EXAMPLE 2.3-1: Laser Machining

Figure 1 illustrates a ceramic disk that is th = 1 cm thick with an outer radius r_{out} = 10 cm. Ceramic can be machined more easily if it is pre-heated using a laser. Therefore, a laser beam with radius r_{laser} = 0.5 cm applies a flux of $\dot{q}''_{laser}$ = 1x10^5 W/m^2 to the center of the disk. The conductivity of the ceramic is k = 1.4 W/m-K. The disk is clamped to a machine bed at T_c = 20ºC, the contact resistance between the machine bed and the disk is R''_c = 2.8x10^{-4} m^2-K/W. The outer edge is cooled by exposure to coolant and can be considered to be at T_c = 20º. The top surface is exposed to ambient air; however, the heat transfer coefficient is so small that the surface may be modeled as being adiabatic except where it is experiencing laser heating.

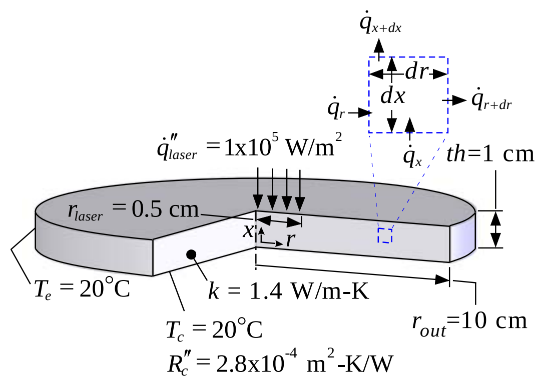

Figure 1: Ceramic disk exposed to laser pre-heating.

a.) Develop an analytical model capable of predicting the temperature in the disk.

A differential control volume is shown in Figure 1 and leads to the energy balance:

$$\dot{q}_x + \dot{q}_r = \dot{q}_{x+dx} + \dot{q}_{r+dr}$$

or

$$0 = \frac{\partial \dot{q}_x}{\partial x} dx + \frac{\partial \dot{q}_r}{\partial r} dr$$

Substituting the rate equations:

$$\dot{q}_x = -k\, 2\, \pi\, r\, dr \frac{\partial T}{\partial x}$$

and

$$\dot{q}_r = -k\, 2\, \pi\, r\, dx \frac{\partial T}{\partial r}$$

into the differential energy balance leads to:

$$0 = \frac{\partial}{\partial x}\left[-k\,2\,\pi\,r\,dr\,\frac{\partial T}{\partial x}\right]dx + \frac{\partial}{\partial r}\left[-k\,2\,\pi\,r\,dx\,\frac{\partial T}{\partial r}\right]dr$$

or

$$r\,\frac{\partial^2 T}{\partial x^2} + \frac{\partial}{\partial r}\left[r\,\frac{\partial T}{\partial r}\right] = 0$$

and the boundary conditions are:

$$\left.\frac{\partial T}{\partial r}\right|_{r=0} = 0$$

$$T_{r=r_{out}} = T_c$$

$$k\left.\frac{\partial T}{\partial x}\right|_{x=0} = \frac{1}{R''_c}\left(T_{x=0} - T_c\right)$$

$$k\left.\frac{\partial T}{\partial x}\right|_{x=th} = \begin{cases} \dot{q}''_{laser} & \text{for } 0 < r < r_{laser} \\ 0 & \text{for } r_{laser} < r < r_{out} \end{cases}$$

As stated above, neither the x or the r-directions are homogeneous; however, the boundary conditions in the x-direction can be made homogeneous by defining the transformation variable:

$$\theta = T - T_c$$

so that the governing equation becomes:

$$r\,\frac{\partial^2 \theta}{\partial x^2} + \frac{\partial}{\partial r}\left[r\,\frac{\partial \theta}{\partial r}\right] = 0 \qquad (1)$$

and the boundary conditions become:

$$\left.\frac{\partial \theta}{\partial r}\right|_{r=0} = 0 \qquad (2)$$

$$\theta_{r=r_{out}} = 0 \qquad (3)$$

$$R''_c\,k\left.\frac{\partial \theta}{\partial x}\right|_{x=0} = \theta_{x=0} \qquad (4)$$

$$k\frac{\partial\theta}{\partial x}\bigg|_{x=th} = \begin{cases} \dot{q}''_{laser} & \text{for } 0 < r < r_{laser} \\ 0 \text{ for } r_{laser} < r < r_{out} \end{cases} \tag{5}$$

Notice that the two homogeneous boundary conditions are in the r-direction and therefore we will need eigenfunctions in this dimension. The solution is expressed as the product of θX and θR which are functions only of x and r, respectively.

$$\theta(x,y) = \theta X(x)\,\theta R(r)$$

Substituting the product solution into the governing partial differential equation, Eq. (1), leads to:

$$r\,\theta R\frac{d^2\theta X}{dx^2} + \theta X\frac{d}{dr}\left[r\frac{d\theta R}{dr}\right] = 0$$

Dividing by the product $r\ \theta X\ \theta R$ leads to:

$$\underbrace{\frac{\dfrac{d^2\theta X}{dx^2}}{\theta X}}_{\pm\lambda^2} + \underbrace{\frac{\dfrac{d}{dr}\left[r\dfrac{d\theta R}{dr}\right]}{r\,\theta R}}_{\mp\lambda^2} = 0$$

The choice of the sign for the λ^2 term is important. In this problem, the eigenfunctions must be in the radial direction so we choose the sign that leads to Bessel function solutions for θR:

$$\frac{d^2\theta X}{dx^2} - \lambda^2\,\theta X = 0 \tag{6}$$

$$\frac{d}{dr}\left[r\frac{d\theta R}{dr}\right] + \lambda^2\,r\,\theta R = 0 \tag{7}$$

We will focus on solving the eigenproblem first. The solution to Eq. (7) can be determined by returning to Section 1.8.4 and following the flow chart in Figure 1-54. Maple can be used as well:

```
> restart;
> ODEr:=diff(r*diff(thetar(r),r),r)+lambda^2*r*thetar(r)=0;
```

$$ODEr := \left(\frac{d}{dr}\text{thetar}(r)\right) + r\left(\frac{d^2}{dr^2}\text{thetar}(r)\right) + \lambda^2\,r\,\text{thetar}(r) = 0$$

```
> thetars:=dsolve(ODEr);
```

$$thetars := \text{thetar}(r) = \_C1\ \text{BesselJ}(0, \lambda\,r) + \_C2\ \text{BesselY}(0, \lambda\,r)$$

Therefore:

$$\theta R = C_1 \,\text{BesselJ}\left(0,\lambda\, r\right) + C_2 \,\text{BesselY}\left(0,\lambda\, r\right) \tag{8}$$

Substituting Eq. (8) into the boundary condition at r = 0, Eq. (2), leads to:

$$C_1 \frac{d}{dr}\left[\text{BesselJ}\left(0,\lambda\, r\right)\right]_{r=0} + C_2 \frac{d}{dr}\left[\text{BesselY}\left(0,\lambda\, r\right)\right]_{r=0} = 0$$

The derivatives are evaluated using Maple:

```
> diff(BesselJ(0,lambda*r),r);
```

$$-\text{BesselJ}(1, \lambda\, r)\, \lambda$$

```
> diff(BesselY(0,lambda*r),r);
```

$$-\text{BesselY}(1, \lambda\, r)\, \lambda$$

so that:

$$-C_1\, \lambda \underbrace{\text{BesselJ}\left(1,0\right)}_{0} - C_2\, \lambda \underbrace{\text{BesselY}\left(1,0\right)}_{\to -\infty} = 0$$

Return to Section 1.8 and examine Figure 1-56, notice that BesselJ(1,0) approaches 0 while BesselY(1,0) approaches negative infinity; this information can also be obtained from Maple:

```
> limit(BesselJ(1,x),x=0);
```

$$0$$

```
> limit(BesselY(1,x),x=0);
```

$$\mathit{undefined}$$

Therefore, C_2 must be zero and θR is given by:

$$\theta R = C_1 \,\text{BesselJ}\left(0,\lambda\, r\right)$$

Substituting Eq. (8) with C_2 = 0 into the boundary condition at $r = r_{out}$, Eq. (3), leads to:

$$C_1 \,\text{BesselJ}\left(0,\lambda\, r_{out}\right) = 0 \tag{9}$$

Equation (9) is satisfied if C_1 = 0, but this would provide the solution θR = 0 which is not useful. Figure 2 shows the 0^{th} order Bessel function of the 1^{st} kind (i.e., Bessel_J(0,x)) as well as one of the more familiar eigenfunctions, cosine. Notice that BesselJ(0,x) oscillates about zero every time the argument changes by 2π in the same way that sine and cosine do; therefore, there are an infinite number of eigenvalues λ_i that will satisfy Eq. (9) associated with an infinite number of eigenfunctions. However, the eigencondition for this problem cannot be used to explicitly solve for the eigenvalues; rather, an implicit equation for the eigenvalues results from Eq. (9):

$$\text{BesselJ}\left(0,\lambda_i\, r_{out}\right)=0 \text{ where } i=1,2,\ldots\infty \tag{10}$$

and the eigenfunctions for this problem are:

$$\theta R_i = C_{1,i}\,\text{BesselJ}\left(0,\lambda_i\, r\right) \text{ where BesselJ}\left(0,\lambda_i\, b\right)=0 \text{ for } i=1,2,\ldots\infty \tag{11}$$

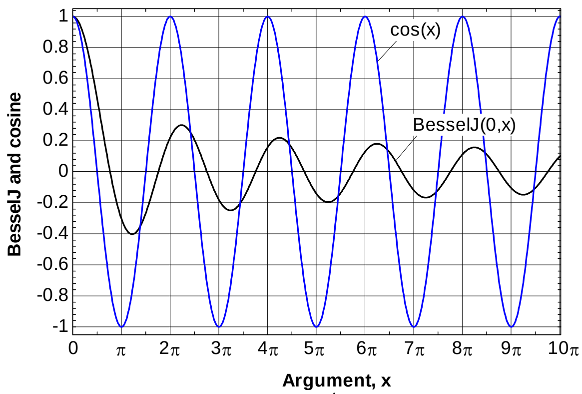

Figure 2: 0th order Bessel function of the 1st kind as well as the cosine function.

The solution to the ordinary differential equation for θX, Eq. (6), is:

$$\theta X_i = C_{3,i}\sinh\left(\lambda_i\, x\right)+C_{4,i}\cosh\left(\lambda_i\, x\right)$$

and so the general solution for each eigenvalue is:

$$\theta_i = \theta R_i\,\theta X_i = \text{BesselJ}\left(0,\lambda_i\, r\right)\left[C_{3,i}\sinh\left(\lambda_i\, x\right)+C_{4,i}\cosh\left(\lambda_i\, x\right)\right]$$

It is always a good idea to use Maple to check your work as you move through a problem like this one; enter the eigencondition and general solution into Maple:

```
> restart;
> BesselJ(0,lambda*r_out)=0;
```

$$\text{BesselJ}(0,\, \lambda\, r\_out) = 0$$

```
> theta:=(x,r)->BesselJ(0,lambda*r)*(C3*sinh(lambda*x)+C4*cosh(lambda*x));
```

$$\theta := (x,\, r) \rightarrow \text{BesselJ}(0,\, \lambda\, r)\,(C3\, \sinh(\lambda\, x) + C4\, \cosh(\lambda\, x))$$

and verify that the solution satisfies the governing differential equation, Eq. (1):

```
> r*diff(diff(theta(x,r),x),x)+diff(r*diff(theta(x,r),r),r);
```

$$r\,\text{BesselJ}(0, \lambda\, r)\,(C3\,\sinh(\lambda\, x)\,\lambda^2 + C4\,\cosh(\lambda\, x)\,\lambda^2)$$
$$- \text{BesselJ}(1, \lambda\, r)\,\lambda\,(C3\,\sinh(\lambda\, x) + C4\,\cosh(\lambda\, x))$$
$$- r\left(\text{BesselJ}(0, \lambda\, r) - \frac{\text{BesselJ}(1, \lambda\, r)}{\lambda\, r}\right)\lambda^2\,(C3\,\sinh(\lambda\, x) + C4\,\cosh(\lambda\, x))$$

```
> simplify(%);
```

$$0$$

and the boundary conditions in the r-direction:

```
> eval(diff(theta(x,r),r),r=0);
```

$$0$$

```
> theta(x,r_out);
```

$$\text{BesselJ}(0, \lambda\, r\_out)\,(C3\,\sinh(\lambda\, x) + C4\,\cosh(\lambda\, x))$$

```
> simplify(%);
```

$$0$$

The general solution is the sum of the solutions for each eigenvalue:

$$\theta = \sum_{i=1}^{\infty} \theta_i = \sum_{i=1}^{\infty} \text{BesselJ}\left(0, \lambda_i\, r\right)\left[C_{3,i} \sinh\left(\lambda_i\, x\right) + C_{4,i} \cosh\left(\lambda_i\, x\right)\right] \tag{12}$$

Equation (12) is substituted into the non-homogeneous boundary condition at $x = 0$, Eq. (4):

$$R_c''\, k \sum_{i=1}^{\infty} \text{BesselJ}\left(0, \lambda_i\, r\right) \lambda_i \left[C_{3,i} \underbrace{\cosh\left(\lambda_i\, 0\right)}_{1} + C_{4,i} \underbrace{\sinh\left(\lambda_i\, 0\right)}_{0}\right] =$$
$$\sum_{i=1}^{\infty} \text{BesselJ}\left(0, \lambda_i\, r\right)\left[C_{3,i} \underbrace{\sinh\left(\lambda_i\, 0\right)}_{0} + C_{4,i} \underbrace{\cosh\left(\lambda_i\, 0\right)}_{1}\right]$$

which can be simplified to:

$$\sum_{i=1}^{\infty} \text{BesselJ}\left(0, \lambda_i\, r\right)\left(R_c''\, k\, \lambda_i\, C_{3,i} - C_{4,i}\right) = 0 \tag{13}$$

Equation (13) can only be true if:

$$R_c''\, k\, \lambda_i\, C_{3,i} = C_{4,i} \tag{14}$$

Substituting Eq. (14) into Eq. (12) leads to:

$$\theta = \sum_{i=1}^{\infty} C_i \,\text{BesselJ}(0,\lambda_i\, r)\left[\sinh(\lambda_i\, x) + R''_c\, k\, \lambda_i \cosh(\lambda_i\, x)\right] \tag{15}$$

Equation (15) is substituted into the final non-homogeneous boundary condition at $x = th$, Eq. (5):

$$k \left.\frac{\partial \theta}{\partial x}\right|_{x=th} = k \sum_{i=1}^{\infty} C_i \,\text{BesselJ}(0,\lambda_i\, r)\,\lambda_i \left[\cosh(\lambda_i\, th) + R''_c\, k\, \lambda_i\, \sinh(\lambda_i\, th)\right] = \begin{cases} \dot{q}''_{laser} & \text{for } 0 < r < r_{laser} \\ 0 \text{ for } r_{laser} < r < r_{out} \end{cases} \tag{16}$$

When we reached this point for our previous problems, the eigenfunctions were either sines or cosines and we took advantage of their orthogonality to reduce the summation to an equation for each constant C_i. Bessel functions are also orthogonal provided that they are first multiplied by a weighting function, r, before integrating. That is, Eq. (16) is multiplied by r Bessel_J(0,λ_j r) and then integrated from $r = 0$ to $r = r_{out}$:

$$k \sum_{i=1}^{\infty} C_i\, \lambda_i \left[\cosh(\lambda_i\, th) + R''_c\, k\, \lambda_i\, \sinh(\lambda_i\, th)\right] \int_0^{r_{out}} r \,\text{BesselJ}(0,\lambda_j\, r)\,\text{BesselJ}(0,\lambda_i\, r)\, dr =$$

$$\dot{q}''_{laser} \int_0^{r_{laser}} r\, \text{BesselJ}(0,\lambda_j\, r)\, dr$$

Because the weighted eigenfunctions are orthogonal, all of the terms in the summation integrate to zero except for the one in which $i = j$; therefore, the summation is again reduced to an equation for each constant:

$$k\, C_i\, \lambda_i \left[\cosh(\lambda_i\, th) + R''_c\, k\, \lambda_i\, \sinh(\lambda_i\, th)\right] \int_0^{r_{out}} r\, \text{BesselJ}^2(0,\lambda_i\, r)\, dr = \dot{q}''_{laser} \int_0^{r_{laser}} r\, \text{BesselJ}(0,\lambda_i\, r)\, dr \tag{17}$$

The integrals in Eq. (17) may be evaluated using tables of Bessel function relations or, more conveniently, using Maple. The first integral is:

```
> restart;
> int(r*(BesselJ(0,lambda*r))^2,r=0..r_out);
```

$$\frac{1}{2}\, \frac{r\_out\,\left(\sqrt{\pi}\,\lambda\, r\_out\, \text{BesselJ}(0, \lambda\, r\_out)^2 + \sqrt{\pi}\,\lambda\, r\_out\, \text{BesselJ}(1, \lambda\, r\_out)^2\right)}{\sqrt{\pi}\,\lambda}$$

```
> simplify(%);
```

$$\frac{1}{2}\, r\_out^2\,\left(\text{BesselJ}(0, \lambda\, r\_out)^2 + \text{BesselJ}(1, \lambda\, r\_out)^2\right)$$

and the second integral is:

```
> int(r*BesselJ(0,lambda*r),r=0..r_laser);
```

$$\frac{r\_laser \, \text{BesselJ}(1, r\_laser \, \lambda)}{\lambda}$$

Substituting these results into Eq. (17) leads to:

$$\frac{k\, C_i\, \lambda_i\, r_{out}^2}{2}\left[\cosh\left(\lambda_i\, th\right) + R_c''\, k\, \lambda_i\, \sinh\left(\lambda_i\, th\right)\right]\left[\text{BesselJ}^2\left(0, \lambda_i\, r_{out}\right) + \text{BesselJ}^2\left(1, \lambda_i\, r_{out}\right)\right] = \frac{\dot{q}''_{laser}\, r_{laser}\, \text{BesselJ}\left(1, \lambda_i\, r_{laser}\right)}{\lambda_i}$$

Solving for the constants:

$$C_i = \frac{2\, \dot{q}''_{laser}\, r_{laser}\, \text{BesselJ}\left(1, \lambda_i\, r_{laser}\right)}{k\, \lambda_i^2\, r_{out}^2\left[\cosh\left(\lambda_i\, th\right) + R_c''\, k\, \lambda_i\, \sinh\left(\lambda_i\, th\right)\right]\left[\text{BesselJ}^2\left(0, \lambda_i\, r_{out}\right) + \text{BesselJ}^2\left(1, \lambda_i\, r_{out}\right)\right]} \qquad (18)$$

The solution is programmed in EES. The inputs are entered:

```
"EXAMPLE 2.3-2: Laser Machining"
$UnitSystem SI MASS  RAD PA K J
$Tabstops 0.2 0.4 0.6 0.8 3.5

"Inputs"
th=1 [cm]*convert(cm,m)                        "thickness of disk"
r_out=10 [cm]*convert(cm,m)                    "outer radius of disk"
r_laser= 0.5 [cm]*convert(cm,m)                "radius of laser"
q``_dot_laser=1e5 [W/m^2]                      "laser flux"
k=1.4 [W/m-K]                                  "conductivity "
T_c=ConvertTemp(C,K,20 [C])                    "temperature of coolant and bed"
R``_c=2.8e-4 [m^2-K/W]                         "contact resistance"
```

The eigenvalues are computed using Eq. (10), which is repeated below:

$$\text{BesselJ}\left(0, \lambda_i\, r_{out}\right) = 0 \text{ where } i = 1, 2, \ldots\infty \qquad (10)$$

The eigenvalues cannot be computed explicitly. Instead, the implicit equation must be solved over different ranges of the independent variable in order to obtain subsequent eigenvalues. Without a computer, we would be reduced to using only a few of these eigenvalues which would be manually entered from a table. However, it is possible to automate the process of finding these eigenvalues using EES. First, specify how many terms will be used in the series and identify the lower and upper limits that characterize the range of each of these eigenvalues. Figure 2 shows that the zeros of the function BesselJ(0,x) are located in successive multiples of

π, much like the cosine function. An array is created that defines the lower and upper limits as well as suitable guess values for each of the eigenvalues:

```
N=100                                                "number of terms in solution"
duplicate i=1,N
    lowerlimit[i]=(i-1)*pi/r_out                     "lower limit of eigenvalue"
    upperlimit[i]=i*pi/r_out                         "upper limit of eigenvalue"
    guess[i]=lowerlimit[i]+pi/(2*r_out)              "guess value for eigenvalue"
end
```

The implicit equation that defines the eigenvalue is entered into EES:

```
duplicate i=1,N
    BesselJ(0,lambda[i]*r_out)=0                     "solve for eigenvalues"
end
```

Solving the EES program at this point will lead to the same eigenvalue for each term in the series, which can be observed by opening the Array table. EES determines the solution to the set of equations that falls within the upper and lower limits specified in the Variable Information window. It is possible to specify an array of limits and guess values that are defined by the arrays lowerlimit, upperlimit, and guess. Navigate to the Variable Information Window and de-select the show arrays option. Enter the arrays (specified by the [ ] indicator) into the appropriate column of the lambda entry, as shown in Figure 3.

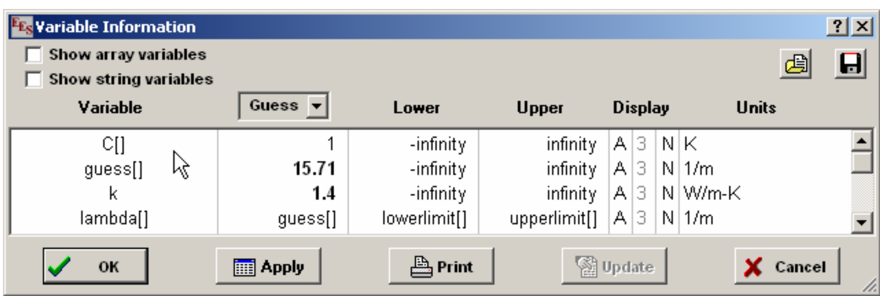

Figure 3: Variable Information window with limits and guess values set for lambda

Solving the EES problem with these limits and guess values will result in the identification of the correct eigenvalues for each term in the series.

A position to evaluate the solution is specified in terms of a dimensionless axial and radial coordinate:

```
x_bar=0.5 [-]                                        "dimensionless axial position"
r_bar=0.5 [-]                                        "dimensionless radial position"
x=x_bar*th                                           "x-position"
r=r_bar*r_out                                        "r-position"
```

Equation (18) is used to evaluate each of the constants and Eq. (15) is used to evaluate the associated term of the series:

```
duplicate i=1,N
```

```
    C[i]=2*q``_dot_laser*r_laser*BesselJ(1,lambda[i]*r_laser)/(k*lambda[i]^2*r_out^2*&
        (cosh(lambda[i]*th)+R``_c*k*lambda[i]*sinh(lambda[i]*th)*((BesselJ(0,lambda[i]*r_out))^2+&
        (BesselJ(1,lambda[i]*r_out))^2)))
    theta[i]=C[i]*BesselJ(0,lambda[i]*r)*(sinh(lambda[i]*x)+R``_c*k*lambda[i]*cosh(lambda[i]*x))
end
```

The solution for the temperature is obtained by summing the terms and adding T_c:

```
T=sum(theta[1..N])+T_c                                  "temperature"
T_Celsius=converttemp(K,C,T)                            "in C"
```

A parametric table is generated that contains the temperature evaluated over a grid that includes the entire computational domain. The contour plot generated in EES is shown in Figure 4. Note that the highest temperatures occur at r= 0 and x = th where the laser strikes the disk, as expected.

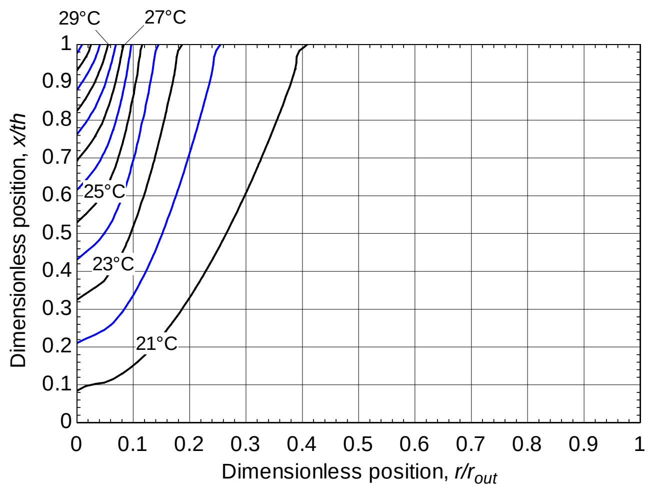

Figure 4: Contour plot of the temperature distribution in the work piece.

This problem is a good example of a situation where the use of analytical solutions might be useful beyond simply providing verification for numerical solutions. The analytical solution developed here will not break down, even at very small length scales. A numerical solution might have problems simultaneously resolving both the large spatial scales associated with the entire disk and the very small spatial scales associated with the irradiated spot. Consequently, one modeling approach that is sometimes used is a hybrid model; a numerical solution of the work piece would deal with details regarding the overall, possibly complex shape of the material as well as the details of the cooling provided to its edges. The numerical model would not consider in detail the region struck by the laser; rather the heat must be applied uniformly over a region that is large enough to be sufficiently resolved. The numerical model can be used to provide boundary conditions (i.e., the “far-field” conditions”) for the analytical solution which is used to analyze the very local effects of the laser heating. There are many situations where it is necessary to consider effects that occur over a wide range of length or time scales. It is often the case that multiple models, each tailored to resolve effects of a certain type, must be “stitched together” through suitably defined boundary conditions. We will not deal with these problems substantially in this textbook, but you should be alert for opportunities where two models of various types can be used together to provide a more insight than either one alone.