High Energy Density Radiative Transfer Benchmark Solutions via Heterogeneous Computing

TEXAS A&M ENGINEERING

CERT Center for Exascale Radiation Transport

Daniel A. Holladay, Ryan G. McClarren

Department of Nuclear Engineering, Texas A&M University, College Station, TX, USA 77843

3-T

- The three temperature (3-T) equations of thermal radiative transfer model the exchange of energy between photons, electrons, and ions in a dense plasma.
- In particular, radiation is modeled with full transport and electrons are modeled with a conduction (diffusion) model.
- In 1-D cartesian coordinates under the gray approximation, the equations are:

$$\frac{1}{c}\frac{\partial I}{\partial t} + \mu\frac{\partial I}{\partial x} + \sigma_a I = \sigma_a \frac{acT_e^4}{2} + Q_r$$

$$c_{v,e}\frac{\partial T_e}{\partial t} - \frac{\partial}{\partial x}\left(\kappa_e \frac{\partial T_e}{\partial x}\right) = c\sigma_a\left(E_r - aT_e^4\right) + \gamma_{ei}\left(T_i - T_e\right) + Q_e$$

$$c_{v,i}\frac{\partial T_i}{\partial t} = \gamma_{ei}\left(T_e - T_i\right) + Q_i$$

Solution Method

- Non-dimensionalize equations
- Linearize equations
  - $\kappa_e \propto T_e^3$, $c_{v,i} \propto T_i^3$, $c_{v,e} \propto T_e^3$, and $\gamma_{ei} \propto T_i^3 + T_i^2 T_e + T_i T_e^2 + T_e^3$
  - All equations are now in terms of T_i^4, T_e^4, I, and it's angular integral, E_r.
- Perform Fourier transform in spatial variable and Laplace transform in temporal variable.
  - Linear system of PDE's becomes a linear system of equations.
- Solve for transformed variables, i.e. $\mathbf{u} = \{E_r,\ aT_e^4,\ aT_i^4\}$.
- Invert the transforms.

Problems of Interest

- BCs: $\lim_{x\to\pm\infty} \mathbf{u}(x,t) = 0$
- ICs: $\mathbf{u}(x,0) = 0$
- Sources:
  - $Q_i = Q_e = 0$.
  - Assume $Q_r = \delta(x)\,\delta(t)$, the solution of this equation is $\mathbf{u}_{\text{planar}}$.
  - Use plane-to-point transform: $\mathbf{u}_{\text{point}}(r,t) = -\frac{1}{2\pi r}\frac{\partial \mathbf{u}_{\text{planar}}(x,t)}{\partial z}\Big|_{z=r}$
  - Due to linearity of equations, superposition of solutions can be used to find solutions of various geometries(e.g. $\mathbf{u}_{\text{sphere}}(r,t) = \int_{V_{\text{sphere}}} \mathbf{u}_{\text{point}}(r,t)\,\mathrm{d}V$)

Inverting the Transforms

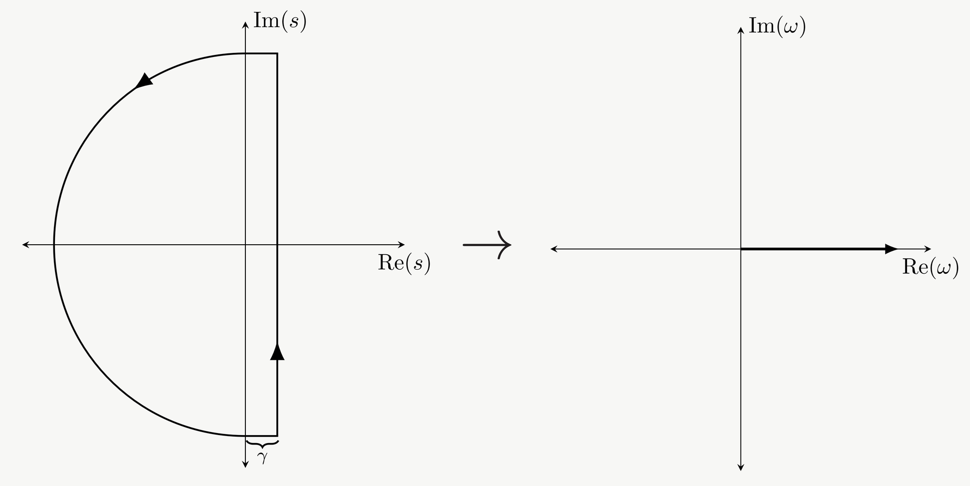
- After non-dimensionalizing, linearizing, taking temporal Laplace and spatial Fourier transforms, and solving the system of linear equations, we have forms for the solution variables in terms of transformed spatial and temporal quantities.

$$\mathbf{u}(x,t) = \frac{-i}{4\pi^2}\oint_{\text{Bromwich}}\int_{-\infty}^{+\infty} \hat{\mathbf{u}}(k,s)\exp(-ikx)\exp(st)\,\mathrm{d}k\mathrm{d}s$$

- It can be shown that for this case that $\mathcal{L}^{-1} = \mathcal{F}^{-1}$ and also the imaginary portion of the integral is odd and thus its contribution must 0.

$$\mathbf{u}(x,t) = \frac{1}{\pi^2}\int_0^{+\infty}\int_0^{+\infty} \hat{\mathbf{U}}_{\text{even}}(k,\omega,x,t)\,\mathrm{d}k\mathrm{d}\omega$$

- Graphically:



- Evaluating this solution does involve computing the double integrals numerically.

Inversion Integrand Properties

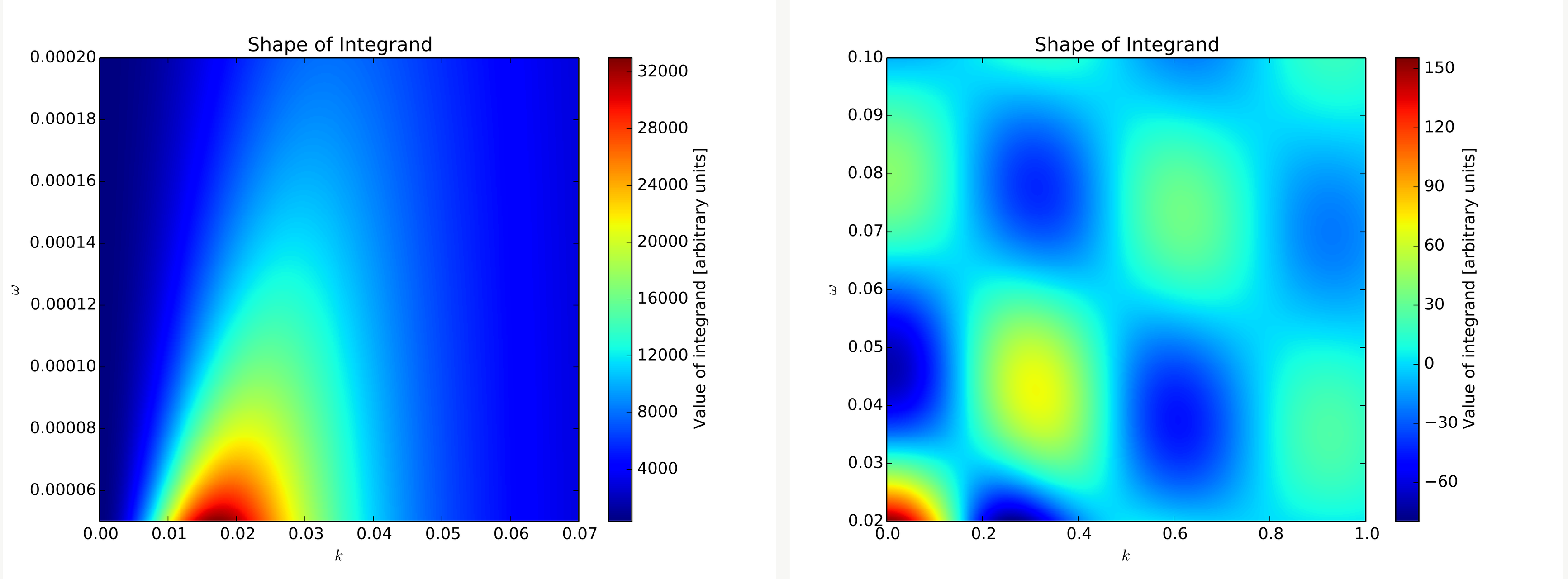


Figure : Not only are the integrands highly oscillatory, but their scale varies drastically.

Computing the Integrals

- Choose initial integration bounds based upon zeros of the integrand.
- Multiply upper limit by some constant (in this case 2) and integrate over new region(s).
- Assume "globally convergent" integrands such that convergence is achieved if successive integral estimates differ by less than *tol*.
- Example:

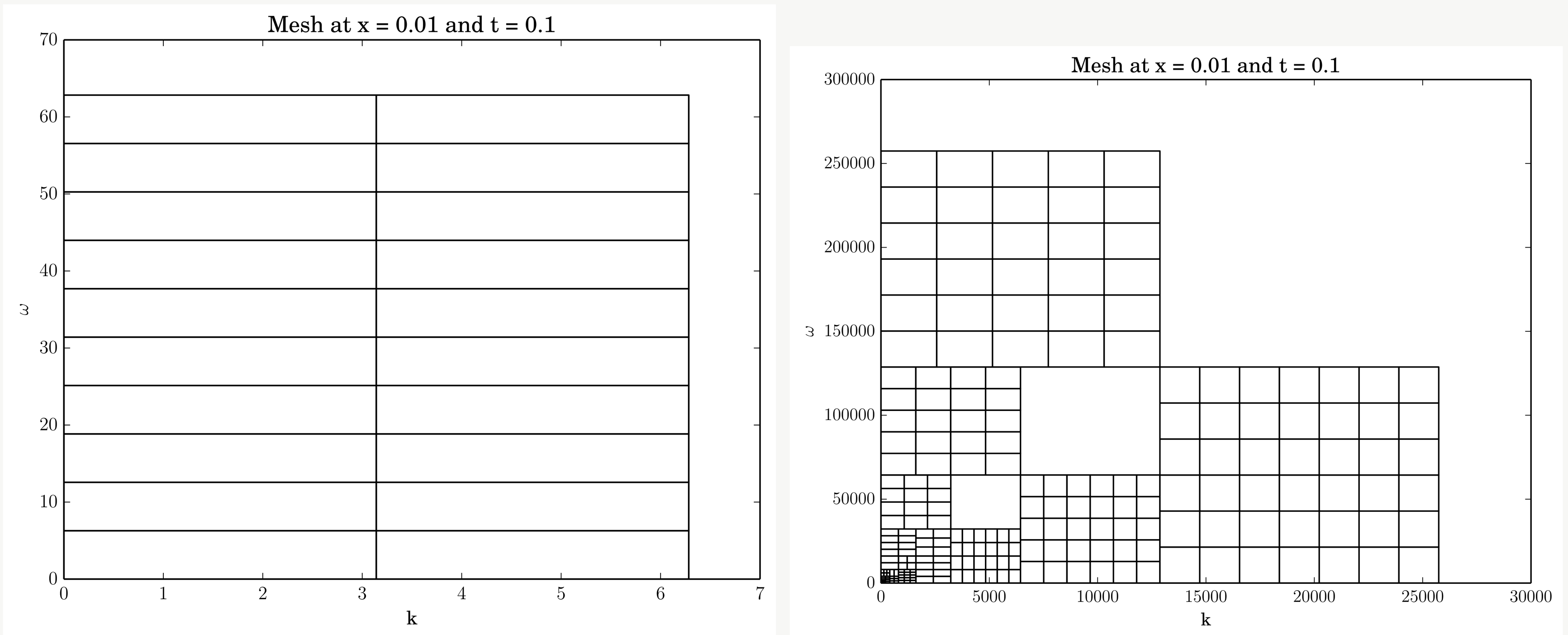


Figure : Initial integration region → all regions after convergence

- For both inner and outer iterations, Wynn-ϵ, Wynn-ρ, iterated Brezinski-θ, and iterated Aitken δ^2 accelerators are used.
- When 2 methods agree to *tol*, that sequence (either inner or outer) is considered converged.

Results

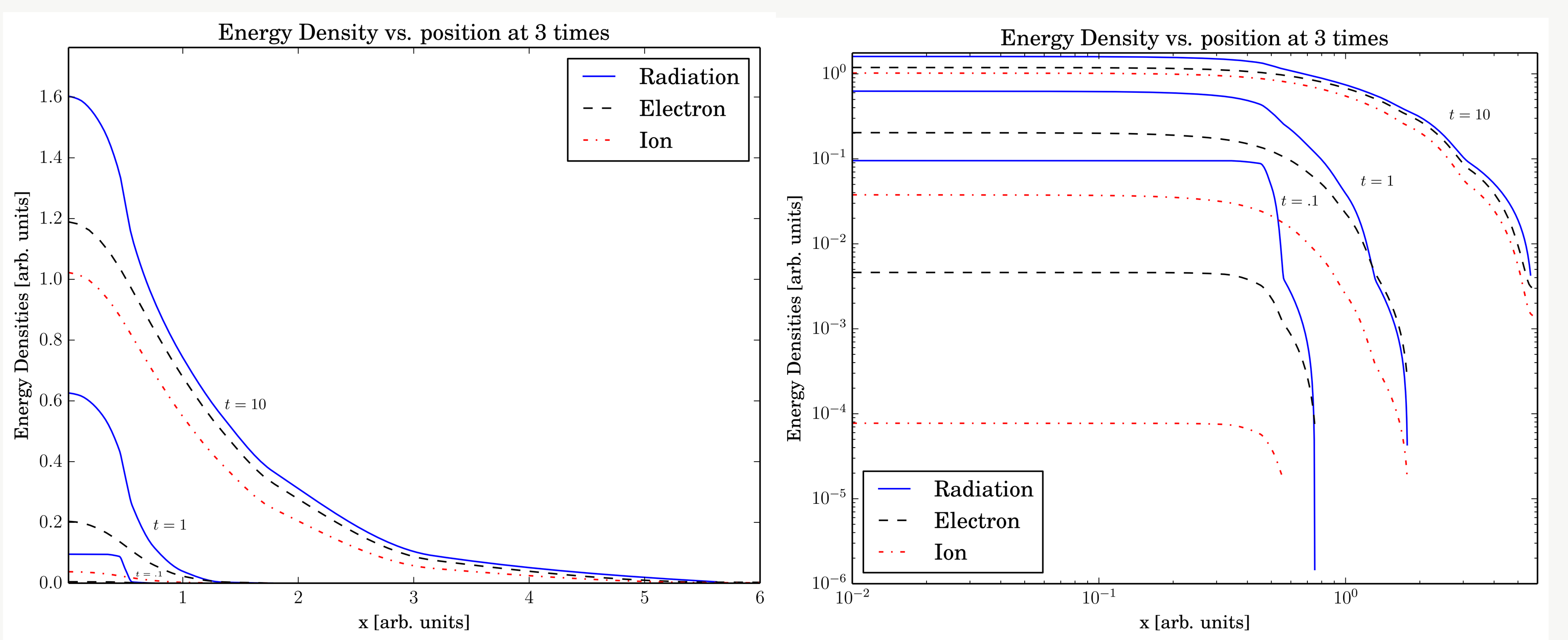


Figure : Solution grows with time when the source is on as is expected.

Heterogenous Computing Results

- To determine usefulness of GPUs for computing integrals, the factor speedup over a single core cpu is a good figure to look at.
- At peak performance, the use of 4 NVIDIA Tesla GPUs was $\sim 561\times$ faster than single core cpu.
- Another way to compare is to look at the number of function evaluations performed per second per Watt.
  - Thermal Design Power wattage was used for both the CPU (80 W) and GPUs (913 W) in question.
  - CPU FOM: 0.0590
  - GPU FOM: 2.903 ($\sim 50\times$ CPU FOM)

TEXAS A&M ENGINEERING EXPERIMENT STATION · SFU SIMON FRASER UNIVERSITY · CENTER FOR LARGE-SCALE SCIENTIFIC SIMULATIONS · University of Colorado Boulder · NUCLEAR ENGINEERING TEXAS A&M UNIVERSITY