# Introduction to the Tidy3D Heat Solver

Tom Chen, Senior Photonic Engineer at Flexcompute

**Abstract**

Many optical and photonic systems are neither isolated nor passive. In numerous cases, it is essential to understand the thermal response of these systems and the consequent changes in optical performance due to thermal effects. A common example is an integrated photonic waveguide with a metallic heater on top. The heater induces localized heating, causing a change in the refractive index of the surrounding materials. This, in turn, alters the waveguide mode and modifies the propagation characteristics of the mode, forming the basis for thermo-optic modulators. In modeling such scenarios, Tidy3D's heat solver serves as an effective tool. This article gives a general introduction to the heat solver and the various features it supports.

## 1 Heat transfer mechanisms

To understand what the heat solver is solving, we need to first understand the mechanisms of heat transfer. There are mainly three heat transfer pathways: conduction, convection, and radiation.

## Conduction

Conduction is the heat transfer through the material due to a temperature gradient. More generally, the heat conduction equation is given by

$$\rho c_p \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) = Q,$$

where $\rho$ is the density of the material, $c_p$ is the specific heat capacity of the material, $T$ is temperature, $t$ is time, $k$ is the thermal conductivity of the material, and $Q$ is the heat source, representing heat generation per unit volume. As of the current version of Tidy3D, the heat solver only solves the steady-state temperature distribution. That is, $\frac{\partial T}{\partial t} = 0$ and the equation simplifies to

$$-\nabla \cdot (k \nabla T) = Q.$$

This is the equation the heat solver solves for within the simulation domain and the thermal conductivity is the material property that we need to specify for all included materials.

## Convection

Heat convection occurs through the net displacement of a fluid that transports heat content along with its velocity. It refers to the dissipation of heat from a solid surface to a fluid, typically characterized by a heat transfer coefficient $h$. The convective heat transfer is described by

$$\frac{dQ}{dt} = h(T - T_\infty),$$

where $\frac{dQ}{dt}$ is the rate of heat transferred and $T_\infty$ is the ambient temperature. The heat solver focuses on modeling the heat transfer between solids. Convection is only supported as a boundary condition.

### Radiation

Radiation is the transfer of heat in the form of electromagnetic waves, typically considered for interactions between surfaces or between a surface and its surroundings. Like convection, radiation is usually modeled separately and applied as a boundary condition, especially when dealing with surfaces exposed to the environment. As of the current version of Tidy3D, the radiation boundary condition is not supported yet.

## 2 Material Properties

When conducting a heat simulation, it is essential to accurately specify the thermal properties of the materials, much like defining the refractive index in optical simulations. Two types of mediums are supported: solid and fluid. When setting the medium as fluid, the heat solver equation will **not** be calculated in the regions with that medium. When setting the medium as solid, the thermal conductivity needs to be specified. If the objective is to simulate heat transfer and observe how temperature variations affect the optical properties of the medium, both the optical medium type and the optical perturbation parameters, including the thermo-optic coefficient, must be defined. This coefficient quantifies the change in the refractive index with temperature.

# 3 Heat Sources

Currently, only a uniform volumetric heat source is supported. Users can specify the volumetric rate of heating. The source can be applied to any structure within the simulation domain.

In practice, one often wants to model a heater with external current applied. To model this Joule heat source, we can calculate the volumetric Joule heat generation using

$$\frac{dP}{dV} = \frac{1}{\sigma}\left(\frac{I}{w_{\text{heater}}h_{\text{heater}}}\right)^2,$$

where $\sigma$ is the electrical conductivity of the heater material, $I$ is the applied current, $w_{\text{heater}}$ and $h_{\text{heater}}$ are the width and thickness of the heater, respectively.

# 4 Boundary Conditions

A crucial step in setting up a heat simulation is to specify appropriate boundary conditions to ensure accurate physical representation. These boundary conditions can be applied to the surface of a structure, the interface between different structures, the interface between different media, the boundary of the simulation domain, or the interface between a structure and the simulation domain. Currently, three types of boundary conditions are supported.

## Temperature Boundary Condition

This boundary condition specifies the temperature on the boundary of the domain. It directly sets the temperature $T = T_0$ on the boundary surface.

### Heat Flux Boundary Condition

This boundary condition specifies the heat flux across the boundary. It sets the derivative of the temperature (which corresponds to the heat flux) normal to the boundary:

$$k\frac{\partial T}{\partial n} = Q.$$

### Convection Boundary Condition

This boundary condition represents convective heat transfer between the domain and the surrounding environment:

$$-k\frac{\partial T}{\partial n} = h(T - T_\infty).$$

When using this boundary condition, users can specify $h$ and $T_\infty$.

## 5 Meshing

Good meshing ensures the accuracy of the simulation result. The heat solver uses unstructured mesh, unlike the rectangular grid in the FDTD solver. Currently, two mesh types are supported.

### Uniform Unstructured Mesh

When using the uniform unstructured mesh type, users only need to specify the grid size $dl$ and the mesher will automatically generate a mesh to accommodate the structures and the specified grid size. Note that the uniform mesh still considers the geometric details of the structures. It takes into account the parameters *min_edges_per_circumference*(the minimum number of mesh segments per circumference of an object) and *min_edges_per_side*(the minimum number of mesh segments per any side of an object) and assigns a smaller mesh size locally if needed. The minimal allowed mesh size is controlled by *relative_min_dl*, which is defined as the minimal allowed mesh size relative to the largest dimension of the simulation domain.

### Distance Unstructured Mesh

Distance unstructured mesh type uses an adaptive grid based on the distance to material interfaces. The mesh size is controlled by *dl_interface*, which is the mesh size near material interfaces, and *dl_bulk*, which is the mesh size away from material interfaces. The minimal allowed mesh size is also controlled by *relative_min_dl*. This is the recommended meshing type.

## 6 Examples and Tutorials

See the following tutorials and examples on how to run heat simulations as well as coupled heat and optic simulations:

Heat solver Python tutorial

Thermally tuned waveguide in Python

Thermally tuned waveguide in GUI

Thermally tuned ring resonator in Python

Thermally tuned ring resonator in GUI

Thermo-optic modulator with a doped silicon heater in Python