Optimisation of d2q9-bgk Lattice Boltzmann Scheme with MPI

James Elgar, za18968

May 5, 2021

1 Introduction

This report will explore the use of distributed memory parallelism to optimize a given algorithm. The algorithm solves a d2q9-bgk Lattice Boltzmann scheme over a grid of cells. The report will evaluate different approaches to distributed memory parallelism, mostly using the Message Parsing Interface, MPI, standard. When optimizing a distributed memory problem there are various factor which can effect the performance of the algorithm. This report will focus of the optimization of individual nodes, the data layout of the cells array, the use of blocking vs non-blocking sends and final compare the use of MPI for running with multiple processor with OpenCL for running with a graphics card.

2 MPI

In this MPI implementation 4 nodes were used. In order to distribute the work across these 4 different nodes, the grid is divided up into 4 sections. When dividing a grid into different sections there are generally 2 approaches, in square sub grids or in rows (or columns). Figure 1 is an example of splitting a grid up as a set of square subsection. These and Figure 2

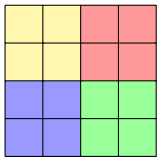

Figure 1: Tiled distribution

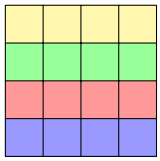

Figure 2: Rows distribution

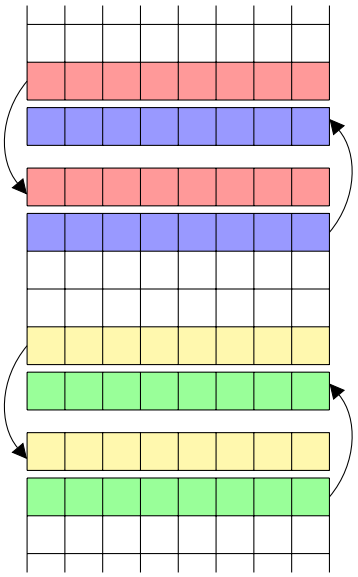

Figure 3: Cell regions and halos

The provided algorithm was optimised using shared memory parallelism with OpenMP. This means every every thread has access to the same region of memory which is manipulated during the calculations. When migrating a program to use distributed memory the important consideration is what memory is required by each node.

3 SOA vs AOS

4 OpenMP MPI

5 Non-Blocking Send

6 OpenCL