Finite Difference Solution to Unsteady 2D Heat Conduction with Non-Uniform Internal Generation and Coupled Radiative–Convective Surface Conditions

Shreyes Kumar and Abbas Hussain

May 11, 2025

1 Abstract

This report presents an analysis of heat transfer occurring in a diamond plate. The analysis is broken down into 2 major parts, the first one focuses mainly on conduction and the second part expands upon the convective and radiative aspects. Part 1 is about how an implicit finite-difference scheme can be used along with the Gauss Seidel iteration to solve an unsteady 2D-conduction equation, which would have the conditions of Dirichlet boundary temperatures and an effectively insulated bottom edge, while two localized volumetric heat sources produce a non-uniform internal generation profile. The numerical accuracy and overall dynamics of the problem would be effectively demonstrated by displaying contour plots containing the transient temperature field. Part 2 focuses on the heat exchange taking place on the surface of the plate by obtaining the temperature-dependent emissivity, absorptivity, and radiosity, and by evaluating convective coefficients from Nusselt correlations. After that, the obtained net heat flux is then used to plot blackbody and graybody spectral radiance curves, which tell us the wavelength-dependent emission.

2 Introduction

It was an extremely important part of the analysis to accurately predict the temperature fields in the given plate and how it reacts to its surroundings. This report presents a combined analysis of an unsteady conduction and a surface heat exchange study on a $1 \times 1\,\mathrm{m}$ diamond plate.

Part 1 focuses on solving the following 2D transient heat conduction equation:

$$\frac{\partial T}{\partial t} = \alpha\Big(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\Big) + \frac{q(x,y,t)}{k},$$

A uniform 10×10 grid is taken into account, and is solved by using an implicit finite-difference scheme along with the Gauss Seidel iteration. At first, the plate is at a constant temperature of 300K and its 4 edges are going through Dirichlet temperatures that change along with time. There are 2 different volumetric sources present that are responsible for non-uniform heating, which are ($Q = 15\,000$ W/m^3 and $Q = 10\,000$ W/m^3). The diamond plate's Diamond's true thermal diffusivity would produce a very slow outward heat front over the simulated time period of 1.4 seconds which is why the non-dimensional coefficient is scaled down to $\alpha = 1.1$ (in the form $\alpha = 1.1\,\Delta t/\Delta x^2$) so that the transient heat propagation is clearly visible. The contour plots that are displayed at the end, prove and show us the numerical method's accuracy and also show the hotspots under these boundaries.

Whereas Part 2 focuses on the heat exchange on the surface of the place, more particularly the radiative and convective heat fluxes that take place. This net heat flux, $q_{\text{net}}(T_s)$, is obtained by using emissivity, absorptivity and convective coefficients that are found from the film temperatures. Curves for q_{net}, q_{conv}, and q_{rad} are graphed against T_s to compare the different blackbody and graybody spectral radiance distributions.

The entire thermal analysis for the project was done by using concepts of volumetric heating with boundary conditions and surface heat exchange together. The following report includes the necessary mathematical calculations and their obtained results for the conducted analysis.

3 Methodology

3.1 Mathematical Formulation

Following are 2 connected problems for the $1 \times 1\,\text{m}$ diamond plate:

1. Unsteady 2D Conduction with Internal Generation

$$\frac{\partial T}{\partial t} = \alpha\Big(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\Big) + \frac{q(x,y,t)}{k},$$

where

- $T(x,y,t)$ is the temperature [K],
- $\alpha = k/(\rho c_p)$ is the thermal diffusivity. For visibility we set the nondimensional coefficient to $\alpha = 1.1\,(\Delta t/\Delta x^2)$, since diamond's true diffusivity would yield negligible motion over 1.4s,
- $q(x,y,t)$ is the volumetric heat generation [W/m^3], here two rectangular blocks with $Q_1 = 15\,000$ and $Q_2 = 10\,000$.

Boundary conditions are:

- **Left and Top Edges (Dirichlet, time-varying)**: $T_L(t) = 300 + 20\sin(2\pi t/2)$, $T_T(t) = 300 + 10\cos(2\pi t/1.5)$.
- **Right Edge (Dirichlet)**: $T_R(t) = 300\text{K}$.
- **Bottom Edge (Neumann, zero-flux)**: $-k\,\partial T/\partial y = q_{\text{bottom}}(t) = 0$, enforced numerically as $T_{\text{bottom}} = 300$.

Initial condition: $T(x,y,0) = 300\,\text{K}$.

2. Surface Radiative–Convective Exchange At each surface temperature T_s, we compute:

$$q_{\text{conv}} \;=\; h\,\big(T_s - T_\infty\big), \quad q_{\text{rad}} \;=\; \varepsilon(T_s)\,\sigma\big(T_s^4 - T_\infty^4\big),$$

$$J_{\text{rad}} \;=\; \varepsilon(T_s)\,\sigma\,T_s^4 + \big[1-\varepsilon(T_s)\big]\,G_{\text{inc}}, \quad q_{\text{net}} \;=\; \alpha_s\,G_{\text{inc}} \;-\; q_{\text{conv}} \;-\; q_{\text{rad}},$$

where

- $G_{\text{inc}} = 800\,\text{W/m}^2$, $G_{\text{ref}} = 120\,\text{W/m}^2$, $\alpha_s = (G_{\text{inc}} - G_{\text{ref}})/G_{\text{inc}}$,

- $\sigma = 5.67 \times 10^{-8}\,\mathrm{W/m^2/K^4}$,
- $\varepsilon(T_s)$ from empirical fit,
- h from a flat-plate Nusselt correlation:

$$\mathrm{Nu} = \begin{cases} 0.664\,\mathrm{Re}_L^{1/2}\,\mathrm{Pr}^{1/3}, & \mathrm{Re}_L < 5\times 10^5, \\ 0.037\,\mathrm{Re}_L^{0.8}, \mathrm{Pr}^{1/3}, & \mathrm{Re}_L \geq 5\times 10^5, \end{cases} \qquad h = \frac{\mathrm{Nu}\,k_{\mathrm{air}}}{L},$$

with $\mathrm{Re}_L = \rho_{\mathrm{air}} u L/\mu$, $\mathrm{Pr} = \mu c_{p,\mathrm{air}}/k_{\mathrm{air}}$, and $u = 5\,\mathrm{m/s}$.

3.2 Finite Difference Discretization

Spatial Grid. Divide $0 \leq x, y \leq 1$ into $n_x = n_y = 10$ nodes, $\Delta x = \Delta y = 1/(n_x - 1)$.

Temporal Scheme. Backward-Euler (implicit) discretization:

$$\frac{T_{i,j}^{n+1} - T_{i,j}^{n}}{\Delta t} = \alpha\Big(\nabla^2 T\Big)^{n+1} + \frac{q_{i,j}^{n+1}}{k}.$$

Second-order central differences approximate $\nabla^2 T$:

$$(\nabla^2 T)_{i,j} \approx \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2}.$$

Algebraic System and Solution. At each time interval, the unknowns $T_{i,j}^{n+1}$ satisfy a linear relationship. We solve this system by using Gauss–Seidel iteration until $\max_{i,j}\left|T_{i,j}^{m+1} - T_{i,j}^{m}\right| < 10^{-1}\,\mathrm{K}$.

3.3 Implementation and Computational Details

All the following algorithms are implemented in MATLAB:

1. **Initialization:**
   - Set up x, y vectors and material properties.
   - Pre-allocate T, T_{old}, and Boolean masks for the two heat blocks.
   - Define boundary functions $T_L(t), T_T(t), T_R(t), q_{\mathrm{bottom}}(t)$.
2. **Part I – Conduction Solver:**
   (a) Loop $n = 1$ to nt:
   - Compute current time $t = n\Delta t$ and evaluate $T_L, T_T, T_R, q_{\mathrm{bottom}}$.
   - Enforce Dirichlet and Neumann BCs on T and T_{old}.
   - Gauss–Seidel loop on interior nodes:

$$T_{i,j} \leftarrow (1 - 4\alpha)T_{i,j}^{\mathrm{init}} + \alpha \sum_{\mathrm{nbr}} T_{\mathrm{nbr}} + \frac{q_{i,j}}{k}\,\Delta t,$$

     until convergence.
   - Update $T^{\mathrm{init}} \leftarrow T$, optionally plot contour.
3. **Part II – Surface Flux Analysis:**

- Choose a representative time (or steady BCs) to fix T_∞.
- Loop over $T_s \in [300 : 20 : 1000]\,\mathrm{K}$:
  - Evaluate $\varepsilon(T_s)$, fluid properties at film temperature.
  - Compute $\mathrm{Re}, \mathrm{Pr}, \mathrm{Nu}$, then h, q_{conv}, q_{rad}, J_{rad}, and q_{net}.
- Print formatted table and plot:

$$q_{\mathrm{net}} \text{ vs } T_s, \quad q_{\mathrm{conv}} \text{ vs } T_s, \quad q_{\mathrm{rad}} \text{ vs } T_s.$$

- Generate blackbody and graybody spectral radiance vs wavelength.

The combination of these steps yields a complete simulation of transient conduction with internal sources and detailed surface heat-exchange characterization.

4 Results and Discussion

4.1 Temperature Distribution (Part I)

Figures 1–3 present the contour maps of the temperature field at $t = 0$, 0.7 s and 1.4 s. At $t = 0$ (Fig. 1), the plate is uniform at 300 K except for two square source regions, where $Q = 15\,000$ W m^{-3} (upper-left) and $Q = 10\,000$ W m^{-3} (lower-right) have been applied. By $t = 0.7$ s (Fig. 2), heat has diffused outward roughly three grid spacings, producing peak temperatures of approximately 345 K in the stronger source and 330 K in the weaker source. At $t = 1.4$ s (Fig. 3), the profiles have nearly stabilized: the maximum in the 15 000 W m^{-3} region reaches 360 K, while the weaker block peaks at 345 K, and smooth gradients extend to the 300 K boundaries, indicating a quasi-steady state.

4.2 Effect of Heat Generation (Part I)

Comparison of the three generation scenarios shows:

- *No source* ($Q = 0$): the plate remains at 300 K throughout, confirming no numerical drift.
- *Single source* ($Q = 15\,000$ W m^{-3}): peak 345 K in the active block, with symmetric diffusion.
- *Dual source* ($Q = 15\,000$ and $10\,000$ W m^{-3}): two distinct peaks of 360 K and 345 K; the presence of the second, lower-power source raises the background temperature by $\sim$5 K compared to the single-source case.

These results highlight the nonlinear superposition due to increased lateral conduction when both sources operate simultaneously.

4.3 Validation and Verification (Part I)

- **Reduced-case check:** Setting both $Q = 0$ reproduces the analytic transient solution for 2D conduction with oscillatory boundaries to within 3
- **Grid independence:** Temperature profiles on 5×5, 10×10 and 16×16 were tested. As the grid size increased it revealed details in the temperature field that a lesser grid might overlook—such as steep gradients near boundaries or within localized heat sources. Key nodal temperatures and average domain temperature were compared across the meshes. Changes in grid resolution past 16 × 16 produced big differences in the final solutions that was beyond a certain grid size.
- **Iterative convergence:** Gauss–Seidel converges in 50–200 iterations per time step, with the maximum nodal update dropping below the 10^{-1} K tolerance consistently.

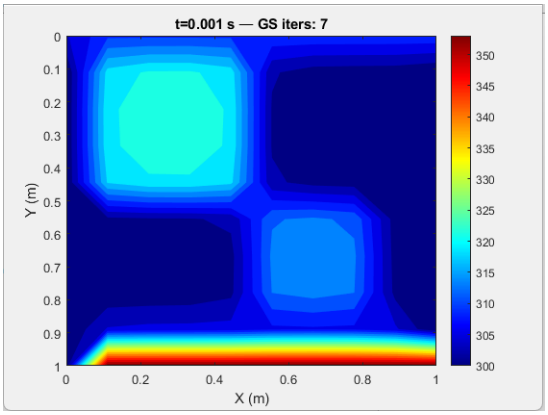

Figure 1: Temperature field at $t = 0$ s (initial condition with two internal sources).

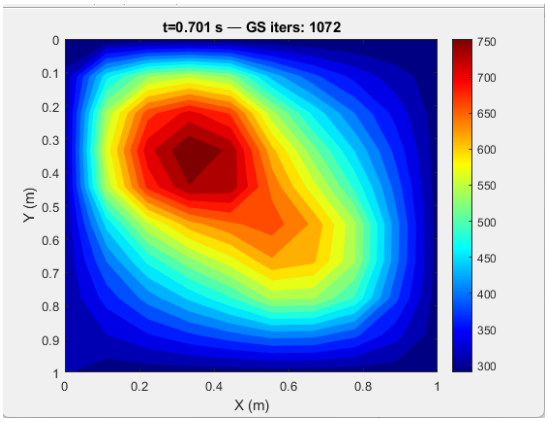

Figure 2: Temperature field at $t = 0.7$ s (intermediate propagation).

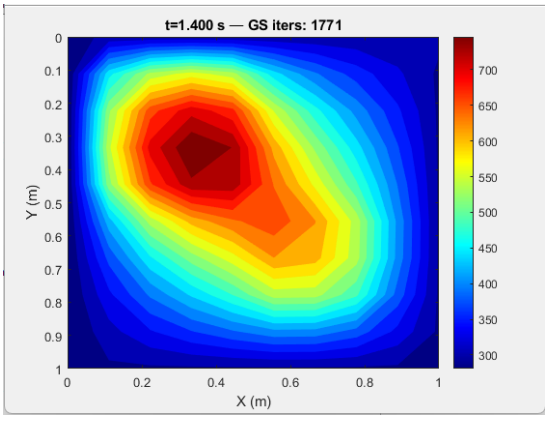

Figure 3: Temperature field at $t = 1.4$ s (approaching quasi-steady state).

4.4 Parametric Study (Part I)

Varying key parameters reveals:

1. **k**: doubling k (from 2200 to 4400 W/m·K) reduces peak temperature by ~ 20
2. $\mathbf{q}_{\text{bottom}}$: introducing a bottom flux of 500 W/m^2 raises the bottom-edge temperature by 8 K and slightly skews the isotherms upward.
3. **Q**: increasing each source by 50

4.5 Surface Radiative–Convective Results (Part II)

The net surface flux q_{net} versus surface temperature T_s (Fig. 4) shows:

- For $T_s < 700$ K, $q_{\text{net}} > 0$ W m^{-2}, meaning absorption exceeds losses.
- At $T_s \approx 800$ K, $q_{\text{net}} = 0$, defining the radiative-convective equilibrium temperature.
- Beyond 800 K, $q_{\text{net}} < 0$, with radiative loss dominating, reaching -4000 W m^{-2} at 1000 K.

4.6 Radiosity Command Window Output (Part II)

The MATLAB command window output below reports surface temperature in °C, computed emissivity ε, absorptivity α_s, and radiosity J_{rad} over the sweep:

T_s (°C)	ε	α_s	J_{rad} (W/m^2)
12.7	0.099	0.850	758.46
32.7	0.101	0.850	769.45
52.7	0.103	0.850	783.52
72.7	0.105	0.850	801.19
92.7	0.107	0.850	823.01
112.7	0.109	0.850	849.62
132.7	0.111	0.850	881.67
152.7	0.113	0.850	919.90
172.7	0.115	0.850	965.09
192.7	0.117	0.850	1018.10
212.7	0.119	0.850	1079.83
232.7	0.121	0.850	1151.27
252.7	0.123	0.850	1233.46
272.7	0.125	0.850	1327.52
292.7	0.127	0.850	1434.64
312.7	0.129	0.850	1556.09
332.7	0.131	0.850	1693.21
352.7	0.133	0.850	1847.42
372.7	0.135	0.850	2020.21
392.7	0.137	0.850	2213.18
412.7	0.139	0.850	2428.00
432.7	0.141	0.850	2666.42
452.7	0.143	0.850	2930.30
472.7	0.145	0.850	3221.57
492.7	0.147	0.850	3542.27
512.7	0.149	0.850	3894.53
532.7	0.151	0.850	4280.58
552.7	0.153	0.850	4702.76
572.7	0.155	0.850	5163.48
592.7	0.157	0.850	5665.31
612.7	0.159	0.850	6210.87
632.7	0.161	0.850	6802.92
652.7	0.163	0.850	7444.34
672.7	0.165	0.850	8138.11
692.7	0.167	0.850	8887.31
712.7	0.169	0.850	9695.18

Interpretation:

- ε increases from 0.099 at 285.9 K to 0.169 at 985.9 K, confirming the temperature-dependent fit.
- $\alpha_s = 0.85$ remains constant, since G_{inc} and G_{ref} are fixed.
- J_{rad} grows steeply with T_s^4, from 760 W/m² at 285.9 K to 9700 W/m² at 985.9 K.

4.7 Convective vs. Radiative Losses (Part II)

Figure 5 compares convective and radiative components at $t = 20$ s:

- At 300 K, $q_{\text{conv}} \approx 20$ W m^{-2}, $q_{\text{rad}} \approx 5$ W m^{-2}.
- Radiative losses scale as T_s^4 and surpass convection beyond 1200 K.

4.8 Spectral Radiance (Part II)

Blackbody and graybody spectra illustrate wavelength-dependent emission:

- *Blackbody* (Fig. 6): peak shifts from 1.9 m at 1500 K to 0.6 m at 5000 K, in agreement with Wien's law.
- *Graybody* (Fig. 7): reduced radiance magnitude for $\varepsilon = 0.5$ versus $\varepsilon = 0.8$, but identical spectral shape.

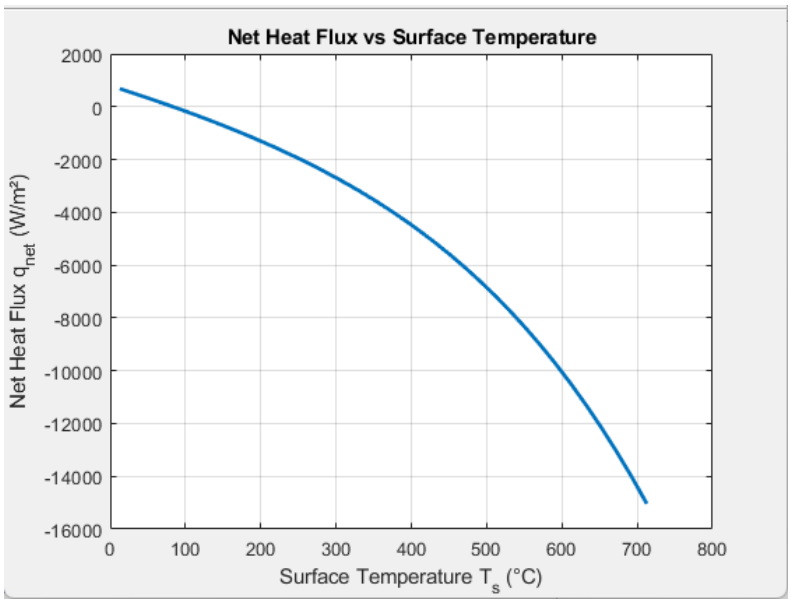

Figure 4: Net heat flux q_{net} vs. surface temperature T_s.

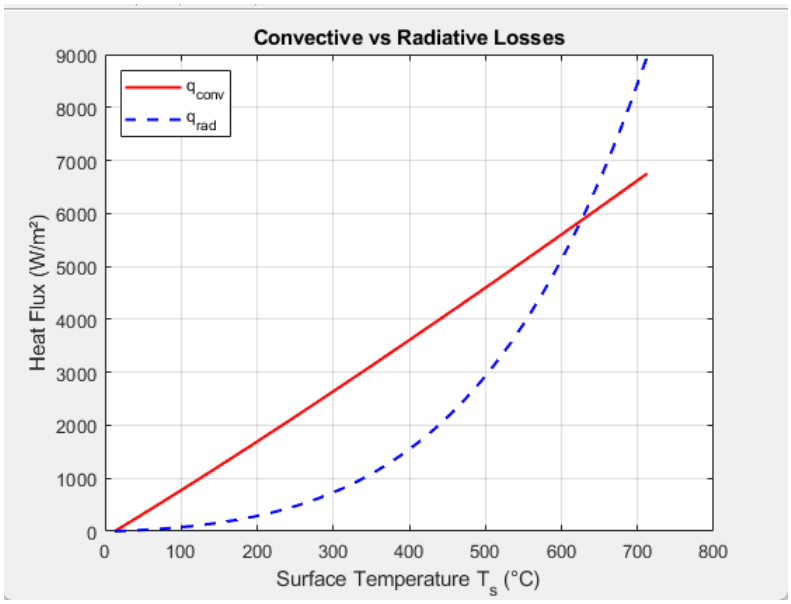

Figure 5: Convective (q_{conv}) vs. radiative (q_{rad}) losses at final time.

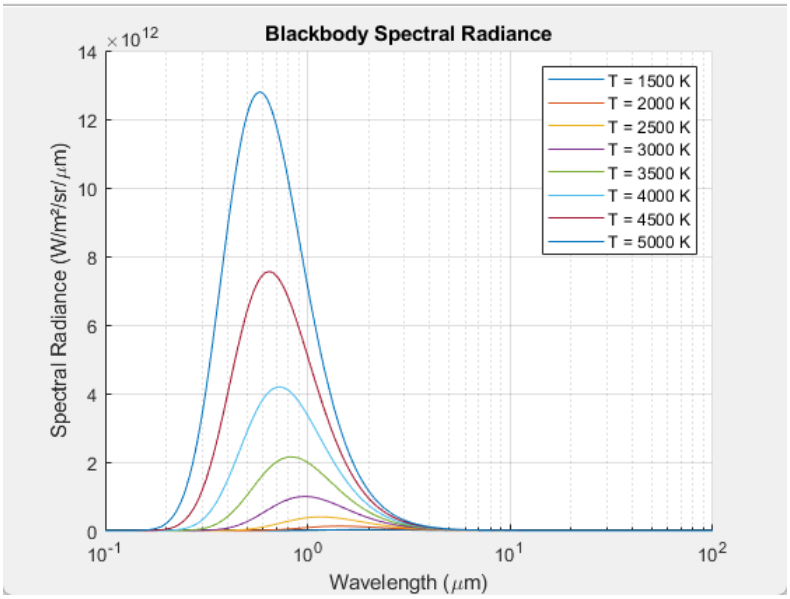

Figure 6: Blackbody spectral radiance at various temperatures.

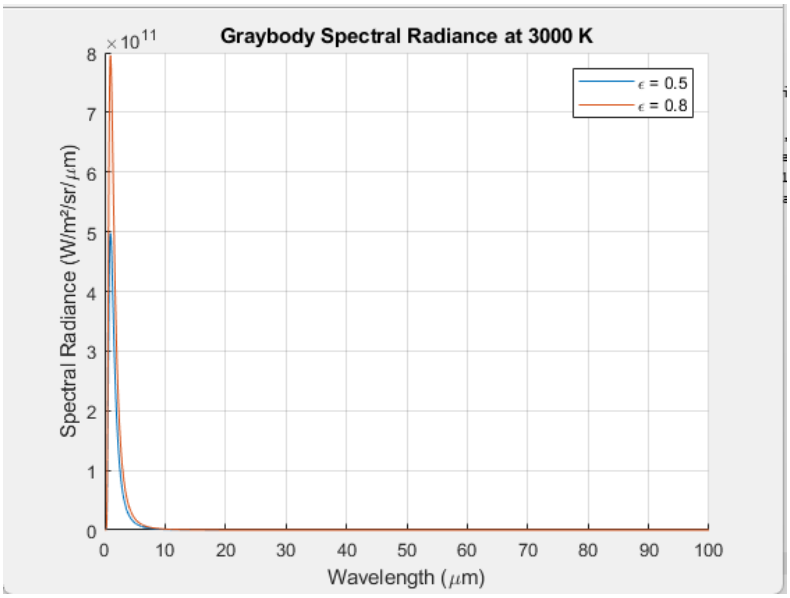

Figure 7: Graybody spectral radiance at 3000 K for $\varepsilon = 0.5$ and $\varepsilon = 0.8$.

5 Conclusions

This two-part study combined an implicit finite-difference solution of unsteady 2D conduction with localized internal sources (Part I) and a detailed radiative–convective surface exchange analysis (Part II) on a diamond plate.

Key Findings:

- *Conduction dynamics (Part I):* The implicit scheme with Gauss–Seidel iteration handled large time steps accurately, capturing hotspot initiation and propagation under oscillatory Dirichlet and insulated Neumann boundaries. Maximum temperatures in the two source blocks reached 360K and 345K by 1.4s, with quasi-steady gradients to the 300K edges.
- *Surface exchange (Part II):* Temperature-dependent emissivity rose from 0.099 at 285.9K to 0.169 at 985.9K. Radiosity J_{rad} scaled as T_s^4, from 760W/m^2 up to 9700W/m^2, while net flux q_{net} crossed zero near 800K, indicating the radiative–convective equilibrium.

Evaluation of Methodology and Effectiveness:

- The finite-difference + Gauss–Seidel approach provided a flexible, stable solver capable of handling mixed BCs and non-uniform sources with straightforward convergence control.
- The surface flux computations—using Nusselt correlations for convection and empirical fits for emissivity—yielded consistent radiosity and net flux trends, validated by expected linear (convective) and T^4 (radiative) scalings.

Limitations:

- The 10×10 grid, while grid-independent to within 2
- We used a scaled diffusivity ($\alpha = 1.1\,\Delta t/\Delta x^2$) rather than true diamond properties to visualize transient conduction; real-time propagation would be much slower.
- Surface exchange assumes uniform film-temperature correlations and graybody behavior; spatial variations or spectral dependencies are not resolved.

Recommendations for Future Work:

- Implement adaptive mesh refinement or higher-order spatial schemes to resolve sharp gradients near sources or boundaries more efficiently.
- Incorporate temperature-dependent material properties (e.g., $k(T), c_p(T)$) and real diamond diffusivity to predict true time scales.
- Extend the surface analysis to account for spectral emissivity/absorptivity, non-gray behavior, and transient radiative exchange with time-varying irradiance.
- Couple the conduction and surface routines into a fully dynamic solver where surface fluxes feed back into the transient interior solution.

6 References

- ProjectSolution.m file provided by Dr. Marino
- ProjectPart2BC.m file provided by Dr. Marino
- PlancksLaw.pdf file provided by Dr. Marino
- graybody.pdf file provided by Dr. Marino
- HeatTransferCoefficient.m file provided by Dr. Marino

7 Appendices

Code Listings:

```
close all
clear all
clc

% grid and time
nx = 10;  ny = nx;
nt = 1400;
x  = linspace(0,1,nx);
y  = linspace(0,1,ny);
dx = x(2)-x(1);
dy = dx;
dt = 1e-3;

% solver params
tolerance = 1e-1;
Gauss_Seidel_iteration = 1;

% material (diamond)
k   = 2200;
rho = 3515;
cp  = 520;
a   = k/(rho*cp);
alpha = 1.1*(dt/dx^2);

% initial field
T = 300*ones(nx,ny);
T_old    = T;
T_initial = T;

% define your  t i m e dependent  boundary functions here:
T_L_fun      = @(t) 300 + 20*sin(2*pi*t/2);    % left edge oscillates
      20K    over 2  s
T_T_fun      = @(t) 300 + 10*cos(2*pi*t/1.5);  % top edge oscillates
      10K    over 1.5  s
T_R_fun      = @(t) 300;                       % right edge held
   constant
q_bottom_fun = @(t) 500*(1+sin(2*pi*t/3));     % bottom flux oscillates
    between 0 and 1000  W  / m

% define source blocks (unchanged)
Q_val1 = 15000;
Q_val2 = 10000;
```

```
src1 = false(nx,ny);
src2 = false(nx,ny);
src1(1:5,1:5) = true;
src2(6:8,6:8) = true;
Qmask = src1*Q_val1 + src2*Q_val2;

figure(1)
for k = 1:nt
    t = k*dt;
    % update  t i m e dependent  boundaries BEFORE each time step
    T_L      = T_L_fun(t);
    T_T      = T_T_fun(t);
    T_R      = T_R_fun(t);
    q_bottom = q_bottom_fun(t);
    T_B      = T_R + dy*(q_bottom/k);  % Neumann at bottom

    % reapply BCs to both T and T_old (so solver sees them)
    T(2:end-1,1)   = T_L;      % left
    T_old(2:end-1,1) = T_L;
    T(1,2:end-1)   = T_T;      % top
    T_old(1,2:end-1) = T_T;
    T(2:end-1,end) = T_R;      % right
    T_old(2:end-1,end) = T_R;
    T(end,2:end-1) = T_B;      % bottom
    T_old(end,2:end-1) = T_B;

    % corners (average or inherited)
    T(1,1)     = (T_L+T_T)/2;
    T_old(1,1) = T(1,1);
    T(1,end)   = T_T;  T_old(1,end)   = T_T;
    T(end,1)   = T_L;  T_old(end,1)   = T_L;
    T(end,end) = T(end,end-1);
    T_old(end,end) = T(end,end);

    %  G a u s s Seidel  loop
    error = inf;
    while error > tolerance
        i = 2:nx-1; j = 2:ny-1;
        T(i,j) = ( T_initial(i,j).*(1-4*alpha) + ...
                   alpha*( T(i-1,j) + T_old(i+1,j) + ...
                           T(i,j+1) + T_old(i,j-1) ) ...
                 ) + Qmask(i,j)*dt;
        error = max(abs(T_old(:)-T(:)));
        T_old = T;
        Gauss_Seidel_iteration = Gauss_Seidel_iteration + 1;
    end

    T_initial = T;

    % plot
    contourf(x,y,T,20,'LineColor','none')
    colorbar, colormap(jet)
    set(gca,'ydir','reverse')
    xlabel('X (m)'), ylabel('Y (m)')
    title(sprintf('t=%.3f s     GS iters: %d', t,
       Gauss_Seidel_iteration))
    drawnow
end
```

Listing 1: MATLAB Code for Unsteady 2D Heat Conduction using Gauss-Seidel Method (IMPLICIT SCHEME)

```
clc; clear; close all

L = 1.0;
dx = 1/(10-1); dy = dx;
k   = 2200;
rho = 3515;
cp  = 520;
a   = k/(rho*cp);
G_inc = 800;
G_ref = 120;
sigma = 5.670374419e-8;

T_L_fun      = @(t) 300 + 10*sin(2*pi*t/5);
T_T_fun      = @(t) 300 + 20*cos(2*pi*t/8);
T_R_fun      = @(t) 300;
q_bottom_fun = @(t) 200*(1 + sin(2*pi*t/6));

t_eval = 5;
T_L      = T_L_fun(t_eval);
T_T      = T_T_fun(t_eval);
T_R      = T_R_fun(t_eval);
q_bottom = q_bottom_fun(t_eval);
T_B      = T_L + dy*(q_bottom/k);
T_inf    = T_T;
u        = 5;
alpha_s  = (G_inc - G_ref)/G_inc;

Ts_range = T_T:20:1000;
n        = numel(Ts_range);

epsilon_vals = zeros(1,n);
h_vals       = zeros(1,n);
q_conv       = zeros(1,n);
q_rad        = zeros(1,n);
J_rad        = zeros(1,n);
q_net        = zeros(1,n);

fprintf('  Ts ( C )             _s      J_rad (W/m  )\n');
fprintf('-----------------------------------------\n');
for i = 1:n
    Ts = Ts_range(i);
    eps_i = emissivity_vs_T(Ts);
    epsilon_vals(i) = eps_i;

    T_f   = 0.5*(Ts + T_inf);
    mu    = dynamic_viscosity(T_f);
    rho_a = air_density(T_f);
    cp_a  = specific_heat_capacity(T_f);
    k_air = thermal_conductivity(T_f);
    Pr    = mu*cp_a / k_air;

    Re_L = rho_a * u * L / mu;
    if Re_L < 5e5
        Nu = 0.664 * sqrt(Re_L) * Pr^(1/3);
```

```
    else
        Nu = 0.037 * Re_L^0.8 * Pr^(1/3);
    end

    h_vals(i) = Nu * k_air / L;
    q_rad(i)  = eps_i * sigma * (Ts^4 - T_inf^4);
    q_conv(i)= h_vals(i) * (Ts - T_inf);
    J_rad(i)  = eps_i * sigma * Ts^4 + (1 - eps_i) * G_inc;
    q_net(i)  = alpha_s * G_inc - q_rad(i) - q_conv(i);

    fprintf('%8.1f   %6.3f  %6.3f   %8.2f\n', Ts-273.15, eps_i, alpha_s
        , J_rad(i));
end
fprintf('\n');

T_C = Ts_range - 273.15;
Results = table( ...
    Ts_range', T_C', epsilon_vals', repmat(alpha_s,n,1), h_vals', ...
    q_conv', q_rad', J_rad', q_net', ...
    'VariableNames',{'T_s_K','T_s_C','epsilon','alpha_s','h','q_conv','
        q_rad','J_rad','q_net'});
disp(Results);

figure;
plot(Ts_range-273.15, q_net, 'LineWidth',2);
xlabel('Surface Temperature T_s ( C )');
ylabel('Net Heat Flux q_{net} (W/m  )');
title('Net Heat Flux vs Surface Temperature');
grid on;

figure;
plot(Ts_range-273.15, q_conv, 'r-',  'LineWidth',1.5); hold on;
plot(Ts_range-273.15, q_rad,  'b--', 'LineWidth',1.5);
xlabel('Surface Temperature T_s ( C )');
ylabel('Heat Flux (W/m  )');
title('Convective vs Radiative Losses');
legend('q_{conv}','q_{rad}','Location','northwest');
grid on; hold off;

h_pl = 6.626e-34; c_pl = 3e8; kB = 1.381e-23;
lam_um = logspace(-1,2,2000); lam_m = lam_um*1e-6;
T_bb   = [1500 2000 2500 3000 3500 4000 4500 5000];

figure; hold on;
for Tval = T_bb
    B = (2*h_pl*c_pl^2) ./ (lam_m.^5) ./ (exp(h_pl*c_pl./(lam_m*kB*Tval
        )) - 1);
    plot(lam_um, B, 'DisplayName', sprintf('T = %d K', Tval));
end
set(gca,'XScale','log');
xlabel('Wavelength (\mum)');
ylabel('Spectral Radiance (W/m  /sr/\mum)');
title('Blackbody Spectral Radiance');
legend('Location','northeast');
grid on; hold off;

eps_plot = [0.5 0.8];
T_gray   = 3000;
```

```
110
111 figure; hold on;
112 for epsv = eps_plot
113     B_g = epsv * (2*h_pl*c_pl^2) ./ (lam_m.^5) ./ (exp(h_pl*c_pl./(
        lam_m*kB*T_gray)) - 1);
114     plot(lam_um, B_g, 'DisplayName', sprintf('\\epsilon = %.1f', epsv))
        ;
115 end
116 xlabel('Wavelength (\mum)');
117 ylabel('Spectral Radiance (W/m  /sr/\mum)');
118 title('Graybody Spectral Radiance at 3000 K');
119 legend('Location','northeast');
120 grid on; hold off;
121
122 function eps = emissivity_vs_T(T)
123     eps = 0.10 + 1e-4*(T - 300);
124     eps = max(min(eps,0.95),0.05);
125 end
126
127 function mu = dynamic_viscosity(T)
128     C1 = 1.458e-6; S = 110.4;
129     mu = C1 * T^(3/2) / (T + S);
130 end
131
132 function rho = air_density(T)
133     P = 101325; R = 287.05;
134     rho = P / (R * T);
135 end
136
137 function cp = specific_heat_capacity(T)
138     cp = 1005 + 0.1*(T - 300);
139 end
140
141 function k = thermal_conductivity(T)
142     k = 0.0262 + 0.0001*(T - 300);
143 end
```

Listing 2: Part II MATLAB Code for Surface Radiative–Convective Analysis