau_{diff} = \frac{L^2}{\alpha}$$

The lumped time constant is related to the amount of time required for the beam material to equilibrate with the surrounding air:

$$\tau_{lumped} = C_{beam}\,R_{conv,beam}$$

where C_{beam} is the heat capacity of the beam:

$$C_{beam} = w\,th\,L\,\rho\,c$$

and $R_{conv,beam}$ is the convective resistance between the beam surface and the air:

$$R_{conv,beam} = \frac{1}{\overline{h}\,2\left(w + th\right)L}$$

These equations are entered in EES:

```
C_beam=w*th*L*rho*c                     "heat capacity of beam"
R_conv_beam=1/(2*(w+th)*L*h_bar)        "convection resistance between beam and air"
tau_lumped=C_beam*R_conv_beam           "lumped time constant"
alpha=k/(rho*c)                         "thermal diffusivity"
tau_diff=L^2/alpha                      "diffusive time constant"
```

and lead to $\tau_{diff} = 7$ ms and $\tau_{lumped} = 27$ ms. Based on this result, we can expect that the process will be completed on the order of 27 ms (probably somewhat less than this, since the beam will not be able to come into thermal equilibrium with the air due to its conductive link with the pillars). Therefore, a simulation time $t_{sim} = 25$ ms is appropriate.

```
t_sim=0.025 [s]                    "simulation duration"
```

The time rate of change for each of the nodes are calculated using Eqs. (2) through (4):

```
dTdt[1]=0 [K/s]                                          "pillar temperature never changes"
duplicate i=2,(N-1)
    dTdt[i]=k*(T[i-1]+T[i+1]-2*T[i])/(DELTAs^2*rho*c)+rho_e*Current^2/(w^2*th^2*rho*c)&
        +2*(w+th)*h_bar*(T_infinity-T[i])/(w*th*rho*c)                "internal nodes"
end
dTdt[N]=2*k*(T[N-1]-T[N])/(DELTAs^2*rho*c)+rho_e*Current^2/(w^2*th^2*rho*c)&
         +2*(w+th)*h_bar*(T_infinity-T[N])/(w*th*rho*c)               "node at apex"
```

and these are integrated forward in time using the `Integral` command:

```
duplicate i=1,N
    T[i]=T_ini+INTEGRAL(dTdt[i],time,0,t_sim)                    "integrate forward in time"
end
```

The intermediate values of the temperature of each node are stored in an integral table at 0.5 ms intervals using the `$IntegralTable` directive:

```
$IntegralTable time:0.0005, T[1..N]
```

Figure 4 illustrates the temperature at various locations along the beam as a function of time.

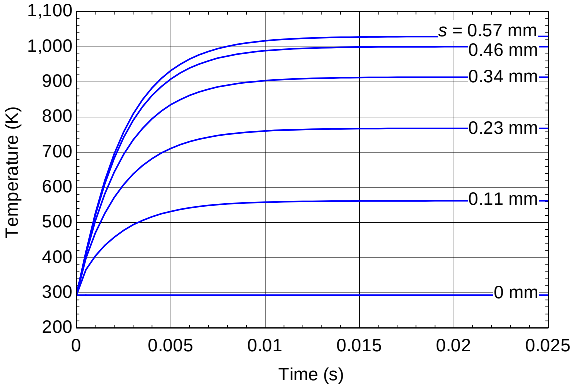

Figure 4: Temperature as a function of time at various values of position.

Notice that the solution has approached steady state after about 10 ms, which is in line with our physical intuition.

The unconstrained motion of the apex of the tip was discussed in EXAMPLE 1.7-1. The total displacement of the beam (ΔL) is obtained by integrating the differential elongation dL along the beam:

$$\Delta L = \int_0^L dL$$

where dL is related to the product of the coefficient of thermal expansion (CTE) and the temperature change:

$$dL = CTE\left(T - T_{ini}\right) ds$$

The integral can be approximated as a summation of the numerical results:

$$\Delta L_j = \sum_{i=2}^{(N-1)} CTE\left(T_{i,j} - T_a\right)\Delta s + CTE\left(T_{N,j} - T_a\right)\frac{\Delta s}{2} \quad \text{for } j = 1..M$$

where node N is treated separately because it is half the length of the internal nodes and node 1 is not included because its temperature does not rise. The summation is accomplished using the sum command in EES:

```
DELTAL=sum(CTE*(T[i]-T_ini)*Deltas,i=2,(N-1))+Deltas*CTE*(T[N]-T_ini)/2 "elongation of beam"
```

Assuming that the joint associated with the apex does not provide a torque on either leg of the beam, the displacement of the apex can be estimated using trigonometry (Figure 5).

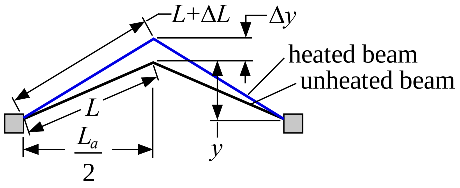

Figure 5: Trigonometry associated with apex motion

The original position of the apex (y) is given by:

$$y = \sqrt{L^2 - \left(\frac{L_a}{2}\right)^2}$$

therefore, the motion of the apex (Δy) is:

$$\Delta y = \sqrt{\left(L + \Delta L\right)^2 - \left(\frac{L_a}{2}\right)^2} - \sqrt{L^2 - \left(\frac{L_a}{2}\right)^2}$$

```
DELTAy=sqrt((L+DELTAL)^2-(L_a/2)^2)-sqrt(L^2-(L_a/2)^2)          "displacement of apex"
DELTAy_micron=DELTAy*convert(m,micron)                            "in µm"
```

The variable DELTAy_micron is added to the IntegralTable directive:

```
$IntegralTable time:0.0005, T[1..N], DELTAy_micron
```

Figure 6 illustrates the actuator motion as a function of time.

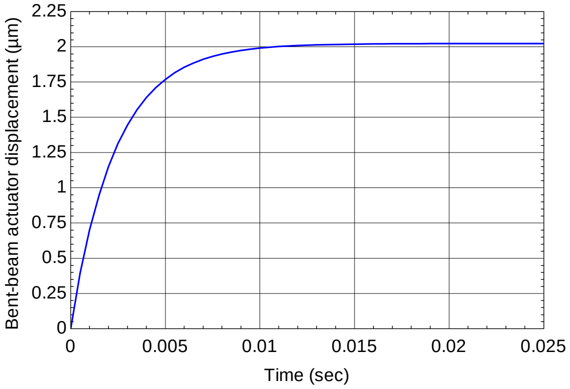

Figure 6: Actuator motion as a function of time.