2D Steady-State Heat Equation using an Iterative Method with MPI

Objective

Write an MPI program to simulate the 2-dimensional (2D) steady-state "heat" equation using an iterative method. The purpose is to give you experience working with MPI.

Problem

The 2D steady-state head equation problem is described here is a classic **example** for parallel implementation.

Assignment

Write an MPI program in C or C++ to implement the 2D heat equation using an iterative algorithm. Demonstrate the program on a 1024x2048 domain for 1, 2, 4, 8, 16, and 32 processes, running for 100 timesteps. Let all temperatures be floating point values in the range 0.0 to 1.0. Let the boundry values be 1.0. (You can change these values to get different outcomes.) Assume the rest of the plate starts with an initial temperature of 0.0. Print out the maximum element update difference (in %) for every subregion at every 5 timesteps.

A good way to test your code is to first write a sequential solution and make sure it is working correctly. Then this version will be your "gold standard" for comparing the parallel results at any timestep.

Feel free to play around with domain size.

Note: MPI allows you to configure all of your ranks to run on a single node. This might be helpful when you are debugging, since it is easier to get a single node in the scheduling queue. Once it is working, running on multiple nodes likely will achieve better performance.