Physics–ML Framework for ICF Capsule Heat Transport

Tim Linke
2025 Computational Physics School for Fusion Research
MIT Plasma Science and Fusion Center, Boston, MA

I. Introduction

In an inertial confinement fusion (ICF) experiment, an array of high-powered lasers is focused onto a small fuel capsule, rapidly heating its outer surface. The intense energy deposition drives material ablation, which in turn produces a compressive reaction force that implodes the capsule to extreme densities and temperatures. During this process, the transport of thermal energy through the capsule plays a critical role: it influences the rate of ablation, heating of the fuel, and ultimately the fusion yield. Modeling the interplay between laser heating, material properties, and heat diffusion is a first step for predicting the influence each parameter has on the success of an ICF experiment.

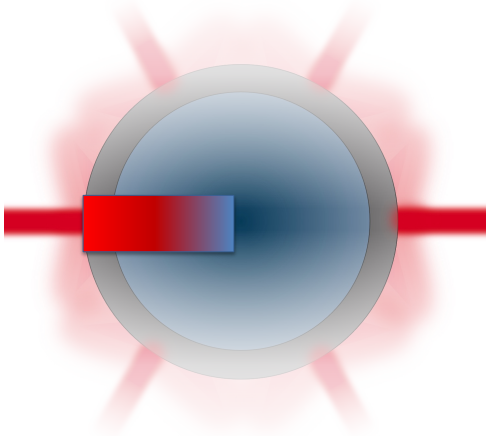

Fig. 1. Laser irradiation interacts with the capsule, ablating the outer shell and generating a heat flux that diffuses throughout the domain.

In this report, I construct a simplified model of the heating process using the one-dimensional heat equation with a pulsed boundary flux representing laser energy deposition (Fig. 1). While far from the full complexity of radiation–hydrodynamics simulations, this model captures the essential process of thermal diffusion into a solid medium under transient, high-intensity loading. I employ parallelization techniques including shared-memory and GPU accelerations to boost computational performance. Recognizing that high-fidelity simulations of the phenomena can be computationally expensive, this report presents a computational framework to enable optimization. I employ a Gaussian process for uncertainty quantification, and train a neural network for fast evaluations of the parameter space.

In essence, the problem serves as a controlled computational testbed for modern parallel programming, surrogate modeling and machine learning techniques. It builds upon the principal skills taught at the Computational Physics School for Fusion Research (CPS-FR), which was held from August 18 to August 23 2025 at the MIT Plasma Science and Fusion Center.

II. A Simple Laser-Driven Heat Model

Modeling laser-driven inertial confinement fusion involves a highly complex interplay of radiation transport, material response, and hydrodynamic motion. In reality, laser light interacts with matter through processes such as inverse bremsstrahlung absorption, scattering, and plasma generation, all of which evolve dynamically in space and time. Capturing these effects in full requires radiation–hydrodynamics simulations with multiple coupled equations. For the purpose of developing a computational testbed, however, several assumptions are adopted to reduce the problem to a simplified model.

First, we assume that the deposited laser energy can be represented as a boundary flux condition at the capsule surface, neglecting volumetric absorption and plasma effects. This captures the essential role of the laser as an external driver without resolving detailed light–matter coupling. Second, we restrict attention to one spatial dimension which allows heat transport to be modeled as a purely diffusive process of radial conduction into the target. Third, we assume material properties such as conductivity, density, and heat capacity remain constant in time and independent of temperature. While in reality they vary under high-energy conditions, this approximation isolates the effect of parameter changes. These assumptions yield a tractable diffusion model suitable for numerical exploration. While these assumptions allow for a compact model, they omit several key aspects of laser–matter interaction, radiative transport, hydrodynamic motion and more, making this testbed highly unsuitable for design studies.

A. Governing Equation

The thermal response of the material is modeled by the one-dimensional heat equation,

$$\frac{\partial T(x,t)}{\partial t} = \alpha \frac{\partial^2 T(x,t)}{\partial x^2}, \qquad 0 < x < L,\ t > 0, \tag{1}$$

where $T(x,t)$ is the temperature, L is the domain length, and $\alpha = \frac{k}{\rho c_p}$ is the thermal diffusivity, with k the thermal conductivity, ρ the density, and c_p the heat capacity.

1) Initial Condition: At the beginning of the simulation, the material is assumed to be at a uniform ambient temperature $T_0 = 400$ K:

$$T(x,0) = T_0, \qquad 0 \le x \le L. \tag{2}$$

2) Boundary Conditions: At the outer shell boundary, a pulsed laser flux is applied for a duration t_p, which is modeled here as a Dirichlet condition:

$$T(0,t) = \begin{cases} T(0,t) + \dfrac{Q}{\Delta x}, & 0 \le t \le t_p, \\ T(0,t), & t > t_p, \end{cases} \tag{3}$$

where Q represents the amplitude of the deposited laser energy per unit time, and Δx is the spatial grid spacing.

At the innermost boundary, a zero-flux boundary represents an insulated boundary where no heat crosses the surface, allowing the temperature at the boundary to change freely in response to the interior temperature distribution:

$$\left.\frac{\partial T}{\partial x}\right|_{x=L} = 0 \tag{4}$$

B. Numerical Discretization

We discretize the heat equation (Eq. 1) using an explicit finite difference method. The spatial domain is divided into n equally spaced grid points with spacing $\Delta x = \frac{L}{n+1}$. Time is discretized into uniform steps of size Δt. The temperature at spatial index i and timestep k is denoted by $T_i^k \approx T(x_i, t_k)$, where $x_i = i\,\Delta x$ and $t_k = k\,\Delta t$. A second-order central difference is used to approximate the spatial derivative,

$$\frac{\partial^2 T}{\partial x^2}(x_i,t_k) \approx \frac{T_{i-1}^k - 2T_i^k + T_{i+1}^k}{(\Delta x)^2}, \tag{5}$$

while a forward Euler scheme is applied in time,

$$\frac{\partial T}{\partial t}(x_i,t_k) \approx \frac{T_i^{k+1} - T_i^k}{\Delta t}. \tag{6}$$

Substituting Eq. 5 and 6 into the governing Eq. 1 yields the explicit update rule

$$T_i^{k+1} = T_i^k + r\left(T_{i-1}^k - 2T_i^k + T_{i+1}^k\right), \qquad i = 1,\ldots,n \tag{7}$$

where the stability parameter is $r = \frac{\alpha\,\Delta t}{(\Delta x)^2}$. For stability of the explicit scheme, it is required that $r \le \frac{1}{2}$.

C. Parallelization

To accelerate the numerical solution, I parallelize the method using OpenMP for both CPU multithreading and GPU acceleration [1] [2]. The discretization of the spatial domain results in a set of algebraic equations that must be solved at each time step. By exploiting the independence of the computations across spatial grid points, I divide the grid points among threads for each time step. Especially on shared-memory CPUs, this allows for simultaneous updates of different portions of the grid. For GPU acceleration, I utilize OpenMP target offloading to transfer the computationally intensive kernel of the spatial domain to the GPU. Memory transfers between host and device were minimized to optimize performance. This approach enables scalable performance across different hardware architectures and allows efficient simulation of heat propagation in large domains. The time-stepping update loop is not parallelized due to its strong dependence on prior time steps, and contributes largely to the serial fraction of the algorithm.

III. Simulation Results

A. Physical Insights

To capture the relationship between laser heating, heat conduction, and energy deposition inside the capsule, I present generated data from the physical model presented in the previous sections. With varying thermal diffusivity α and laser-induced heat flux at the outer boundary, the results describe the temperature evolution throughout the domain by sampling the temperature at the outermost location T_{shell} as well as the innermost temperature T_{fuel}. I consider the four sets of relationships α versus T_{shell}, flux versus T_{shell}, α versus T_{fuel}, and flux versus T_{fuel} to identify systematic patterns. For a range in thermal diffusivity α, values were chosen to represent a wide array of materials, including polymers ($\alpha \approx 10^{-8}$), diamond ($\alpha \approx 10^{-6}$), metals ($\alpha \approx 10^{-4}$), and materials at high temperatures ($\alpha \approx 10^{-3}$). The fluxes are varied between 1 and $10^5 \frac{\text{K m}}{\text{s}}$.

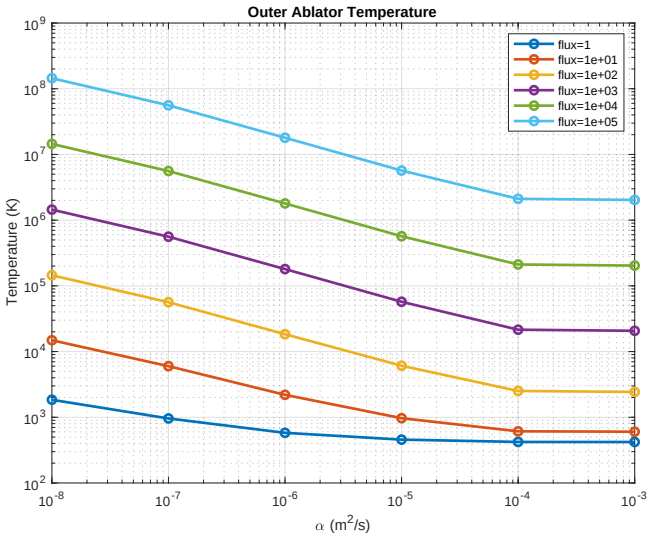

Fig. 2. Evolution of the outer ablator temperature with respect to thermal diffusivity α

When considering the dependence of T_{shell} on α, a clear inverse relationship emerges (Fig. 2). At fixed flux values, large values of α correspond to relatively modest outer shell temperatures, whereas small α leads to extremely high T_{shell}. For instance, in the lowest diffusivity regime ($\alpha \le 10^{-7}\frac{m^2}{s}$), the outer shell reaches values on the order of 10^7-10^8 K. By contrast, at $\alpha \sim 10^{-3}$, T_{shell} remains orders of magnitude smaller under comparable conditions. This behavior is explained by the fact that high diffusivity allows deposited energy to spread quickly into the bulk material, preventing surface overheating, while low diffusivity traps heat near the surface, producing large temperature gradients. The observed monotonic decrease

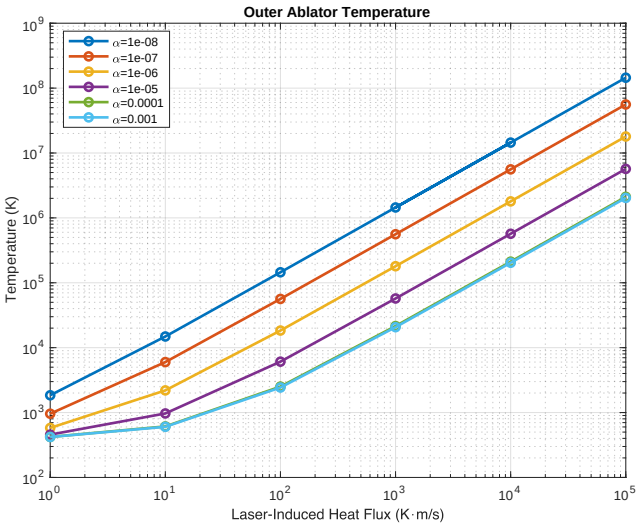

Fig. 3. Evolution of the outer ablator temperature with respect to a laser-induced heat flux

of T_{shell} with increasing α is fully consistent with classical diffusion theory.

The behavior of T_{shell} as a function of incident flux further reinforces this interpretation (Fig. 3). At fixed α, the shell temperature grows nearly linearly with flux, as expected when more power is delivered to the capsule. However, the slope of this growth depends strongly on diffusivity. For large α, the rise is smooth and proportional, with T_{shell} tracking flux across several orders of magnitude. For small α, however, the shell temperature climbs steeply, with heat rapidly accumulating at the surface. For example, at $\alpha = 10^{-8}$, T_{shell} rises from approximately 10^3 K at flux $= 1$ to nearly 10^7 K at flux $= 10^4$, illustrating the inability of the material to redistribute energy inward. Thus, while flux sets the overall scale of heating, α determines whether this heating remains localized or propagates inward effectively.

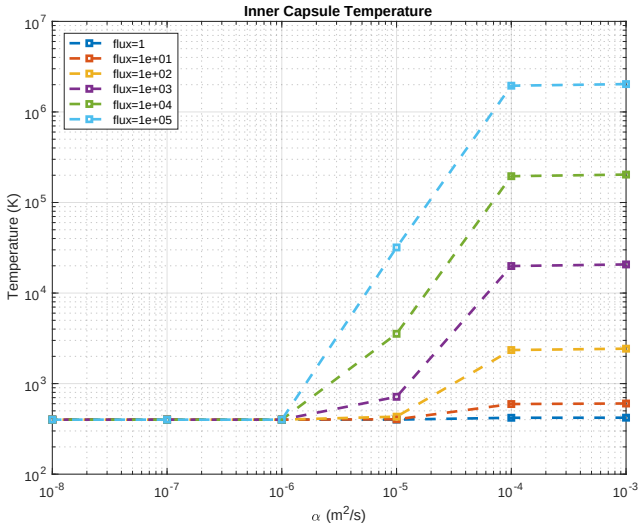

Fig. 4. Evolution of the inner capsule temperature with respect to thermal diffusivity α

Turning attention to the inner fuel temperature T_{fuel}, the results reveal a much more nuanced dependence on α (Fig. 4). At high α, the inner temperature closely follows the outer shell, with $T_{\text{shell}} \approx T_{\text{fuel}}$. This indicates that efficient conduction equilibrates the capsule, allowing the core to share in the energy deposited at the surface. At intermediate α (10^{-4}-10^{-5}), however, a divergence between shell and core temperatures emerges. The core lags significantly behind the shell, reaching only a fraction of the surface temperature. In addition, one observes a nonlinear relationship among different flux values in this regime. At very low α ($\leq 10^{-6}$), the core remains essentially at its baseline of $\sim$ 400 K, regardless of how high the shell climbs. This plateau of T_{fuel} at low diffusivity illustrates a complete bottleneck effect, where conduction is insufficient to carry meaningful energy into the capsule interior.

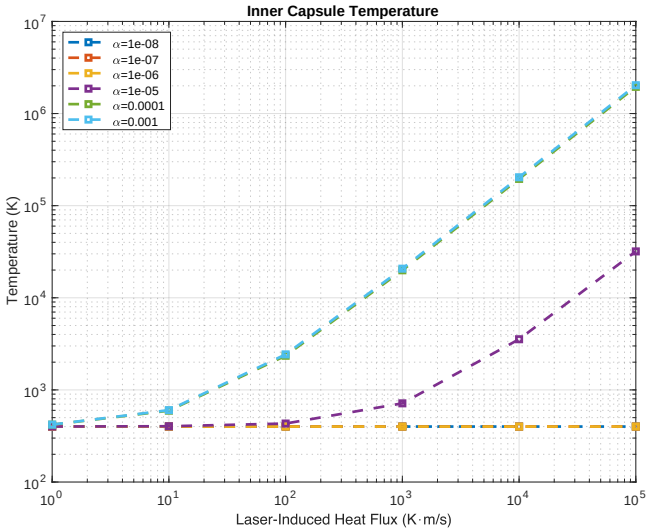

Fig. 5. Evolution of the inner capsule temperature with respect to a laser-induced heat flux

The flux dependence of T_{fuel} further clarifies the transport-limited nature of core heating in Figure 5. At high diffusivity, T_{fuel} increases steadily with flux, closely mirroring the shell response. For example, with $\alpha \sim 10^{-3}$, both T_{shell} and T_{fuel} rise together from the hundreds into the millions of kelvin range as flux increases by four orders of magnitude. At lower α, however, this coupling breaks down: the inner temperature saturates, while the shell continues to climb. Thus, increasing flux without sufficient diffusivity yields diminishing returns for core heating.

The physical origins of these behaviors can be understood by recalling that thermal diffusivity is defined as $\alpha = \frac{k}{\rho c_p}$, where k is the thermal conductivity, ρ is density, and c_p is specific heat capacity. Large α implies rapid spreading of energy and a tendency toward spatial uniformity, while small α produces strong gradients and surface localization. In the extreme low-α limit, the problem effectively reduces to one of boundary heating, with negligible penetration into the bulk. Consequently, the observed results represent two asymptotic regimes: efficient coupling at high diffusivity and decoupled shell-core behavior at low diffusivity.

The role of flux must be considered alongside diffusivity.

Flux represents the input boundary condition as the rate of energy deposition at the shell. For given α, flux sets the scale of possible temperatures. Yet without sufficient α, this energy cannot propagate inward, regardless of how high the flux becomes. In fact, in the low-diffusivity regime, increasing flux only exacerbates the surface-core disparity and drives the shell to extreme temperatures while leaving the core largely unaffected.

From a broader perspective, the results highlight general, important implications for ICF capsule design. First, they state that thermal transport is a limiting factor in energy coupling. Low-diffusivity materials decouple the fuel core from the drive, wasting laser energy in extreme surface heating.

A further implication concerns scaling with laser power. In the favorable high-α regime, both shell and core temperatures scale smoothly with flux. However, the data also demonstrate that in unfavorable transport regimes, no amount of additional flux will compensate for poor conduction.

Altogether, the results point to four distinct heating regimes. In the case of high α and low flux, modest but uniform heating occurs, with both shell and core rising together at a limited scale. With high α and high flux, strong heating of both shell and core takes place, which is the most favorable regime for succcessful ICF experiments. With low α and low flux, heating is minimal overall, and both shell and core remain near baseline. Finally, with low α and high flux, the shell experiences extreme overheating while the core remains nearly cold. While the present analysis focuses on diffusive transport, it is worth noting that real ICF capsules experience a vast amount of additional complexities, including radiative transport, hydrodynamic instabilities, and material ablation. These effects may alter the precise quantitative outcomes.

B. *Computational Scaling*

To test the parallelization strategy, I perform a strong scaling study of our method. Fig. 6 illustrates how the execution time of the fixed-size problem decreases as the number of processing units increases. For this study, a problem size of 1000 cells for 1000 timesteps was considered. In an ideal scenario, doubling the number of processors would halve the runtime, producing a linear scaling curve. However, in practice, parallel performance is often limited by the fraction of code that remains inherently serial, as described by Amdahl's Law [1]. By obtaining the serial fraction of the algorithm, a direct comparison to the ideal scaling is made. Initially, for a small number of threads, the actual speedup closely follows the ideal trend, indicating efficient utilization of parallel resources. As the thread count increases, the curve begins to flatten, revealing diminishing returns due to communication overhead among threads. Ultimately, this overhead becomes so dominant that it degrades the performance significantly. Notably, the plot also shows the performance when using a GPU. A single GPU cannot outperform an increasing number of threads. This is due to the memory transfers associated with hybrid CPU-GPU computations [2]. For the considered problem size, the memory transfer far outweighs any performance gains. These results were obtained on MIT's Engaging cluster, which houses NVIDIA Volta GPUs and AMD EPYC 9734 CPUs.

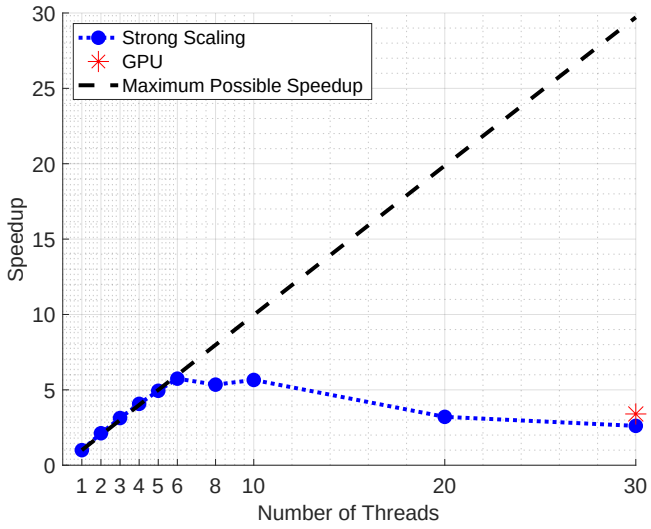

Fig. 6. Laser impeding a capsule. Image generated using AI.

IV. Gaussian Process Regression of the Temperature

To quantify the uncertainty in the data presented in the previous section, I design a Gaussian process (GP). The dataset under consideration consists of measurements of the outer ablator temperature, T_{shell}, as a function of two input variables, α and flux. Initially, the data is transformed using a logarithm to reduce the dynamic range that spans multiple orders of magnitude. Specifically, the inputs α and flux were converted to $\log(\alpha)$ and $\log(\text{flux})$, and the output T_{shell} was transformed to $\log(T_{\text{shell}})$. This log transformation is beneficial for GP modeling, which assumes a roughly Gaussian distribution of the target variable.

To ensure numerical stability and comparable scaling across the input dimensions, the log-transformed inputs are normalized to the unit interval $[0, 1]$ using the standard min-max normalization $X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$. This step is crucial in Gaussian process regression, as the covariance kernel relies on relative distances between points. Unscaled inputs with disparate ranges can lead to poorly conditioned covariance matrices and suboptimal hyperparameter optimization [3].

A two-dimensional Gaussian process regressor is then constructed using a radial basis function (RBF) kernel combined with a white noise kernel. The RBF kernel encodes a smoothness prior, assuming that points closer together in the normalized input space yield similar outputs. The white noise kernel models uncorrelated observational noise, capturing inherent measurement uncertainty. The GP hyperparameters, including the length scales of the RBF kernel and the noise level, are optimized automatically through maximum likelihood.

Predictions are made on a dense grid spanning the range of the original inputs, after log-transformation and normalization. The GP provides both a mean prediction and a standard deviation. The predicted $\log(T_{\text{shell}})$ is transformed back to the original linear scale via exponentiation. The standard deviation is normalized relative to the local mean using $\sigma_{\text{norm}} = \frac{\exp(\mu_{\log} + \sigma_{\log}) - \exp(\mu_{\log})}{|\exp(\mu_{\log})|}$, which provides a dimensionless

measure of relative uncertainty. This normalization ensures that the uncertainty is interpretable across regions of widely varying temperature magnitude.

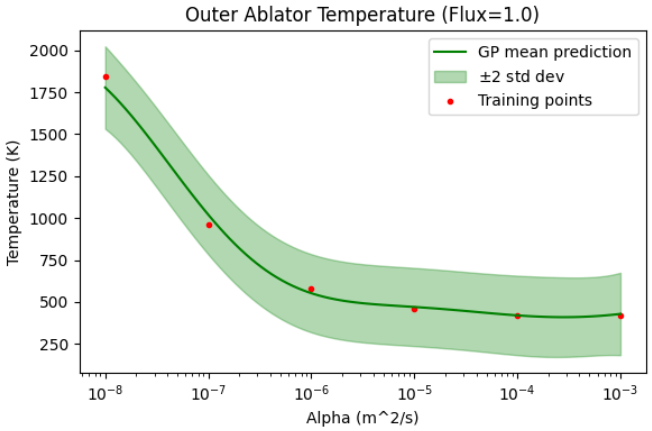

Fig. 7.

The GP predicts a mean and its uncertainty in form of standard deviation across the provided data points. For instance, Fig. 7 shows the one-dimensional temeprature fit of varying α for a constant flux. To span the entire dataset, the input to the GP is expanded to be two-dimensional (α & flux).

The resulting GP is trained using an anisotropic squared–exponential kernel with an additional white noise term. After optimization of the hyperparameters by maximizing the log–marginal likelihood, the resulting kernel is

$$k(\mathbf{x},\mathbf{x}') = \exp\left(-\frac{(x_1-x_1')^2}{2\ell_1^2} - \frac{(x_2-x_2')^2}{2\ell_2^2}\right) + \sigma_n^2\,\delta_{\mathbf{x},\mathbf{x}'}, \tag{8}$$

with characteristic length scales $\ell_1 \approx 0.52$ and $\ell_2 \approx 0.61$, and a noise variance $\sigma_n^2 \approx 3.2\cdot 10^{-5}$. The anisotropic length scales indicate that the model accounts for different sensitivities in the two input directions, while the very small noise variance reflects the near–deterministic character of the data.

The moderate values of the optimized length scales suggest that the Gaussian process captures local variations in the response surface without overfitting to spurious fluctuations. In particular, variations along the first dimension are resolved at a slightly finer scale compared to the second, indicating that the model attributes somewhat greater importance to changes in the first input variable. The negligible noise variance emphasizes that the uncertainty in the posterior predictions is driven primarily by the distance to training points rather than by observational noise, which is consistent with the deterministic nature of the underlying simulations.

The resulting contour plot of the mean temperature predicted by the GP is shown in Figure 8. It reveals a smooth dependence on both α and flux. Training points are concentrated in specific regions, and the GP naturally interpolates across sparsely sampled areas. The logarithmic color scaling highlights orders-of-magnitude variations in T_{shell}, demonstrating that higher flux generally correlates with higher temperatures, while the dependence on α is less steep but still significant.

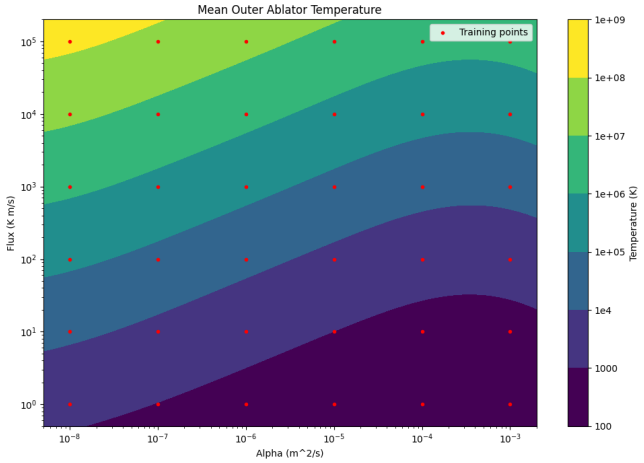

Fig. 8. Mean Temperature predicted by the Gaussian process for a constant flux

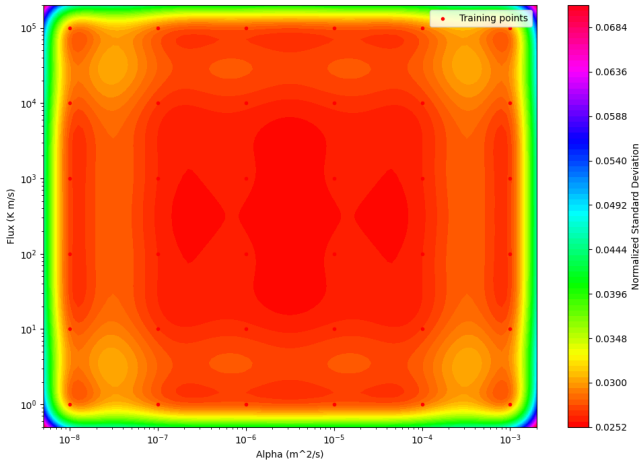

Fig. 9. Standard deviation of the temperature prediction by the Gaussian process.

More importantly, Figure 9 shows the uncertainty associated with each region in the dataset. It shows regions of high confidence in the center of the domain. Regions far from training points, such as the boundaries, exhibit dramatically higher normalized standard deviations, reflecting the GP's conservative extrapolation behavior. Regions in the corners of the domain, where neighboring points are fewer, also exhibit higher uncertainties.

In summary, the Gaussian process regression captures both the central tendency and uncertainty of T_{shell} across the input space. The log transformation, input normalization, and relative standard deviation scaling are key steps that enhance numerical stability, interpretability, and visualization of predictive uncertainty. These results provide both quantitative predictions and qualitative insights into the dependence of the ablator temperature on material and flux parameters.

V. Neural Network Modeling of Heat Equation Behavior

A. Background and Architecture

To approximate the mapping between the input parameters of the physical system (the thermal diffusivity α and flux) and the output temperature response T, a feedforward neural network is developed using JAX. The neural network serves as a nonlinear regression model capable of capturing the wide dynamic range present in the data, where both inputs and outputs span several orders of magnitude. To improve numerical stability and facilitate training, the inputs and outputs are transformed into $\log_{10}$-space, and the input features are standardized to zero mean and unit variance based on the training set statistics.

The network architecture consists of one hidden layer with five hidden units, chosen to balance the model's performance with the small dataset size. Figure 10 shows a visual representation of the network. A sigmoid activation function is used in the hidden layer to introduce nonlinearity. The implementation also supports ReLU activations, but it was observed that sigmoid greatly outperformed ReLU, an expected result for such a small dataset. The output layer is linear, appropriate for regression tasks in transformed space. Weight parameters are initialized with small random values from a normal distribution, and L2 regularization is included in the loss function to mitigate overfitting [4].

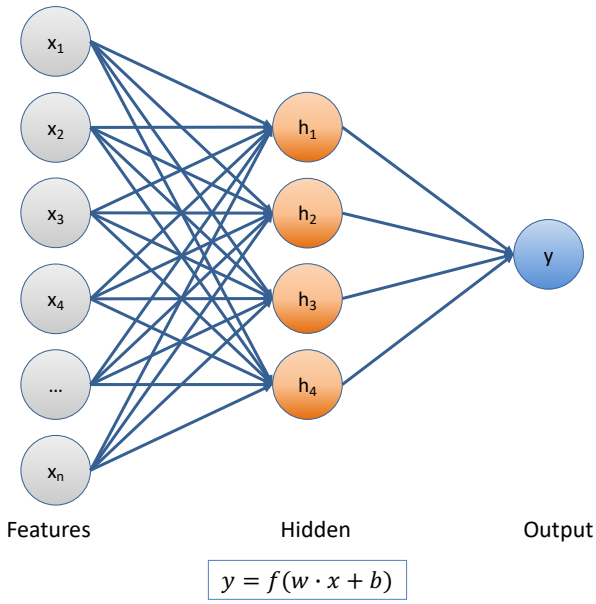

Fig. 10. A fully connected neural network is used to learn the physical behavior of the system.

The optimization is performed with the Adam optimizer, which provides adaptive learning rates for each parameter and improves convergence stability compared to gradient descent. A learning rate of 0.01 was used, and the network was trained for 5000 epochs.

B. Connection to the Physical Problem

The underlying physical problem is governed by the heat equation, where the temperature response depends on both material properties (represented by α) and the applied flux. Analytically, these dependencies are complex, involving nonlinear interactions and strong scaling behaviors across orders of magnitude. The neural network architecture is chosen to approximate this nonlinear functional relationship without requiring explicit assumptions about the form of the governing equations.

The use of a logarithmic transformation, similar to the GP regression highlighted in the previous section, is particularly well aligned with the physics of the problem. Since both inputs and outputs vary exponentially with respect to system parameters, mapping them into log-space stabilizes training. The training dataset consisted of 36 samples spanning a wide range of α and flux values. A split of 80% training and 20% validation data was used, ensuring that validation loss provided an estimate of generalization performance.

C. Results and Loss Interpretation

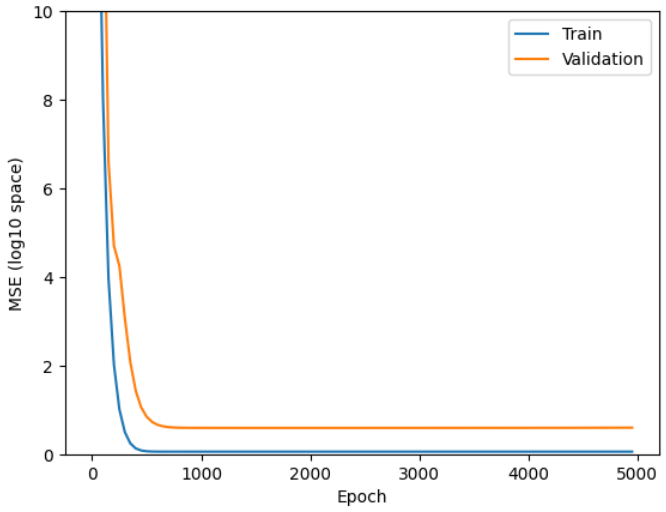

Fig. 11. Training and validation loss of the neural network.

To judge how well the model learned the underlying phenomena of the physical problem, Figure 11 shows the training and validation loss of the neural network. The model achieved a final training mean squared error (MSE) of 0.055 and a validation MSE of 0.59, both computed in log-space. Since the predictions and targets were represented in $\log_{10}$-space, the MSE corresponds to squared errors in orders of magnitude. For example, an MSE of 0.055 indicates that, on average, the model predictions differ from the true target by roughly $\sqrt{0.055} \approx 0.23$ log-units, or about 70% relative error in the original scale.

The substantially higher validation loss compared to training loss suggests that while the model has learned the dominant nonlinear trends in the training data, there is still a gap in its ability to generalize across the entire parameter domain. This discrepancy is expected given the limited dataset size and the wide dynamic range of the outputs. A variety of parameters

were tested to assess their importance, including number of hidden layers and number of hidden nodes. To truly improve the performance, more data points are required.

D. Model Evaluation on Test Data

After training the neural network on the given dataset, the learned parameters $(w = W_1, W_2)$ can be directly applied to new, unseen input data. The test inputs, denoted as (X_{test}), are passed through the trained network to obtain predictions $\hat{Y}_{\text{test}} = f_w(X_{\text{test}})$. This procedure does not involve any gradient updates, as the model weights are frozen after training. The relative error between the predictions $(\hat{Y}_{\text{test}})$ and the ground truth labels (Y_{test}) serves as the primary evaluation metric: $e_{\text{rel},i} = \frac{\hat{y}_i - y_i}{y_i}$.

Consulting the uncertainty regions obtained in Fig. 9, test data points are deliberately chosen to cover both regions of high confidence as well as low confidence. For the high confidence region, inputs near the center of Fig. 9 are elected. For low confidence regions, the four gold-shaded corners within the red domain are chosen.

E. Discussion of Training Loss

The inputs are provided to both the neural network as well as the discretized heat equation. Table I presents the chosen input parameters for thermal diffusivity α and heat flux. The relative error between the neural network predictions and the reference partial differential equation solutions is also given. The results illustrate how the accuracy of the NN depends

TABLE I. RELATIVE ERROR BETWEEN PDE SOLUTION AND NN PREDICTION FOR DIFFERENT INPUT PARAMETERS.

α	Flux	Relative Error
$5.0 \cdot 10^{-6}$	$5.0 \cdot 10^{3}$	0.3313
$5.0 \cdot 10^{-6}$	$5.0 \cdot 10^{2}$	0.7829
$5.0 \cdot 10^{-6}$	$5.0 \cdot 10^{1}$	0.7986
$5.0 \cdot 10^{-7}$	$5.0 \cdot 10^{2}$	0.2042
$5.0 \cdot 10^{-5}$	$5.0 \cdot 10^{2}$	1.3559
$5.0 \cdot 10^{-8}$	$5.0 \cdot 10^{0}$	0.1643
$5.0 \cdot 10^{-8}$	$5.0 \cdot 10^{4}$	0.9101
$5.0 \cdot 10^{-4}$	$5.0 \cdot 10^{0}$	0.2363
$5.0 \cdot 10^{-4}$	$5.0 \cdot 10^{4}$	0.0780

on the operating regime of the system. For moderate values of α and flux (e.g., $\alpha = 5.0 \cdot 10^{-6}$ with flux $= 5.0 \cdot 10^{3}$), the relative error is reasonably small (0.3313), suggesting that the NN captures the dominant physical behavior. However, for other cases such as $\alpha = 5.0 \cdot 10^{-5}$ with flux $= 5.0 \cdot 10^{2}$, the error rises above unity (1.3559), indicating a substantial deviation from the PDE solution. Interestingly, the lowest error (0.0780) occurs at the extreme condition of $\alpha = 5.0 \cdot 10^{-4}$ and flux $= 5.0 \cdot 10^{4}$. Conversely, large discrepancies at lower flux levels highlight potential limitations in training coverage or sensitivity of the underlying physics. Interestingly, no general connection between accuracy and uncertainty is observed. The first five data points in Table I were selected from a high-confidence region, while the last four points were sampled from a low-confidence region.

VI. CONCLUSION

In this work, I developed a comprehensive computational framework for modeling and analyzing laser-driven thermal transport processes. The approach combined several key components, including the design and implementation of a multi-threaded and GPU-accelerated solver for partial differential equations describing the underlying physics, the integration of uncertainty quantification via Gaussian process regression, and the training of a neural network on simulation data to serve as a fast surrogate model. Together, these components enable both high-fidelity simulations and efficient predictive capabilities.

The developed framework provides a foundation for future applications in materials design and process optimization. Specifically, the combination of physics-based modeling, uncertainty quantification, and machine learning opens the possibility of systematically exploring material responses under varying laser heat flux conditions. By coupling the surrogate neural network with optimization routines, one can identify material properties or operating parameters that maximize desired performance metrics, such as thermal stability or energy efficiency, under prescribed laser conditions. This paves the way toward data-driven material optimization and adaptive design in laser-based manufacturing and related technologies.

All data and source code used in this work are openly available on Github.

VII. ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Cristina Rea and her team for their outstanding organization and execution of the Computational Physics School for Fusion Research (CPS-FR). Their efforts provided an exceptional environment that allowed me to explore novel directions in computational physics with confidence and rigor. I am also grateful to Timothy Mattson, Tamara Broderick, and Michelle Kuchera for their insightful workshops, the content of which was invaluable in supporting this work. Finally, I would like to thank the MIT Plasma Science and Fusion Center for the invitation to participate in the CPS-FR, and MIT ORCD for providing access to computational resources used in this work.

REFERENCES

[1] Timothy Mattson. An introduction to parallel programming with openmp. Workshop at the Computational Physics School for Fusion Research, August 2025. URL: https://github.com/tgmattso/ParProgForPhys/blob/main/OpenMP_CPU.pdf.

[2] Timothy Mattson. Programming your gpu with openmp. Workshop at the Computational Physics School for Fusion Research, August 2025. URL: https://github.com/tgmattso/ParProgForPhys/blob/main/OpenMP_GPU.pdf.

[3] Tamara Broderick. Gaussian processes for regression: Models, algorithms, and applications. Workshop at the Computational Physics School for Fusion Research, August 2025. URL: https://tamarabroderick.com/tutorial_2025_cps-fr.html.

[4] Michelle Kuchera. Machine and deep learning. Workshop at the Computational Physics School for Fusion Research, August 2025. URL: https://github.com/alpha-davidson/CPS-Fusion25-ML/tree/main/lectures.