README

December 6, 2023

1 Exercise 5 - Hybrid CPU computing

1.1 Introduction to the physical case

Now you know enough to run, autopsy, and evaluate a real application, which also does meaningful computations, not only communications. Let us investigate one of the most simple, but very important types of partial differential equations in physics and nature – the diffusion equation, in our case applied to describe diffusion of heat. For overall details, you can even start from Wikipedia, and virtually every textbook in physics covers this type of equation to varying detail.

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u$$

where **u(x, y, t)** is the temperature field that varies in space and time, and is thermal diffusivity constant. The two dimensional Laplacian can be discretized using finite differences that form a second order von Neuman stencil:

$$(\nabla^2 u)(i,j) = \frac{u(i-1,j) - 2u(i,j) + u(i+1,j)}{(\Delta x)^2} + \frac{u(i,j-1) - 2u(i,j) + u(i,j+1)}{(\Delta y)^2},$$

Given an initial condition (u(t=0) = u0) one can follow the time dependence of the temperature field with explicit time evolution method:

$$u^{m+1}(i,j) = u^m(i,j) + \Delta t \alpha (\nabla^2 u)^m(i,j),$$

where Δt is the length of the time integration step. For a unique solution, boundary conditions are needed, which can be specified as conditions for temperature, its normal derivative or a combination of both.

We will start with an MPI implementation of a two-dimensional (2D) version of the heat equation. The two dimensional grid is decomposed along both dimensions, and the communication of boundary data is overlapped with computation. Restart files are written and read with MPI I/O. See code usage markdown instructions for more details on how to run the code.

2 Your tasks

Add loop-level parallelism using openMP to this code, and assess whether any of the expected benefits, listed below, discussed in the Lecture 5 materials, can be reached with the methods that you are knowledgeable with? The openMP methods you learnt during Programming Parallel Computers course are sufficient. Provide evidence in the form of tables, plots, … , and present a short analysis on each point.

1. reduction in memory usage?
2. performance increase?
3. extended scale up?

Both your code implementation and the results you collect in a short pdf description of the exercise project will be evaluated (please name it as report.pdf).