Simulation of Heat Sink Cooling System for Modern CPUs

Camacho Barranco, Roberto; Gavranovic, Stefan; Valizadeh, Mahyar

Introduction

This project is focused in integrating the knowledge acquired during the semester and in previous Fluid Dynamics courses to simulate the heat flow across a Heat Sink, similar to the ones typically used in CPU's. This is a very important application, especially because of the role that heat dissipation has in the energy consumption of a system, which is of particular interest in the case of High Performance Computing. The amount of energy for cooling the CPU will be much higher if the heat is not transferred optimally to the environment. The program delivered is capable of creating different geometries for the heat sink in a two-dimensional space, changing the amount and size of the fins. This is done in order to be able to analyze the amount of heat dissipation that each unique geometry provides, and allows for optimization of the parameters of the heat sink.

Input parameters

In order to create the executable file, the whole project must be unzipped into a folder and then the `make` command should be called from the terminal. After making the project, the following command should be used:

```
./sim dataFile geometry
```

The software takes as an input a `.dat` file which name should be specified as `dataFile` without the file extension. The input `geometry` as second argument should be either `vertical` or `horizontal` depending in which type of geometry will be simulated. The parameters that should be included in the `.dat` file are briefly described next:

- **size of the domain:** `xlength, ylength`
- **number of cells:** `imax, jmax`
- **time steps:** `dt, t_end, tau`
- **output:** `dt_value`
- **pressure solver (SOR):** `itermax, eps, omg, gamma, alpha`
- **Reynolds number:** `Re`
- **Prandtl number:** `Pr`
- **coefficients:** `beta` (thermal expansion), `kratio` (ratio of conductivity of fluid to solid), `Kappa` (thermal diffusivity of solid). These values are calculated using properties of air and aluminum.
- **gravitation:** `GX, GY`
- **pressure initialization:** `PI`
- **temperature initialization:** `TI` (fluid), `TSI` (solid)

- **velocity initialization:** `UI, VI`
- **fluid domain boundaries:** `wl, wr, wt, wb`
- **type of boundary (Neumann or Dirichlet):** `wlt, wrt, wbt, wtt`
- **temperature at boundaries:** `TL, TR, TB, TT`
- **heat flow at boundaries:** `QL, QR, QB, QT`
- **pressure change:** `lp, rp, dp`
- **fins for heat sink:** `fins, finWidth, finHeight, finDistance, finCore`

Geometry description

As mentioned earlier, there are two different geometries that are handled by the software automatically, and are specified by several parameters in the `.dat` file. The following is a graphical depiction of how the parameters are defined. For the "vertical" case, the width and height parameters are switched from the intuitive approach, and the number of fins and width of the core should be multiplied by two. This was implemented in this way for it to be possible to easily switch between cases and have a geometry close to reality in both cases.

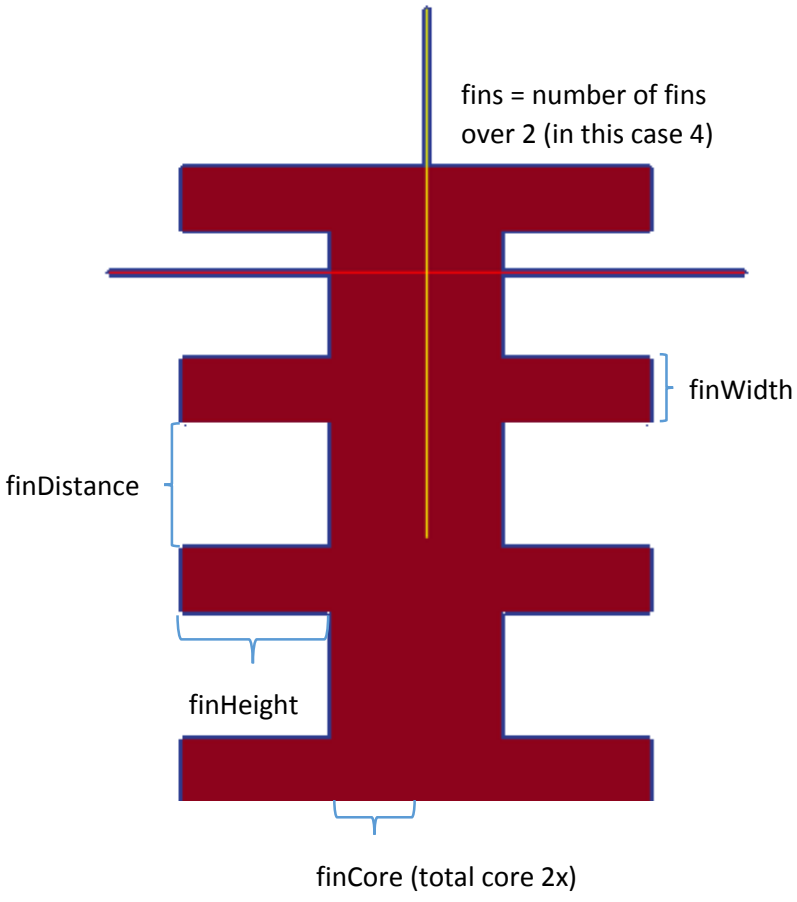

Figure 1: Vertical Core

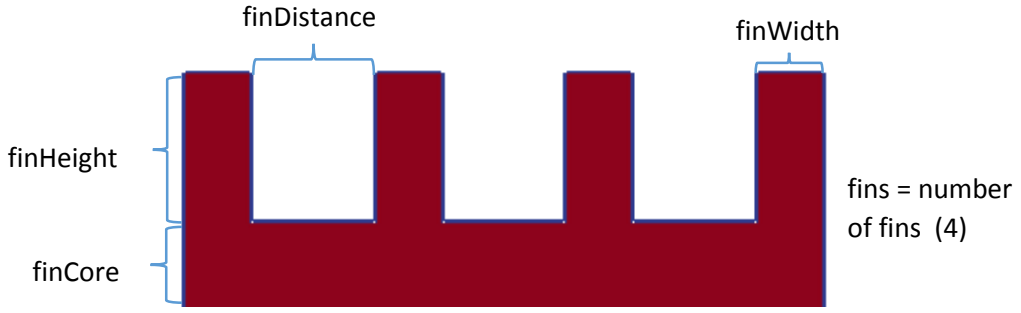

Figure 2: Horizontal Core

Theoretical Background

For conjugate heat transfer analyses, the solid material conduction and the fluid convection are analyzed simultaneously. The type of fluid convection (natural, forced or mixed) determines the analysis parameters. To simulate forced convection, one must obtain a converged flow solution, disable the flow and run additional uncoupled thermal-only heat transfer iterations. But in this case natural convection is simulated, which requires energy equation to be coupled with flow simulation. [1]

Natural and free convection flows are largely dominated by buoyancy forces. The buoyancy forces are generated by density gradients which vary primarily with temperature since pressure gradients are relatively small in these flows. Natural convection flows may be laminar or turbulent but, again, for simplification just the laminar case is considered here, which requires a limitation on values of Reynolds and Prandtl numbers.

One way of modeling the above phenomena is to use a monolithic (i.e. tightly coupled) discretization for both solid and fluid subdomains with the interface boundary conditions. However, one generally solves different equations in the solid and fluid subdomains. Typically, the unsteady thermal diffusion equation is computed in the solid subdomain

$$\frac{\partial T}{\partial t} = \kappa \Delta T \qquad (1),$$

and the Navier–Stokes equations coupled with energy equation using Boussinesq approximation (eq. 4) are solved in the fluids subdomain. [2]

$$\frac{\partial}{\partial t}\vec{u} + (\vec{u} \cdot \operatorname{grad})\vec{u} + \operatorname{grad} p = \frac{1}{Re}\Delta\vec{u} + (1 - \beta T)\vec{g} \qquad (2)$$

$$\frac{\partial T}{\partial t} + \vec{u} \cdot \operatorname{grad} T = \frac{1}{Re}\frac{1}{Pr}\,\Delta T + q^m \qquad (3)$$

$$\varrho(T) = \varrho_\infty\big(1 - \beta(T(\vec{x}, t) - T_\infty)\big), \qquad \text{with } \beta := -\frac{1}{\varrho}\frac{\partial \varrho}{\partial T} \qquad (4)$$

A natural way of avoiding such difficulties is to solve the coupled thermal equations in a partitioned manner. In a partitioned solution procedure, the fluid and solid subdomains are solved sequentially on decomposed subdomains. A key component of a partitioned procedure is the formulation and

implementation of the interface conditions characterizing the fluid–structure coupling. The fluid and structure equations are alternately integrated in time by separate solvers with the aid of Dirichlet and Neumann boundary conditions along the interface. This technique for solving coupled mechanical system was introduced by Felippa and Park, which is often referred to the conventional sequential staggered (CSS) procedure. [2]

At the interface, the Dirichlet and Neumann continuity conditions are given by

$$\mathrm{T}_{+} = \mathrm{T}_{-} \text{ and } \mathrm{q}_{+} = \mathrm{q}_{-}$$

for the fluid subdomain (+) and the solid subdomain (-). We discretize both subdomains with the finite-difference technique. Using these boundary conditions one can obtain the values of temperature on ghost cell for both of subdomains.

$$T|_{\Gamma_+} = T_w \text{ and } -\kappa_+ \left.\frac{\partial T}{\partial n}\right|_{\Gamma_+} = q_w$$

$$T|_{\Gamma_-} = T_w \text{ and } -\kappa_- \left.\frac{\partial T}{\partial n}\right|_{\Gamma_-} = q_w$$

The important factor here is the ratio of heat conductivity of fluid to solid structure.

$$\kappa_{ratio} = \frac{\kappa_+}{\kappa_-}$$

The other temperature boundary conditions of the simulated problem is Neumann condition for top and bottom boundary with heat source underneath the structure and Dirichlet conditions on inflow parts on sides.

References

[1] Autodesk, "Basic Heat Transfer," 2013. [Online]. Available: http://wikihelp.autodesk.com/Simulation_CFD/enu/2013/Help/0407-Learning407/0726-Guidelin726/0737-Basic_He737. [Accessed 09 07 2013].

[2] M. Griebel, T. Dornsheifer and T. Neunhoeffer, Numerical Simulation in Fluid Dynamics: A practical introduction, Munich: SIAM, 1997.

[3] P. K. Felippa CA, "Staggered transient analysis procedures for coupled mechanical systems: formulation," *Computer Methods in Applied Mechanics and Engineering,* no. 24, p. 61–111, 1980.