Journal of Computational Physics 275 (2014) 143–153

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

High-performance implementations and large-scale validation of the link-wise artificial compressibility method

Christian Obrecht [a,*], Pietro Asinari [b], Frédéric Kuznik [a], Jean-Jacques Roux [a]

[a] CETHIL UMR 5008 (CNRS, INSA-Lyon, UCB-Lyon 1), Université de Lyon, France
[b] SMaLL, Dipartimento Energia, Politecnico di Torino, Italy

A R T I C L E I N F O

Article history:
Received 17 December 2013
Received in revised form 17 June 2014
Accepted 23 June 2014
Available online 10 July 2014

Keywords:
Computational fluid dynamics
Link-wise artificial compressibility method
High-performance computing
Lid-driven cubic cavity
CUDA

A B S T R A C T

The link-wise artificial compressibility method (LW-ACM) is a recent formulation of the artificial compressibility method for solving the incompressible Navier–Stokes equations. Two implementations of the LW-ACM in three dimensions on CUDA enabled GPUs are described. The first one is a modified version of a state-of-the-art CUDA implementation of the lattice Boltzmann method (LBM), showing that an existing GPU LBM solver might easily be adapted to LW-ACM. The second one follows a novel approach, which leads to a performance increase of up to 1.8× compared to the LBM implementation considered here, while reducing the memory requirements by a factor of 5.25. Large-scale simulations of the lid-driven cubic cavity at Reynolds number Re = 2000 were performed for both LW-ACM and LBM. Comparison of the simulation results against spectral elements reference data shows that LW-ACM performs almost as well as multiple-relaxation-time LBM in terms of accuracy.

1. Introduction

Although the use of unstructured meshes is widespread in computational fluids dynamics (CFD), alternative approaches using Cartesian grids such as the lattice Boltzmann method (LBM) have gained increasing interest in recent years. However, while unstructured meshes are specifically intended for representing complex boundaries, Cartesian grid approaches need additional techniques to address this issue. Using nested meshes with hierarchical data structures such as octrees [8] is a possible method at the expense of the regularity of the data access pattern. Another way consists in incorporating additional treatments for boundary nodes based on techniques such as immersed boundary methods [11] or cut cell methods [10], inaccurate resolution of the boundary layers being a possible shortcoming. Nevertheless, from a computational standpoint, CFD solvers based on uniform Cartesian meshes are especially well-suited for high-performance implementations on massively parallel processors such as graphics processing units (GPUs) [16].

Sharing many similarities with the LBM, the artificial compressibility method (ACM) has been recently given a novel formulation, known as the link-wise ACM (LW-ACM) [3], which involves a finite set of links on a regular Cartesian mesh. Besides other interesting features, the LW-ACM enables to use specific techniques from both LBM and finite differences. In this paper, we describe two GPU implementations of the LW-ACM in three dimensions within the framework of the NVIDIA CUDA technology. Given the algorithmic similarities between LBM and LW-ACM, our first approach reinvests common GPU

* Corresponding author.
E-mail address: christian.obrecht@insa-lyon.fr (C. Obrecht).

http://dx.doi.org/10.1016/j.jcp.2014.06.062

implementation techniques of the LBM [13]. However, the LW-ACM updating rule makes possible to recover all the necessary informations from the hydrodynamic variables of the fluid. Hence our second implementation takes advantage of this specific feature to reduce considerably the memory requirements as well as the amount of data transferred between GPU and device memory. In addition, for validation and comparison purposes, we performed large-scale simulations of the lid-driven cubic cavity using either LW-ACM or LBM, and matched our simulation results against highly accurate reference data.

The remainder of the paper is organised as follows. In Section 2, we briefly introduce the LW-ACM and discuss its algorithmic aspects. Section 3 describes the two GPU implementations of LW-ACM. In Section 4, we report performance of both LW-ACM implementations and compare these results with a state-of-the-art CUDA LBM solver. Section 5 presents our simulations of the lid-driven cubic cavity and Section 6 provides some concluding remarks.

2. Link-wise artificial compressibility method

2.1. Artificial compressibility equations

The artificial compressibility method (ACM), which was first introduced by Chorin in 1967 [6], is a numerical approach for solving the incompressible Navier–Stokes equations (INSE). Using the Einstein summation convention, the INSE are expressed as:

$$\partial_t u_i + u_j \partial_j u_i = -\frac{1}{\rho_0}\partial_i p + \nu \partial^2_{jj} u_i + F_i, \tag{1}$$

$$\partial_j u_j = 0, \tag{2}$$

where u_i are the components of the fluid velocity $\boldsymbol{u}$, p is the pressure, F_i are the components of the external force per unit mass, ρ_0 is the density, ν is the kinematic viscosity, and t is the time. The indices $i, j = 1, 2, 3$ refer to the spatial coordinates.

Since pressure does not appear in Eq. (2), i.e. the continuity equation, most mainstream numerical methods resort to the derived pressure Poisson equation:

$$\partial^2_{jj} p = -\rho_0 \partial^2_{ij}(u_i u_j). \tag{3}$$

It should be noted that the adoption of an implicit time-marching, although natural, impairs the adaptability of these approaches to massive parallelism.

The ACM is in contrast based on the artificial compressibility equations (ACE), a modified form of the INSE in which the continuity equation is replaced by:

$$\partial_t \rho + \partial_j u_j = 0, \qquad p = \rho/\zeta, \tag{4}$$

where ρ is defined as the artificial density, ζ as the artificial compressibility, and $p = \rho/\zeta$ as the artificial equation of state. The ACE yield an artificial speed of sound: $c_s = 1/\sqrt{\zeta}$, and thus an artificial Mach number: $\mathrm{Ma} = \sqrt{\zeta} \times \max \|\boldsymbol{u}\|$.

The presence of the pressure time derivative in Eq. (4) allows for explicit time-integration. Although ACM was primarily intended for steady flows, it is known to yield also accurate solutions for the time-dependent INSE in the limit of vanishing Mach number [15].

2.2. Link-wise formulation

The LW-ACM is a discrete formulation of the ACM within a framework similar to the one of the LBM. It operates on a regular Cartesian spatial mesh of mesh size δx with a regular time step δt. In accordance with the established practice of LBM, we shall express all following quantities in terms of lattice units, i.e. adopt δx as unit of length and δt as unit of time.

The mesh is associated to a lattice stencil, i.e. a finite set of velocities $\{\boldsymbol{\xi}_\alpha\}$ where $\alpha = 0, \ldots, Q-1$. This stencil is usually chosen such as to link the mesh points to some of their nearest neighbours on the mesh. In the present work, we used the three-dimensional D3Q19 stencil, which is represented in Fig. 1. The coordinates of the $\boldsymbol{\xi}_\alpha$ velocities in the D3Q19 stencil are defined as:

$$[\boldsymbol{\xi}_\alpha] = \begin{cases} (0,0,0) & \alpha = 0, \\ (\pm1,0,0), (0,\pm1,0), (0,0,\pm1) & \alpha = 1,\ldots,6, \\ (\pm1,\pm1,0), (\pm1,0,\pm1), (0,\pm1,\pm1) & \alpha = 7,\ldots,18. \end{cases} \tag{5}$$

The hydrodynamics is represented by a set of Q dependent variables $\{f_\alpha\}$ defined at the mesh points such that:

$$\rho = \sum_\alpha f_\alpha, \tag{6}$$

$$\rho \boldsymbol{u} = \sum_\alpha f_\alpha \boldsymbol{\xi}_\alpha. \tag{7}$$