David Luby
ME 603
Project Deliverable 2
5-13-2022

Exploration of Cooling Parameters in CPU Heatsink Design

Introduction

A CPU is a computer component that requires cooling during operation to prevent from overheating and failure. The chips are typically cooled by a metal (HS) and either air or water, employing the principles of conduction and convection to remove heat from the CPU. The conductive heat transfer away from the CPU comes from the HS resting on top of the chip, and the convective heat transfer from the CPU is enabled mostly by fans or a pump.

The CPU creates a gradient with the air around the HS, driving its cooling. As the chip heats up during use, it conducts heat into the base of the sink, which then expels heat to local air. The air is blown away in a convective motion by a fan, making way for cooler particles to warm up and repeat the cycle.

While it is true that fans facilitate most of the convective cooling, HS design parameters can significantly enhance the fans' effectiveness in doing so. As the surface area of the sink increases, the amount of heat transferred from it—and therefore from the CPU—increases. These types of sink designs are often arrays of flat fins or pins that extend upward from the base of the sink where it contacts the CPU. The effect of the large surface area in tandem with the convective motion caused by the fans cools the chips for safe operation.

For the following simulation, a rectangular, three-dimensional CPU was slotted underneath a HS of equal dimensions as a control. An experimental HS with circular pins extending out of its top surface was compared against the bare sink model to test whether an increase in surface area makes a significant difference in cooling.

Methodology and Results

MATLAB's Partial Differential Equation Toolbox (MPDET) was used to simulate the effect of pins on cooling in a CPU. The add-on provides a library of functions used to solve the partial differential equations (PDE) underlying the heat transfer of the two models (via finite element analysis).

Three geometry meshes—a CPU with a bare HS on top, a CPU with an extended surface sink having nine pins, and a CPU with an extended surface sink having 25 pins —were constructed and are shown in Figures 1, 2, and 3. Each mesh was associated with a transient heat transfer simulation, material properties, and initial conditions and boundary conditions.

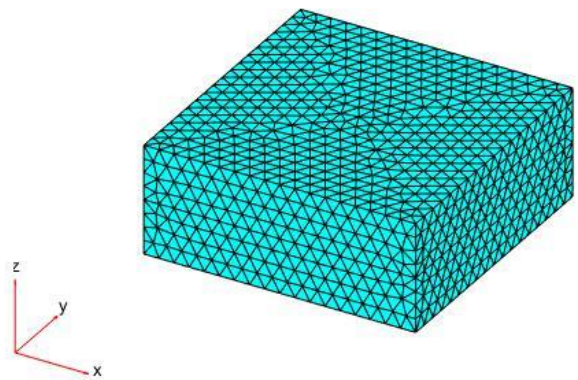


Figure 1: Mesh geometry of bare CPU and heat sink configuration.

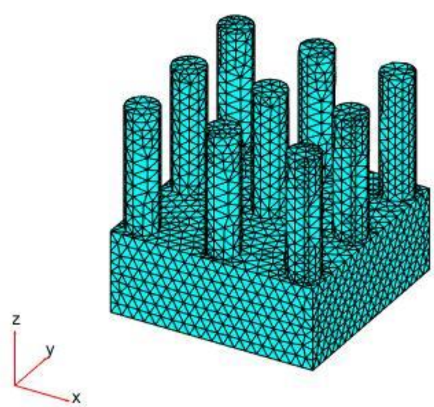


Figure 2: Mesh geometry of CPU and 9-pin heat sink configuration.

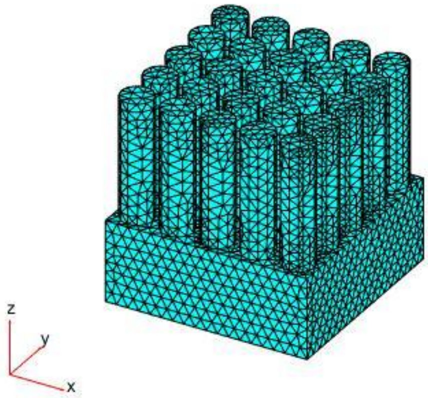


Figure 3: Mesh geometry of CPU and 25-pin heat sink configuration.

The important distinction beyond their unique geometries came from the formulation of the models' boundary conditions. The bare sink only interacted with the convective effects of the fan on its top surface, while the extruded sinks interacted on their top faces *and* all pin faces. The combination of the geometric and boundary distinction between the two types of models represents the variable with which the pins' effects were evaluated. Additional considerations used to evaluate the effect of the pin extrusions are owed to the 16-pin increase seen in Figure 3 compared to Figure 2.

Configuration	*V* (m/s)	*Pr*	*Length or Diameter* (m)	*v* (m^2/s)	*Re*	*Nu*	*h (W/m^2-K)*
Flat	5	0.707	0.05	1.57E-05	-	31.03	27.81
9 Pin	8.33		0.008		4239.4	17.2	56.21
25 Pin	25		0.008		12718	28.55	93.32

Table 1: Calculation of convection coefficient (h) and accessory values. Left to right: maximum velocity (*V*), Prandtl number (*Pr*), length or diameter or surface, kinematic viscosity (*v*), Reynolds number (*Re*), Nusselt number (*Nu*), average convection coefficient.

Equation 1: $\mathbf{10.45 - V_{inf} + 10V_{inf}^{0.5}}$

- *For flat plate convection coefficient in bare CPU.*

Equation 2: $V_{max} = \frac{S_T V_{INF}}{(S_T - D)}$

Equation 3: $Re_{D,max} = \frac{DV_{max}}{v}$

Equation 4: $Nu_{D,max} = C_1 Re_{D,max}^{m} Pr^{.36} \left(\frac{Pr}{Pr_s}\right)^{0.25}$

Equation 5: $\bar{h} = \frac{Nu_{D,max} k}{D}$

- *For convection in banks of tubes in extended surface CPU.*

Average convective heat transfer coefficients (h) and their associated values were calculated for a flat plate and banks of tubes using Equations 1-5 and are listed in Table 1. The temperature distributions of the bare and extended surface sinks were calculated with their respective convection coefficients using MPDET to solve the two PDE. The maximum unit temperature against time with one minute temporal resolutions over one hour were plotted and are shown in Figure 4 and Figure 5.

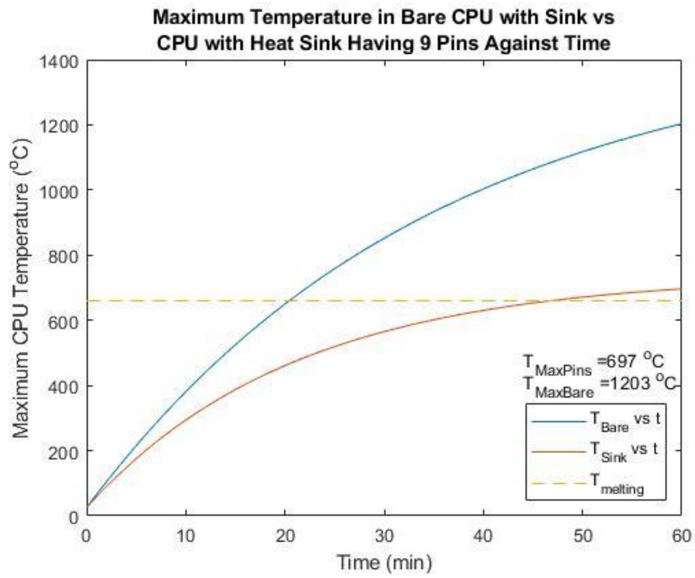


Figure 4: Comparison of CPU with bare heat sink configuration against CPU with 9-pin heat sink.

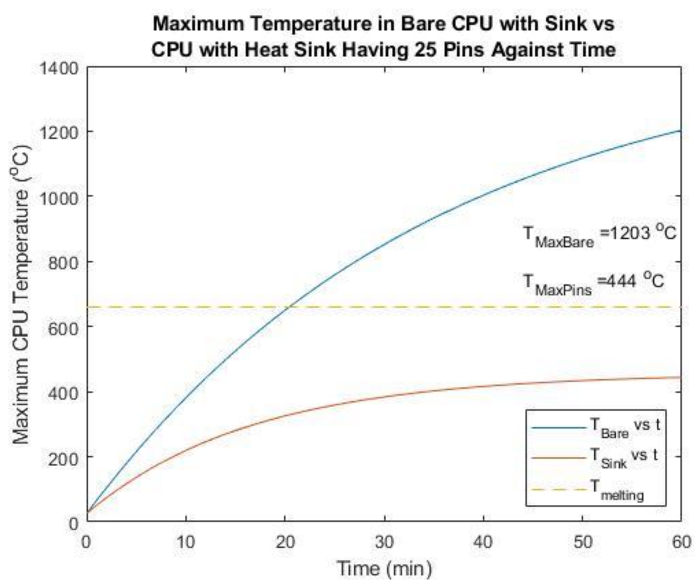


Figure 5: Comparison of CPU with bare heat sink configuration against CPU with 25-pin heat sink.

Configuration	*time (hr)*	*Melting Temp (C)*	*Steady State Temp (C)*	*Difference*
Flat	2	661	1417	-756
9 Pin			736	-75
25 Pin			455	206

Table 2: Steady state temperatures after two hours for all three configurations. Negative *difference* representative of steady state temperature beyond melting point of aluminum.

Steady-state temperatures (SST) constrained at maximum by the melting temperature of aluminum were derived as the maximum temperature in each model after two hours and are tabulated in Table 2.

Figure 6 and Figure 7 represent temperature overlayed on each model at a time of 7 seconds. Both figures feature the bare CPU unit in contrast to the 9 and 25-pin units.

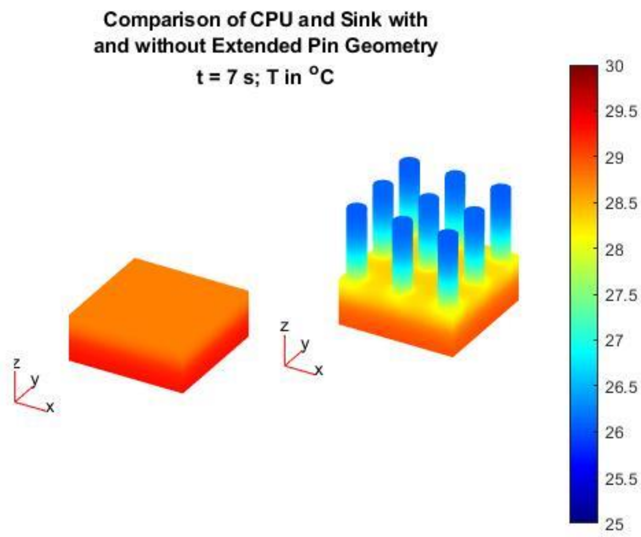


Figure 6: Temperature color map overlayed onto CPU with bare heat sink (left) and CPU with 9-pin heat sink configuration (right) in Celsius for time of seven seconds.

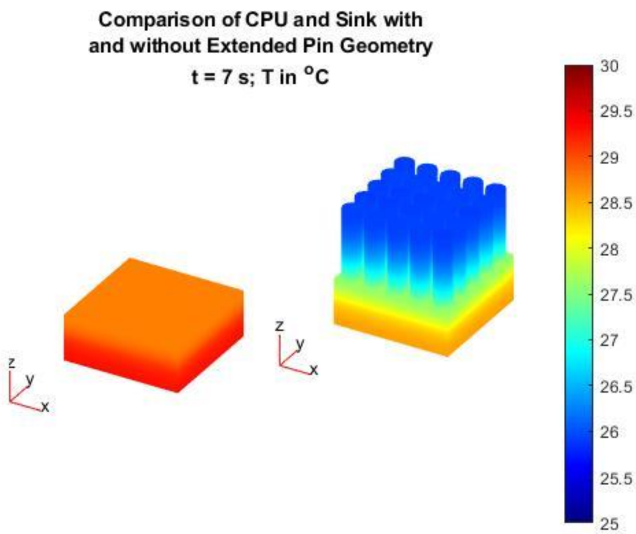


Figure 7: Temperature color map overlayed onto CPU with bare heat sink (left) and CPU with 25-pin heat sink configuration (right) in Celsius for time of seven seconds.

Discussion

The simulation's results substantiate the proposal that an increase in surface area for a heat sink will improve its capacity to cool its CPU based on the data in Table 2. The maximum temperature for the base CPU and sink configuration was over double the melting temperature for aluminum—this design is invalid. For the 9-pin configuration, the final temperature after two hours still exceeded the melting temperature of aluminum by about 1.1 times. This represents another failed design—but better. In the 25-pin case, the steady state temperature after two hours is viable at about 69% of the melting temperature for aluminum, demonstrating that the increased number of pins, and therefor surface area, resulted in improved cooling for the system.

Figure 4 and Figure 5 similarly show the impact of introducing pins to the heat sink. In both figures, for all time, the temperature of the extended surface HS are below that of the CPU with a bare sink. In Figure 5, it is obvious that the lower line (representative of the 25-pin HS) is approaching its steady state temperature after one hour, while the bare HS configuration is not yet close. The cause of this observation is to be attributed to the increased surface area from the additional pins.

The color map overlay shown for the 9-pin configuration in Figure 6 demonstrates that the CPU with an extruded surface configuration enhances the cooling of the system as evidenced by the lighter yellow and orange tones seen in the CPU and base of the HS on the right-side model. The same conclusion is prompted from looking at Figure 7; the base of the CPU with extruded surface HS is a much lighter orange than that of the bare HS CPU on the left. Further, the 25-pin CPU with HS in Figure 7 is a much lighter tone when compared the 9-pin CPU with HS in Figure 6 because of the additional 16 pins.

As the number of pins on a HS increases, this simulation provides that so too does the heat removal rate from the CPU. This is evidenced by overall lower temperatures simulated in the 9-pin configuration and the 25-pin configuration. Adding just 16 pins resulted in a 38% reduction in maximum temperature after two hours of uptime and a viable system featuring a steady state temperature below the melting temperature of aluminum. This cause and effect of additional pins resulting in improved cooling for the CPU is impressive, quite sensitive, and reflective of the underlying cooling mechanism: increasing surface area results in improved cooling.

Table of Contents

David Luby

```
ME 603
Project Deliverable No. 2
5-13-2022
```

```
clear;clc;
```

MUST CHANGE n TO 25 TO RETRIEVE ALL OF MY FIGURES. DONT WANT TO OUTPUT THIS

TWICE

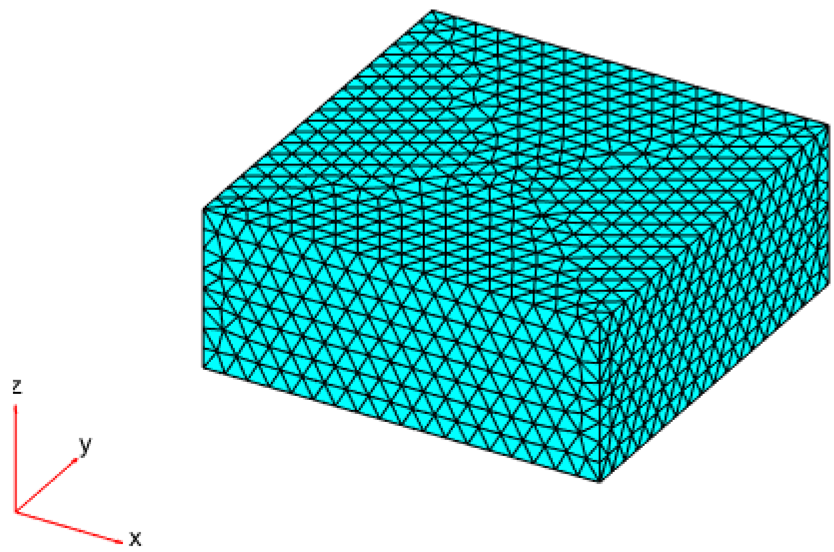
No. 1 -- Creating Geometries

```
% preallocating thermal models
cpuMod = createpde('thermal','transient');
sinkMod = createpde('thermal','transient');

% initializing variables
sLen = .05; % slab length (m)
sWid = .05; % slab width (m)
sHie = .01; % slab height (m)

pRad = .004; % pin radius (m)
pHie = .03; % pin height (m)
n = 9; % number of pins (must be square number)
```

No. 1a -- Bare CPU Model

```
% creating slab geometry description matrix (gd)
```

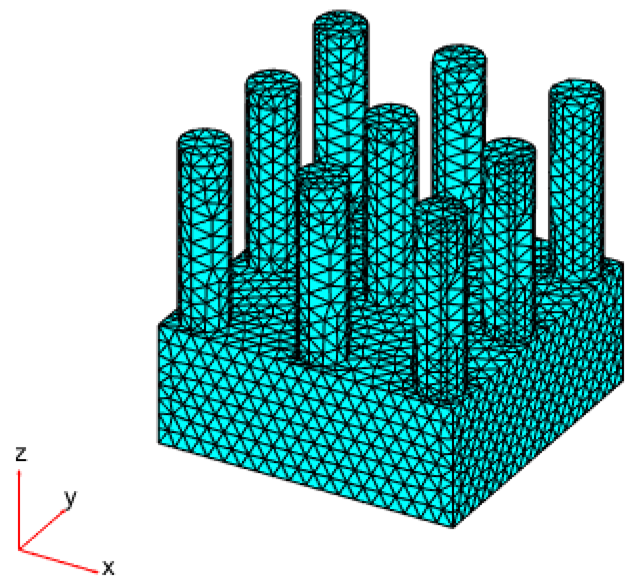
```
    % where: gd = [rect 4*edge xBL xTL xTR xBR yBL yTL yTR yBR]
        % and: 'BL' = bottom left, 'TL' = top left, etc.
slab = [3 4 0 0 sLen sLen 0 sWid sWid 0]';

gdCpu = slab; % complete geometry description for CPU

% creating set formula vector (sf)
    % where: sf = 'slab+pin-(cat+dog)'
        % specifies geometry description interactions
sfCpu = 'slab';

% creating name-space matrix (ns)
    % ns = char('slab','circle','dog')'
        % where: each name can be called in set formula vector

nsCpu = char('slab')';

% creating 2-d geometry from gd,sf,ns
gCpu = decsg(gdCpu,sfCpu,nsCpu);

% merging thermal preallocation with 2-d geometry
gCpu = geometryFromEdges(cpuMod,gCpu);

% extruding
gCpu = extrude(gCpu,sHie); % sink
siFaces = 1:1:gCpu.NumFaces;
siCells = 1:1:gCpu.NumCells;

gCpu = extrude(gCpu,[1],sHie); % cpu
cpuFaces = siFaces(end)+1:1:gCpu.NumFaces;
cpuCells = siCells(end)+1:1:gCpu.NumCells;

cpuMod.Geometry = gCpu;

% generating mesh
cMsh = generateMesh(cpuMod);

% % % plotting bare CPU
% figure
% pdegplot(gCpu,'FaceLabels','on')
% ylim([-sWid,2*sWid])
% axis equal

figure
pdemesh(cpuMod);
```



No. 1b -- CPU with Sink Model

```
% using same 'slab' matrix

% updating pin geometry description
    % where: gd = [circ xCenter yCenter radius]
pins = zeros(10,1);% initialize pins with zeros to match slab matrix
k = 0;
dist = linspace(.005,.045,n^.5); % pin spacing vector
for i = 1:n^.5
    for j = 1:n^.5
        k = k+1;
        pins(1:4,k) = [1 dist(i) dist(j) pRad]'; % pin geometry desc
 matrix
    end
end

gdSink = [slab,pins]; % geometry description for plane and pins

% updating name-space matrix (ns)
s = 's';
l = 'l';
a = 'a';
b = 'b';
```

```
for i = 1:n

    s = strcat(s,'p');
    l = strcat(l,'i');
    a = strcat(a,'n');
    b = strcat(b,num2str(i));
end
nsSink = char(s,l,a,b);

% updating set formula vector (sf)
sfSink = 'slab';

for i = 1:n
    add = strcat('+pin',num2str(1));
        sfSink = strcat(sfSink,add);
end

% creating 2-d geometry from gd,sf,ns
gSink = decsg(gdSink,sfSink,nsSink);

% merging thermal preallocation with 2-d geometry
gSink = geometryFromEdges(sinkMod,gSink);

% extruding
gSink = extrude(gSink,sHie); % extrude base of sink
pFaces = n+3:1:n*2+2; % pin face ID's
conFaces = [n+2 pFaces]; % convection surface faces
gSink = extrude(gSink,pFaces,pHie); % extrude pins
faces = gSink.NumFaces; % sink faces index
cells = gSink.NumCells; % sink cells index
gSink = extrude(gSink,[1:1:n+1],sHie); % extrude cpu
faces1 = gSink.NumFaces; % cpu faces index
cells1 = gSink.NumCells; % cpu cells index
sFaces = 1:1:faces; % sink faces vector
sCells = 1:1:cells; % sink cells vector
cFaces = 1+faces:1:faces1; % cpu faces vector
cCells = 1+cells:1:cells1; % cpu cells vector


% updating model geometry
sinkMod.Geometry = gSink;

% generating mesh
csMsh = generateMesh(sinkMod);

% % plotting CPU-sink
% figure
% pdegplot(sinkMod.Geometry,'FaceLabels','on')
% ylim([-sWid,2*sWid])
% axis equal

figure
pdemesh(sinkMod)
```



No. 2 -- Finding Convection Coefficient (h)

initalizing given values

```
tInf = 25+273.15; % T infinity (K)
vInf = 5; % V infinity (m/s)
v = interp1([250 300],[11.44E-6 15.89E-6],tInf); % air kin. vis. @
 tInf (m^2/s)
pR = interp1([250 300],[.720 .707],tInf); % prandtl number
pRs = .701; %prandtl at surface

% for flat sink
hFlat = 10.45-vInf + 10*vInf^.5;

% for pins
sT = dist(2)-dist(1); % transverse pitch (m)
sL = sT; % longitudinal pitch (m)
d = pRad*2; % diameter of pin (m)
vMax = sT*vInf/(sT-d); % vMax between pins (m/s)

reDM = vMax*d/v; % reynolds number for max fluid velocity

if 10 < reDM < 10^2
    c1 =.8;m = .4;
elseif 10^2 < reDM < 2E5
    c1 = .27; m = .63;
```

```
elseif 2E5 < reDM < 2E6
    c1 = .021; m = .84;
end
% nusselt number
nu = c1*reDM^m*pR^.36*(pR/pRs)^.25;

% correction constants for nPins < 20
c2 = 1;
if n^.5 == 2
    c2 = .8;
elseif n^.5 == 3
    c2 = .86;
elseif n^.5 == 4
    c2 = .9;
elseif n^.5 == 5
    c2 = .92;
elseif n^.5 == 6
    c2 = interp1([7,10],[.95,.97],9);
elseif n^.5 == 7
    c2 = .95;
end
nuD = nu*c2;

k = interp1([250,300],[22.3E-3 26.3E-3],tInf); % thermal conductivity
 @ tInf
hAv = nuD*k/d; % average convection coefficient (w/(m^2-k))
```

No. 3 -- 3D Transient Conduction Analysis

initializing givens

```
tIni = 25+273.15; % T initial(K)
q = 100; % q (W)
t = 0:60:3600; % time vector (s)
vol = sLen*sWid*sHie;
qDot = q/vol; % heat transfer rate from the CPU
% copper and aluminum material properties (K,rho,C,tMelt)
kCu = 401;rCu = 8933;cCu = 385;mCu = 1358-273.15; % from table A.1
kAl = 237;rAl = 2702;cAl = 903;mAl = 933-273.15;
boltz = 5.67E-8; % boltzmann W/(m^2-K^4)
```

for bare cpu

```
internalHeatSource(cpuMod,qDot,'Cell',cpuCells); % initializing q' for
 cpu
cpuMod.StefanBoltzmannConstant = boltz; % appending boltzmann constant
% initializing thermal properties
thermalProperties(cpuMod,'Cell',siCells,'ThermalConductivity',kCu,...
    'MassDensity',rCu,'SpecificHeat',cCu);
thermalProperties(cpuMod,'Cell',cpuCells,'ThermalConductivity',kAl,...
    'MassDensity',rAl,'SpecificHeat',cAl);
% initializing boundary conditions
```

```
thermalBC(cpuMod,'Face',siFaces(2),'ConvectionCoefficient',hFlat, ...
                'AmbientTemperature',tInf);
% intializing initial conditions
thermalIC(cpuMod,tIni);
% solving thermal model
tResCpu = solve(cpuMod,t);

%figure
%pdeplot3D(cpuMod,'ColorMapData',tResCpu.Temperature(:,end)-273.15)
```

for heat sink

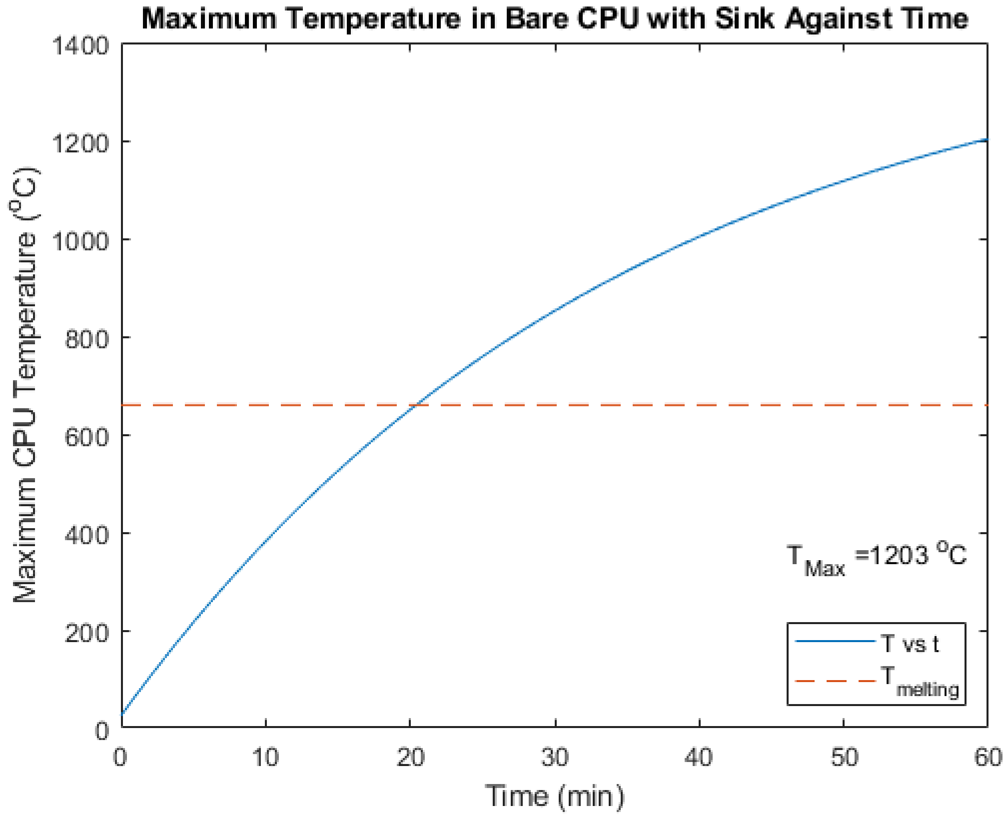
```
internalHeatSource(sinkMod,qDot,'Cell',cCells); % initializing q' for
 cpu
sinkMod.StefanBoltzmannConstant = boltz; % appending boltzmann
 constant
% initializing thermal properties
thermalProperties(sinkMod,'Cell',sCells,'ThermalConductivity',kAl,...
    'MassDensity',rAl,'SpecificHeat',cAl);
thermalProperties(sinkMod,'Cell',cCells,'ThermalConductivity',kCu,...
    'MassDensity',rCu,'SpecificHeat',cCu);
% initializing boundary conditions
thermalBC(sinkMod,'Face',conFaces,'ConvectionCoefficient',hAv, ...
                'AmbientTemperature',tInf);
% intializing initial conditions
thermalIC(sinkMod,tIni);
% solving thermal model
tResSink = solve(sinkMod,t);

%figure
%pdeplot3D(sinkMod,'ColorMapData',tResSink.Temperature(:,end)-273.15)
```

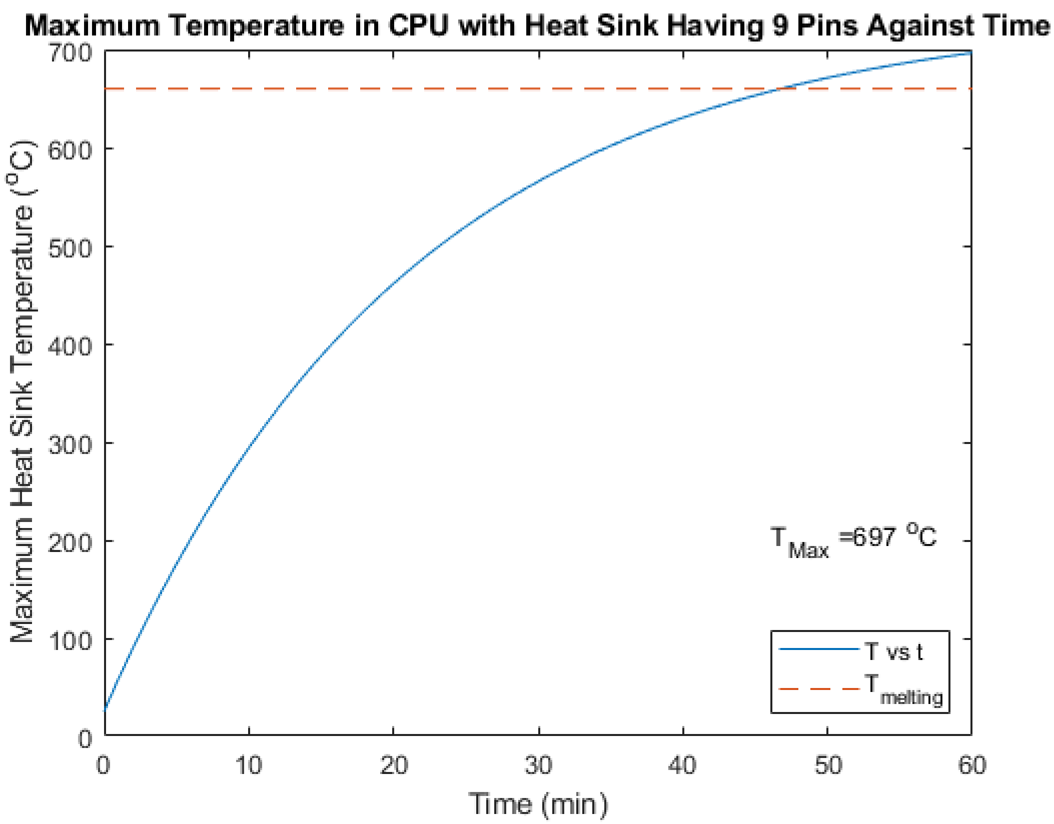
No. 4 -- Maximum Temperature vs Time

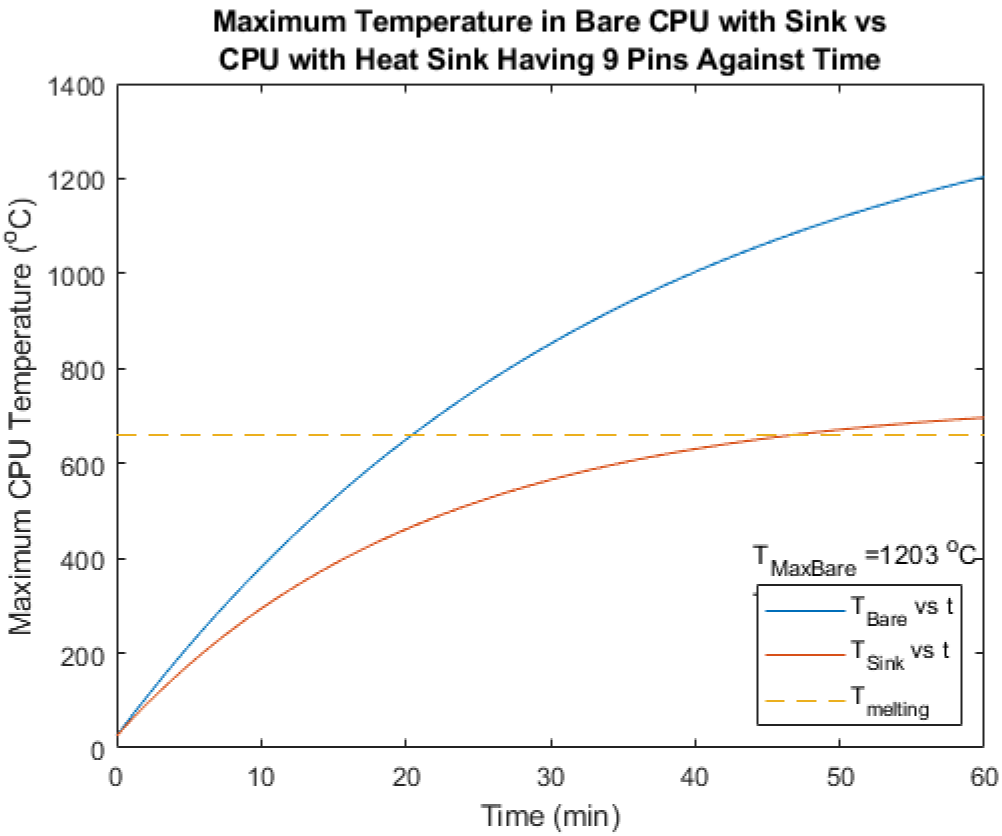
for bare cpu

```
for i = 1:length(tResCpu.Temperature(1,:))
    mTempb(i) = max(tResCpu.Temperature(:,i))-273.15; % max temp at
 each interval
    melt(i) = mAl; % filling melting temp vector
end
figure
plot(t/60,mTempb,t/60,melt,'--')
hold on
legend('T vs t','T_{melting}','Location','southeast')
text(t(47)/60,mTempb(10),strcat('T_{Max} =
 ',num2str(round(mTempb(end),0)),...
    ' ^oC'))
title('Maximum Temperature in Bare CPU with Sink Against Time')
xlabel('Time (min)')
ylabel('Maximum CPU Temperature (^oC)')
hold off
```

```
% for heat sink
for i = 1:length(tResSink.Temperature(1,:))
    mTemp(i) = max(tResSink.Temperature(:,i))-273.15; % max temp at
 each interval
    melt(i) = mAl; % filling melting temp vector
end
figure
plot(t/60,mTemp,t/60,melt,'--')
hold on
legend('T vs t','T_{melting}','Location','southeast')
if n == 9
    xLoc = 47;
    yLoc = 5;
end
if n == 25
    xLoc = 47;
    yLoc = 7;
end
text(t(xLoc)/60,mTemp(7),strcat('T_{Max} =
 ',num2str(round(mTemp(end),0)),...
    ' ^oC'))
title(strcat('Maximum Temperature in CPU with Heat Sink Having',32,...
    num2str(n),' Pins Against Time'))
xlabel('Time (min)')
ylabel('Maximum Heat Sink Temperature (^oC)')
hold off

%combined
figure
plot(t/60,mTempb,t/60,mTemp,t/60,melt,'--')
hold on
legend('T_{Bare} vs t','T_{Sink} vs
 t','T_{melting}','Location','southeast')
if n == 9
    xLoc = t(45)/60;
    yLoc = mTemp(17);
end
if n == 25
    xLoc = t(45)/60;
    yLoc = mTempb(33);
end
text(xLoc,yLoc,strcat('T_{MaxBare} =
 ',num2str(round(mTempb(end),0)),...
    ' ^oC'))
text(xLoc,yLoc-100,strcat('T_{MaxPins} =
 ',num2str(round(mTemp(end),0)),...
    ' ^oC'))
title(strcat({'Maximum Temperature in Bare CPU with Sink vs',...
    ['CPU with Heat Sink Having',32,num2str(n),' Pins Against
 Time']}))
xlabel('Time (min)')
ylabel('Maximum CPU Temperature (^oC)')
hold off
```


Maximum Temperature in Bare CPU with Sink Against Time
Maximum CPU Temperature (°C)
Time (min)
T Max =1203 °C
T vs t
T melting
0
200
400
600
800
1000
1200
1400
0
10
20
30
40
50
60


Maximum Temperature in CPU with Heat Sink Having 9 Pins Against Time
Maximum Heat Sink Temperature (°C)
Time (min)
T Max =697 °C
T vs t
T melting
0
100
200
300
400
500
600
700
0
10
20
30
40
50
60



No 5 -- Sink Steady State

```
tSS = [1, 7200]; % two hour time solution for more accurate steady
 state
tResSteadyBare = solve(cpuMod,tSS);
tResSteady = solve(sinkMod,tSS);
tSteadyBare = max(tResSteadyBare.Temperature(:,end))-273.15
tSteady = max(tResSteady.Temperature(:,end))-273.15
```

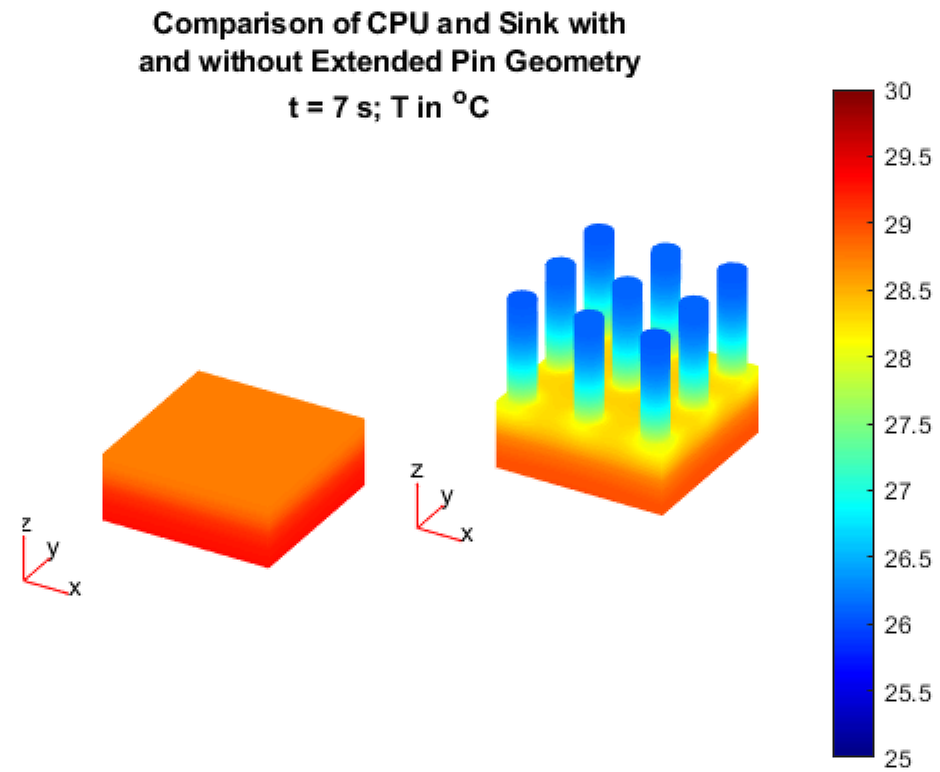
```
tSteadyBare =

   1.4169e+03


tSteady =

  736.2387
```

No. 6 -- Common Plot

offsetting cpu and recalculating -- same as 1a/3a without comments

```
oMod = createpde('thermal','transient');
offX = sLen+.025;
```

```
offY = sLen+.025;
slab = [3 4 -.025 -.025 -offX -offX -.025 -offY -offY -.025]'; %
 offset of bare cpu for plot
gdO = slab;
oCpu = decsg(gdO,sfCpu,nsCpu);
oCpu = geometryFromEdges(oMod,oCpu);
oCpu = extrude(oCpu,sHie);
oFaces = 1:1:oCpu.NumFaces;
oCells = 1:1:oCpu.NumCells;
oCpu = extrude(oCpu,[1],sHie);
oCFaces = oFaces(end)+1:1:oCpu.NumFaces;
oCCells = oCells(end)+1:1:oCpu.NumCells;
oMod.Geometry = oCpu;
oMsh = generateMesh(oMod);
% reentering conditions
internalHeatSource(oMod,qDot,'Cell',oCCells);
oMod.StefanBoltzmannConstant = boltz;
thermalProperties(oMod,'Cell',oCCells,'ThermalConductivity',kCu,...
    'MassDensity',rCu,'SpecificHeat',cCu);
thermalProperties(oMod,'Cell',oCells,'ThermalConductivity',kAl,...
    'MassDensity',rAl,'SpecificHeat',cAl);
thermalBC(oMod,'Face',oFaces(2),'ConvectionCoefficient',hFlat, ...
                'AmbientTemperature',tInf);
thermalIC(oMod,tIni);
tResO = solve(oMod,[1 7]);
tResOs = solve(sinkMod,[1 7]);
% plotting
figure
pdeplot3D(sinkMod,'ColorMapData',tResOs.Temperature(:,2)-273.15)
hold on
pdeplot3D(oMod,'ColorMapData',tResO.Temperature(:,2)-273.15)
caxis([25 30])
title({'Comparison of CPU and Sink with',...
    'and without Extended Pin Geometry','t = 7 s; T in ^oC'})
```



Published with MATLAB® R2021a