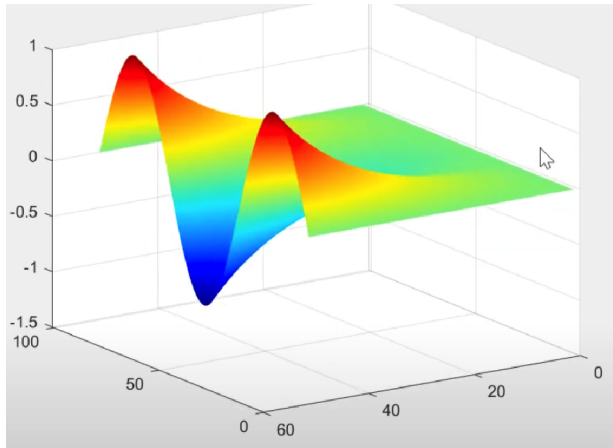# Diffusion in Two Dimensions — Steady State

Completed and Analyzed in class, April 18, 2025

This is our twenty-first notebook. In the twentieth notebook, we did one-dimensional diffusion. Now we are graduating to two-dimensional diffusion. Here is an example:

NB: This is a two-dimensional problem, not a three-dimensional one. The third dimension in the plot is just helping to visualize the temperature, and unless you are red-green colorblind, the information supplied by displaying a third dimension is redundant.

The problem has the type of equation that we are currently studying. The specifics of this problem is that it is a two-dimensional plate that has alternating regions of hot and cold along one side, and is at an intermediate temperature along the other three sides. For more, see: https://youtu.be/2aJ3fFwET68.

We are going to specify the equation on the interior, and then the boundary conditions, and Mathematica is going to figure out the steady-state solution.

If you want to see more of the wide variety of equations that can be solved using the techniques we are using, try https://www.cfm.brown.edu/people/dobrush/am34/Mathematica/ch6/parabolic.html.

Two-Dimensional Diffusion — Steady-State Theory

In one dimension, we had:

$$c\,\frac{dT}{dt} = \sigma\,\frac{d^2\,T}{dx^2}$$

The obvious generalization (I won't bore you with yet another derivation of combinations of second

$$c\,\frac{dT}{dt} = \sigma\!\left(\frac{d^2\,T}{dx^2} + \frac{d^2\,T}{dy^2}\right)$$

$$C\frac{dT}{dt} = \sigma \frac{d^2 T}{dx^2}$$

derivatives) to two dimensions is:

$$c\frac{dT}{dt} = \sigma\left(\frac{d^2 T}{dx^2} + \frac{d^2 T}{dy^2}\right)$$

It can be very interesting to see the time-dependent way that the system settles down to equilibrium, but *now I want to make a huge simplification, which is that the system has already settled down to equilibrium.*

As long as the external sources of heat are supplied in a steady (unchanging) way, the system will eventually settle down. Once it settles down, $\frac{dT}{dt} = 0$, and this is true everywhere in x and y. So for the steady state, our equations become:

$$0 = \sigma\left(\frac{d^2 T}{dx^2} + \frac{d^2 T}{dy^2}\right)$$

and T, whatever it is, is unchanging, and so it is no longer a function of t. It is only a function of x and y. Also, even if We just have this as what we specify to Mathematica:

```
In[2]:= Module[{sigma = 1},
         {sigma (Derivative[2, 0][temp][x, y] + Derivative[0, 2][temp][x, y]) == 0}] //
        TraditionalForm
```
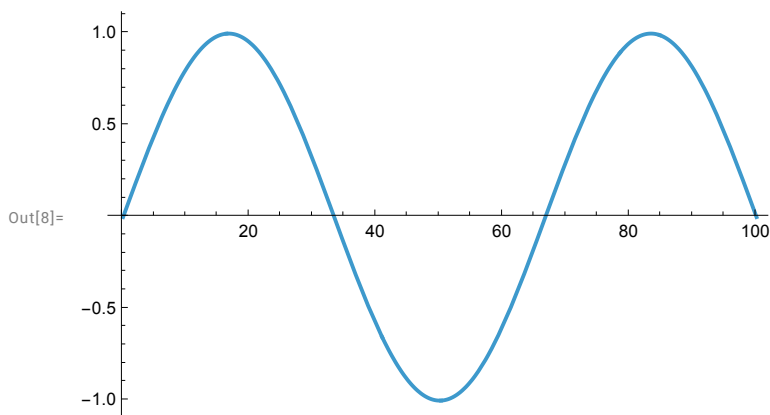
Out[2]//TraditionalForm=

$$\left\{\text{temp}^{(0,2)}(x, y) + \text{temp}^{(2,0)}(x, y) = 0\right\}$$

Adding the Boundary Conditions

Let's make boundary conditions like the one pictured at the beginning of this notebook:

```
In[3]:= lengthX = 100;
        lengthY = 60;
        amplitude = 1;
```

```
mode = 3;
f[x_] := amplitude Sin[mode Pi x / lengthX]
Plot[f[x], {x, 0, lengthX}]
```

Out[8]=

So here is our problem but now with the boundary conditions added:

```
In[9]:= Module[{sigma = 1},
    {sigma (Derivative[2, 0][temp][x, y] + Derivative[0, 2][temp][x, y]) == 0,
     temp[x, 0] == f[x], temp[x, lengthY] == 0, temp[0, y] == 0,
     temp[lengthX, y] == 0}]  // TraditionalForm
```

Out[9]//TraditionalForm=

$$\left\{\text{temp}^{(0,2)}(x, y) + \text{temp}^{(2,0)}(x, y) = 0, \text{temp}(x, 0) = \sin\left(\frac{3\pi x}{100}\right), \text{temp}(x, 60) = 0, \text{temp}(0, y) = 0, \text{temp}(100, y) = 0\right\}$$

No Initial Conditions!

This is a steady-state problem. Time is not involved. We do not need to specify any initial conditions. That means the whole problem is what we already specified in the previous section:

```
In[10]:= twoDimensionalDiffusionProblem = Module[{sigma = 1},
    { sigma (Derivative[2, 0][temp][x, y] + Derivative[0, 2][temp][x, y]) == 0,
     temp[x, 0] == f[x], temp[x, lengthY] == 0, temp[0, y] == 0, temp[lengthX, y] == 0}]
```

Out[10]=

$$\left\{\text{temp}^{(0,2)}[x, y] + \text{temp}^{(2,0)}[x, y] == 0, \text{temp}[x, 0] == \text{Sin}\left[\frac{3\pi x}{100}\right],\right.$$

$$\left.\text{temp}[x, 60] == 0, \text{temp}[0, y] == 0, \text{temp}[100, y] == 0\right\}$$

A Great Trick — Known as "Separation of Variables"

We ***guess*** that $T(x, y) = u(x) v(y)$.

Upon putting this guess into the differential equations, we ***discover*** that if this trick is to work, then $u(x)$ must satisfy the ordinary differential equation:

$$\frac{d^2 u}{d x^2} = -\lambda u$$

and the solutions that have $u(0) = 0$ and $u(l_x) = 0$ are

$$u_n(x) = \sin\frac{n\pi x}{l_x}, \text{ with } n = 1, 2, 3, \ldots$$

and $\lambda_n = \frac{n^2 \pi^2}{l_x^2}$

Now that we know the possible $u_n(x)$ and λ_n, we put those into the v equation which is

$$\frac{d^2 v}{d y^2} = \lambda v$$

and which has the solutions

$$v(y) = c_+ e^{\sqrt{\lambda_n} y} + c_- e^{-\sqrt{\lambda_n} y}$$

$$l_y$$

$$\frac{d^2 v}{dy^2} = \lambda\, v$$

$$v(y) = c_+\, e^{\sqrt{\lambda_n}\, y} + c_-\, e^{-\sqrt{\lambda_n}\, y}$$

If this solution is to be zero at l_y, we must have

$$0 = c_+\, e^{\sqrt{\lambda_n}\, l_y} + c_-\, e^{-\sqrt{\lambda_n}\, l_y}$$

If the full solution is to be $\sin\frac{3\pi x}{l_x}$ at $y = 0$ that tells us we must have the solutions with $u_3(x)$ and λ_3 and that

$$1 = c_+ + c_-$$

So

$$0 = c_+\, e^{\sqrt{\lambda_n}\, l_y} + (1 - c_+)\, e^{-\sqrt{\lambda_n}\, l_y} = c_+\left(e^{\sqrt{\lambda_n}\, l_y} - e^{-\sqrt{\lambda_n}\, l_y}\right) + e^{-\sqrt{\lambda_n}\, l_y}$$

So

$$c_+ = \frac{-e^{-\sqrt{\lambda_n}\, l_y}}{e^{\sqrt{\lambda_n}\, l_y} - e^{-\sqrt{\lambda_n}\, l_y}}$$

and

$$c_- = 1 - c_+ = 1 - \frac{-e^{-\sqrt{\lambda_n}\, l_y}}{e^{\sqrt{\lambda_n}\, l_y} - e^{-\sqrt{\lambda_n}\, l_y}} = \frac{e^{\sqrt{\lambda_n}\, l_y}}{e^{\sqrt{\lambda_n}\, l_y} - e^{-\sqrt{\lambda_n}\, l_y}}$$

So

$$v(y) = c_+\, e^{\sqrt{\lambda_n}\, y} + c_-\, e^{-\sqrt{\lambda_n}\, y} = \frac{-e^{-\sqrt{\lambda_n}\, l_y}}{e^{\sqrt{\lambda_n}\, l_y} - e^{-\sqrt{\lambda_n}\, l_y}}\, e^{\sqrt{\lambda_n}\, y} + \frac{e^{\sqrt{\lambda_n}\, l_y}}{e^{\sqrt{\lambda_n}\, l_y} - e^{-\sqrt{\lambda_n}\, l_y}}\, e^{-\sqrt{\lambda_n}\, y}$$

Stick in from what we learned solving the $u(x)$ equation, that we must $\lambda_3{}^2 = \frac{3\pi}{l_x}$, and we then have

$$v(y) = \frac{-e^{-\frac{3\pi}{l_x}\, l_y}}{e^{\frac{3\pi}{l_x}\, l_y} - e^{-\frac{3\pi}{l_x}\, l_y}}\, e^{\frac{3\pi}{l_x}\, l_y\, y} + \frac{e^{\frac{3\pi}{l_x}\, l_y}}{e^{\frac{3\pi}{l_x}\, l_y} - e^{-\frac{3\pi}{l_x}\, l_y}}\, e^{-\frac{3\pi}{l_x}\, l_y\, y} = \frac{\sinh\left(\frac{3\pi}{l_x}\, (l_y - y)\right)}{\sinh\left(\frac{3\pi}{l_x}\, l_y\right)}$$

In the very last step, I used that these particular combinations of exponentials are called sinh (pro-nounced "cinch" as in "cinching a knot"), and they have companion combinations called cosh (pro-nounced "kosh") that we did not need.
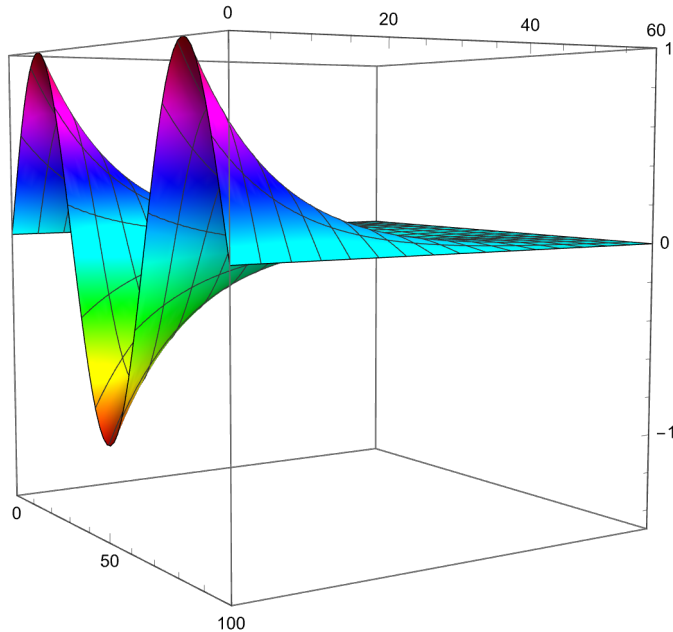
So

$$T(x, y) = u(x)\, v(y) = \sin\frac{3\pi x}{l_x}\ \frac{\sinh\!\left(\frac{3\pi}{l_x}(l_y - y)\right)}{\sinh\!\left(\frac{3\pi}{l_x} l_y\right)}$$
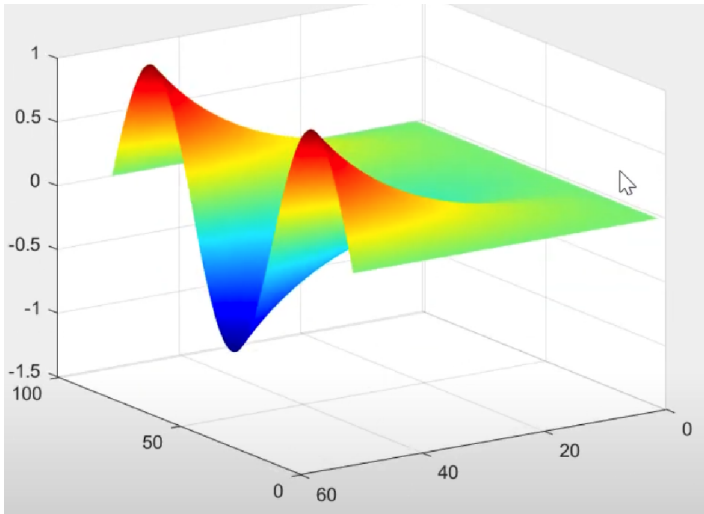
Graphing the Solution

In[11]:= `exactSolution[x_, y_] := Sin`$\left[\dfrac{3\ \text{Pi}}{\text{lengthX}}\ x\right]$ $\dfrac{\text{Sinh}\left[\dfrac{3\ \text{Pi}}{\text{lengthX}}\ (\text{lengthY} - y)\right]}{\text{Sinh}\left[\dfrac{3\ \text{Pi}}{\text{lengthX}}\ \text{lengthY}\right]}$

In[12]:= `Plot3D[exactSolution[x, y], {x, 0, lengthX}, {y, 0, lengthY},`
`    PlotRange → {{0, lengthX}, {0, lengthY}, {-1.5, 1}}, PlotPoints → 60,`
`    MaxRecursion → 6, ColorFunction → Function[{x, y, z}, Hue[z]], BoxRatios → {1, 1, 1}]`
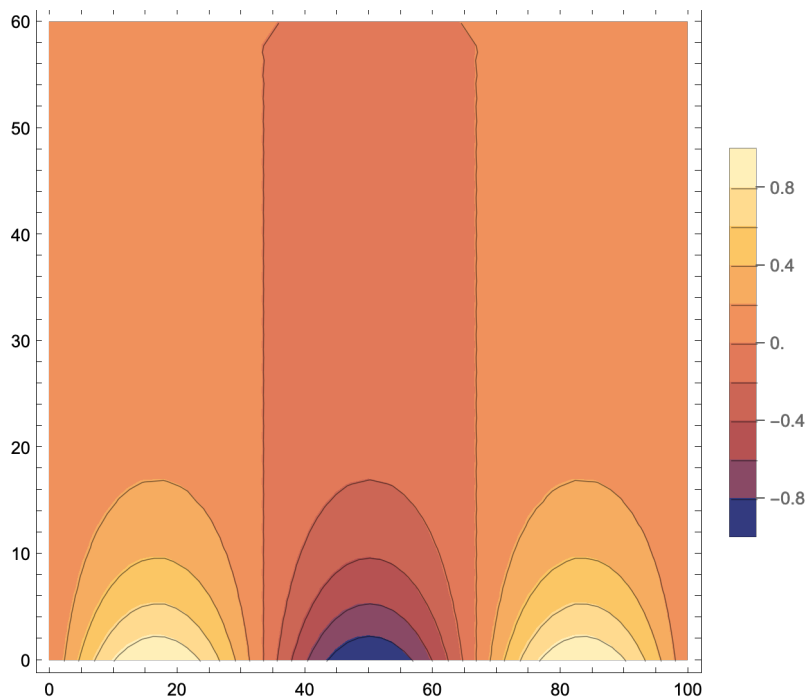
Out[12]=

We have reproduced what we set out to reproduce (but see the Epilogue below):

It is good to have shading of a 2-D plot (rather than 3-D) plots in your arsenal of output techniques:

In[31]:=
```
ContourPlot[exactSolution[x, y], {x, 0, lengthX}, {y, 0, lengthY},
  PlotRange → {Automatic, Automatic, {-1, 1}}, PlotLegends → Automatic]
```

Out[31]=

Epilogue

I had no intention when I originally planned this particular notebook in our steady progression of ever-more-complicated notebooks to show you so much math. I wanted Mathematica to do the work! I only

did the separation of variables trick above because I couldn't get Mathematica to give me the right answer. If someone sees what is wrong with the far more pedantic approach below, please let me know.

That said, there will be some value, when we get to the notebooks for the final two Fridays of the course on quantum mechanics, that you have already had some exposure to the separation of variables trick.

Before we do that, I am going to do one more diffusion notebook, using a completely different technique. I will do it in three dimensions, and instead of heat, it will be a simple model of diffusion of pollution downwind of a smokestack.

Making Mathematica Numerically Solve the Problem

In[13]:= `twoDimensionalDiffusionSolutionRule =`
`  NDSolve[twoDimensionalDiffusionProblem, temp, {x, 0, lengthX}, {y, 0, lengthY}][[1]]`

Out[13]=

$\left\{ \texttt{temp} \rightarrow \texttt{InterpolatingFunction} \Big[\boxed{\blacksquare ~\sim~ \begin{matrix} \text{Domain: } \{\{0., 100.\}, \{0., 60.\}\} \\ \text{Output: scalar} \end{matrix}} \Big] \right\}$

Convert the rule to a function:

In[14]:= `twoDimensionalDiffusionSolution[t_, x_] =`
`  temp[x, y] /. twoDimensionalDiffusionSolutionRule`

Out[14]=
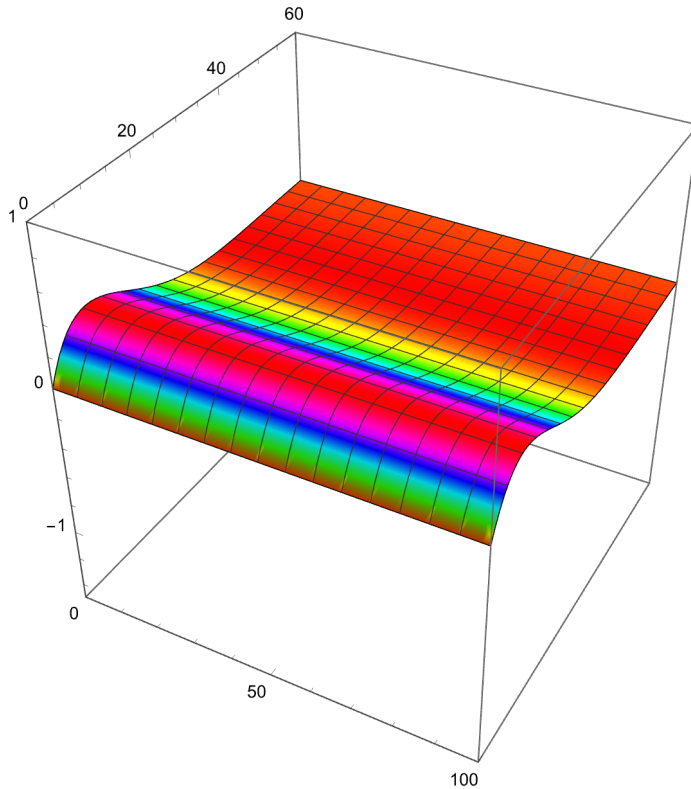
$\texttt{InterpolatingFunction} \Big[\boxed{\blacksquare ~\sim~ \begin{matrix} \text{Domain: } \{\{0., 100.\}, \{0., 60.\}\} \\ \text{Output: scalar} \end{matrix}} \Big] [x, y]$

Plotting the Numerical Solution

In[15]:= `Plot3D[twoDimensionalDiffusionSolution[x, y], {x, 0, lengthX}, {y, 0, lengthY},`
`    PlotRange → {{0, lengthX}, {0, lengthY}, {-1.5, 1}}, PlotPoints → 60,`
`    MaxRecursion → 6, ColorFunction → Function[{x, y, z}, Hue[z]], BoxRatios → {1, 1, 1}]`

Out[15]=

That is clearly wrong. It does not obey the boundary conditions. I checked this code many times before giving up and doing the separation of variables trick.