Dubin 3.2.18 Grilling a Steak 11-15-16

N. T. Gladd

Initialization: Be sure the files *NTGStylesheet2.nb* and *NTGUtilityFunctions.m* is are in the same directory as that from which this notebook was loaded. Then execute the cell immediately below by mousing left on the cell bar to the right of that cell and then typing “shift” + “enter”. Respond “Yes” in response to the query to evaluate initialization cells.

In[8]:=
```
SetDirectory[NotebookDirectory[]];
(* set directory where source files are located *)
SetOptions[EvaluationNotebook[], (* load the StyleSheet *)
 StyleDefinitions → Get["NTGStylesheet2.nb"]];
Get["NTGUtilityFunctions.m"]; (* Load utilities package *)
```

The original version of this notebook is *Dubin 4.2.2 Grilling Refined 2 (6-18-2005)*

Purpose

I solve a heat equation problem from Chapter 3 of *Numerical and Analytical Methods for Scientists and Engineers, Using Mathematica*, Daniel Dubin. The specific Problem 3.1(18) considers grilling a steak.

Solution of PDE and determination of time for temperature to rise to specified level

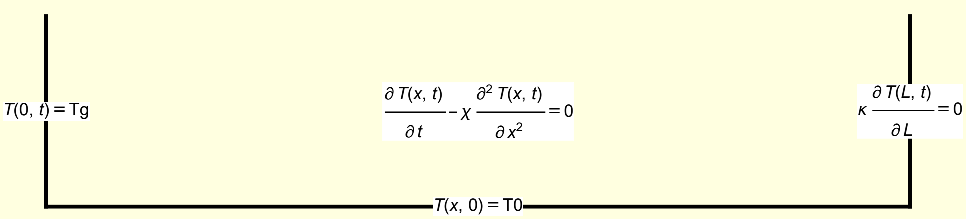
I construct an Association that encapsulates information about this problem, and then apply the function *DSolveHeatEquation* that attempts to solve this problem directly using the Mathematica function *DSolve*.

In[10]:=

```
A1 =
  Module[{description, pde, bcL, bcR, ic, eqns, depVar,
    assumptions, substitutions, simplifications, names, values},
   description = "Dubin problem 3.1(17) Grilling a steak\nHomogeneous
      heat equation, inhomogeneous Dirichlet boundary
      condition\nhomogeneous von Neumann bouncdary condition";
   pde = D[T[x, t], t] - χ D[T[x, t], {x, 2}] == 0;
   bcL = T[0, t] == Tg; (* inhomogeneous Dirichlet *)
   bcR = κ Derivative[1, 0][T][L, t] == 0;
   (* insulatied, homogeneous von Neumann *)
   ic = T[x, 0] == T0;
   eqns = {pde, bcL, bcR, ic};
   depVar = T[x, t];
   assumptions = {L > 0, χ > 0};
   substitutions = {K[1] → n};
   simplifications = {n ∈ Integers};
   values = {description, pde, bcL, bcR, ic,
     eqns, depVar, assumptions, substitutions, simplifications};
   names = {"description", "pde", "bcL", "bcR", "ic", "eqns",
     "depVar", "assumptions", "substitutions", "simplifications"};
   AssociationThread[names, values]];

Module[{soln, G},
 soln = DSolveHeatEquation[A1];
 AppendTo[A1, "soln" → soln];
 Print@ShowPDESetup[A1];
 A1["soln"]]
```

Dubin problem 3.1(17) Grilling a steak
Homogeneous heat equation, inhomogeneous Dirichlet boundary condition
homogeneous von Neumann bouncdary condition



Out[11]=

$$T[x,t] \to Tg + \frac{1}{L} 2 \sum_{n=1}^{\infty} \frac{2\, e^{-\frac{(1-2n)^2 \pi^2 t \chi}{4 L^2}} L\, (T0 - Tg)\, \text{Sin}\left[\frac{(-1+2n)\pi x}{2L}\right]}{(-1+2n)\pi}$$

DSolve immediately solves this problem

In[12]:=

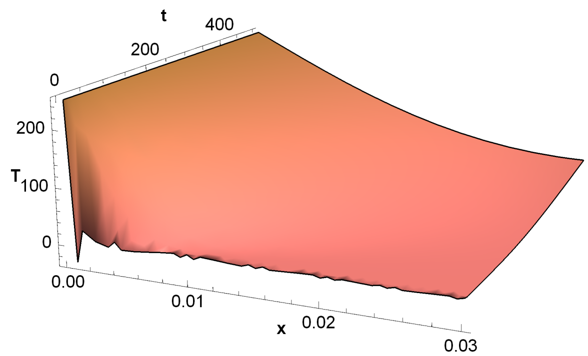
```
Clear[TSoln];
TSoln[x_, t_, χ_, L_, T0_, Tg_, nMax_] :=
```

$$Tg + \frac{1}{L} 2 \sum_{n=1}^{nMax} \frac{2\, e^{-\frac{(1-2n)^2 \pi^2 t \chi}{4 L^2}} L\, (T0 - Tg)\, \text{Sin}\left[\frac{(-1+2n)\pi x}{2L}\right]}{(-1+2n)\pi} \text{ // Activate}$$

In[14]:=

```
Module[{χ = 3 × 10^-7 (* m^2/s *), L = 0.03 (* m *), T0 = 8
   (* deg C *), Tg = 250 (* deg C *), nMax = 50, tMax = 500},
  Plot3D[TSoln[x, t, χ, L, T0, Tg, nMax], {x, 0, L},
   {t, 0, 500}, ColorFunction → (Blend[{Pink, Brown}, #3] &),
   AxesLabel → {Stl["x"], Stl["t"], Stl["T"]},
   Mesh → False, Boxed → False, PlotLegends → Automatic]]
```

Out[14]=

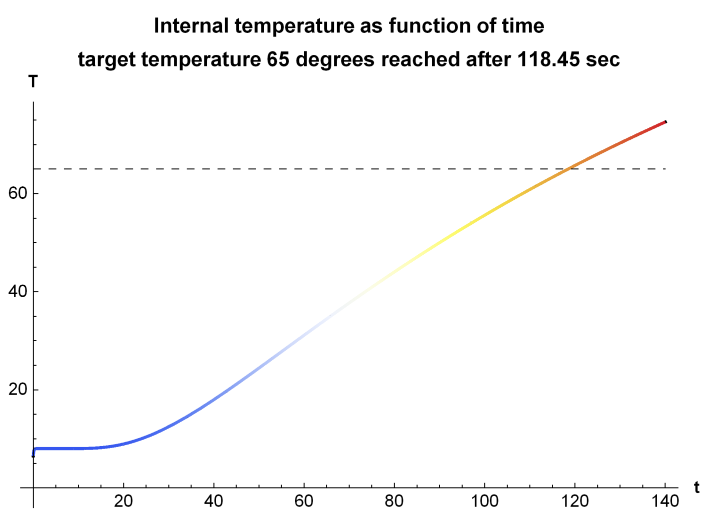


How long does it take for the internal temperature to rise to 65 degrees C.

In[15]:=

```
Module[{χ = 3 × 10^-7 (* m^2/s *), L = 0.03 (* m *),
   T0 = 8 (* deg C *), Tg = 250 (* deg C *), nMax = 50,
   tMax = 500, Ttarget = 65 (* deg C *), tDone, lab, refLine},
  refLine = {Directive[Black, Dashed], Line[{{0, Ttarget}, {140, Ttarget}}] };
  tDone = FindRoot[TSoln[0.01, t, χ, L, T0, Tg, nMax] == Ttarget, {t, 100}][[1, 2]];
  lab =
   Stl@StringForm["Internal temperature as function of time\ntarget temperature ``
       degrees reached after `` sec", Round@Ttarget, NF2@tDone];
  Plot[{TSoln[0.01, t, χ, L, T0, Tg, nMax]}, {t, 0, 140},
   ColorFunction → "TemperatureMap", AxesLabel → {Stl["t"], Stl["T"]},
   Epilog → refLine, PlotLabel → lab]]
```

Out[15]=



Functions

In[3]:=

```
Clear[ShowPDESetup];
ShowPDESetup[A_] := Module[{top = 1.0, right = 1.0,
   boundaries, labels, textInterior, textIC, textBCL, textBCR},
  boundaries = Line /@ {{{0, 0}, {right, 0}},
     {{0, 0}, {0, top}}, {{right, 0}, {right, top}}};
  labels = Text[PhysicsForm[A[#[[1]]] ], #[[2]]] & /@
    {{"pde", {right / 2, top / 2}}, {"ic", {right / 2, 0.0}},
     {"bcL", {0.0, top / 2}}, {"bcR", {right, top / 2}}};
  Graphics[{Directive[Black, Thick], boundaries, labels}, Axes → False,
   AspectRatio → 0.25, ImageSize → 500, PlotLabel → Stl[A["description"]]]]
```

In[4]:=

```
Clear[DSolveHeatEquation];
DSolveHeatEquation[A_] :=
 Module[{soln},
  soln =
   DSolve[A["eqns"], A["depVar"], {x, t}, Assumptions → A["assumptions"]][[1, 1]];
  soln = soln //. A["substitutions"];
  soln = Simplify[soln, A["simplifications"]];
  soln]
```