Optimisation of 2D lattice Boltzmann method using CUDA

Adam Koleszar
Supervisor: Irene Moulitsas

School of Aerospace, Transport and Manufacturing
Cranfield University

Background

The lattice Boltzmann method is a novel but already well-known mesoscopic method in computational fluid dynamics that is able to solve complex problems where the traditional Navier-Stokes approach could hardly work.

CUDA is a proprietary GPU programming API developed by nVidia© for their GPUs only. With it developers program general parallel algorithms run by the graphic processor in the computer.

The goal of this project to optimise and enhance an existing LBM implementation in CUDA[1]. The code had 3 different collision models implemented and validated, namely: BGKW, TRT and MRT.

Research

Initially the original CUDA LBM code was examined and tried for the lid-driven cavity geometry. Then the program was profiled for running times and memory usage using the built-in time measurement functions.

Later one-by-one all of its functions were rewritten or optimised with CUDA specific techniques to speed them up, reduce memory usage based on the knowledge of the underlying architecture.

These were our main considerations:

- Reduce the number of threads
- Unfold loops inside GPU functions
- Rearrange boundary conditions to use less resources
- Implement the residual norms computation on GPU

The program was also enhanced with the following features:

- Command-line interface
- Unit tests
- Result comparison functionality for validation against the original program
- User manual and documentation[2]

Key Results

The final program has all of its predecessor's functionality but it has a 30x speed-up on average and can reach up to 45x in special cases. It has also at least halved memory usage.

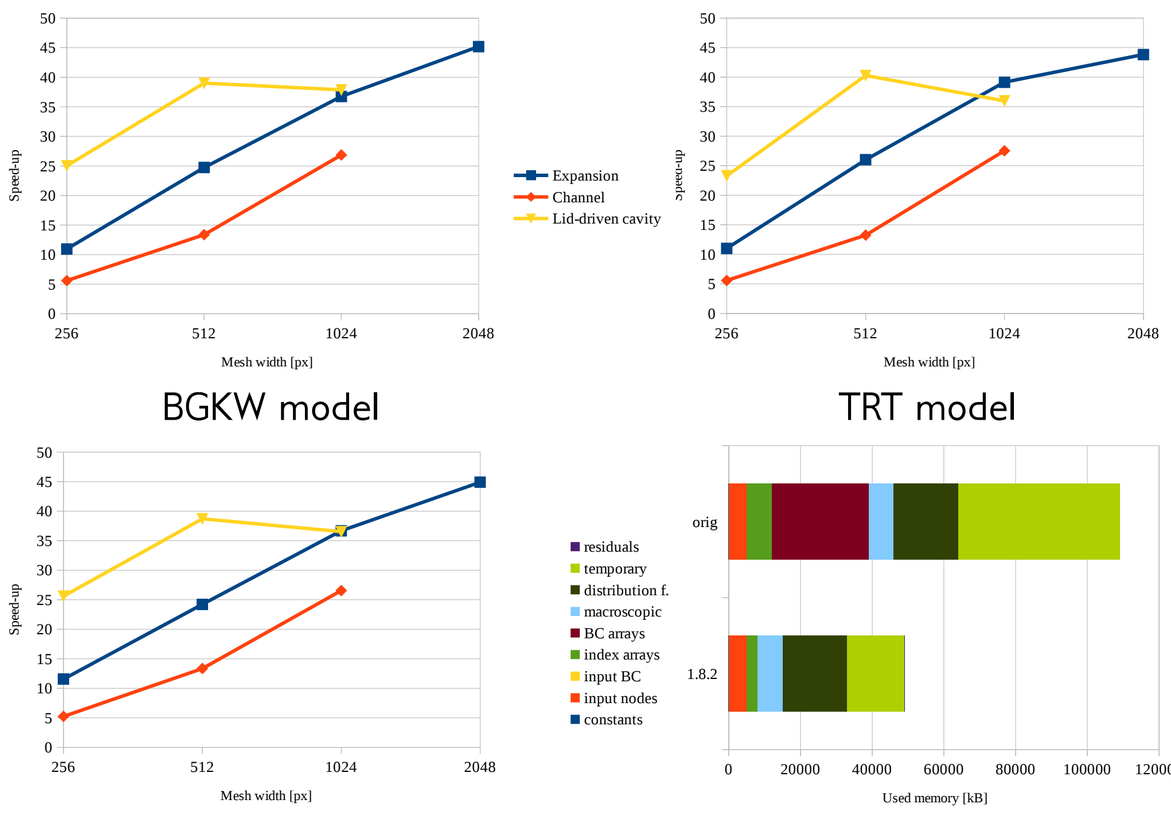

Figure 1. Speed-up diagrams for the different meshes and collision models and the reduced memory usage

Conclusion

Thanks to changes implemented the solver improved in speed with reduced memory usage. The validation revealed that it holds all the qualities and numerical properties as the original version.

Thanks to other improvements on the code this project became more maintainable and easier to improve.

[1] T. I. Józsa – Parallelization of lattice Boltzmann method using CUDA platform, Cranfield 2014

[2] *http://public.cranfield.ac.uk/e102081/LBM*