Journal of Computational Physics 284 (2015) 22–39

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Towards an ultra efficient kinetic scheme. Part III: High-performance-computing ☆

Giacomo Dimarco [a,b], Raphaël Loubère [b], Jacek Narski [b,*]

[a] Department of Mathematics and Computer Science, University of Ferrara, Ferrara, Italy
[b] Université de Toulouse, UPS, INSA, UT1, UTM, CNRS, UMR 5219, Institut de Mathématiques de Toulouse, F-31062 Toulouse, France

ARTICLE INFO

Article history:
Received 15 May 2014
Received in revised form 12 December 2014
Accepted 13 December 2014
Available online 18 December 2014

Keywords:
Kinetic equations
Parallel computation
GPU
OPEN-MP
CUDA
Discrete velocity models
Boltzmann-BGK equation
Semi-Lagrangian schemes
Three dimensional simulations

ABSTRACT

In this paper we demonstrate the capability of the fast semi-Lagrangian scheme developed in [20] and [21] to deal with parallel architectures. First, we will present the behaviors of such scheme on a classical architecture using OpenMP and then on GPU (Graphics Processing Unit) architecture using CUDA. The goal is to prove that this new scheme is well adapted to these types of parallelizations, and, moreover that the gain in CPU time is substantial on nowadays affordable computers. We first present the sequential version of our high-order kinetic scheme and focus on important details for an effective parallel implementation. Then, we introduce the specific treatments and algorithms which have been developed for an OpenMP and CUDA parallelizations. Numerical tests are shown for the full 3D/3D simulations. These assess the important speed-up factor of the method gained between the sequential code and the parallel versions and its very good scalability which makes this approach a real competitor with respect to existing schemes for the solution of multidimensional kinetic models.

1. Introduction

In this paper we deal with a new numerical method for the solution of the space inhomogeneous BGK-Boltzmann equation. In particular, the aim of this work is to extend the fast semi-Lagrangian approach developed in [20,21] to exploit high performance computing resources using OpenMP and Graphical-Processor-Unit (GPU) under CUDA language.

Kinetic models have been originally derived for dealing with dilute gases which are far from thermodynamical equilibrium [9,17]. Nowadays, they are used for many different applications ranging from rarefied gas dynamics and plasma physics to biological and/or socio-economic models. From the computational point of view, there are many difficulties associated with the numerical discretization of kinetic equations. One of the most important is certainly the dimensionality of the problem. The distribution function which defines at each instant of time the state of the system depends on seven independent variables: three coordinates in physical space, three coordinates in velocity space and the time. As a consequence, in most of the cases, especially for multidimensional simulations, the numerical methods which are employed when one wants to deal with more realistic physical problems, are based on stochastic Monte Carlo methods, see for instance [1,8,7,37]. These approaches exhibit a lower computational cost with respect to deterministic methods such as finite volume or

☆ This work was supported by the ANR JCJC project "ALE INC(ubator) 3D" (ANR-11-JS01-0012) and the ANR project 'MOTIMO' (ANR-11-MONU-009-01).
* Corresponding author.
E-mail addresses: giacomo.dimarco@unife.it (G. Dimarco), raphael.loubere@math.univ-toulouse.fr (R. Loubère), jacek.narski@math.univ-toulouse.fr (J. Narski).

http://dx.doi.org/10.1016/j.jcp.2014.12.023
0021-9991/

spectral methods [36,23,22,27,41,42]. On the other hand they are only low accurate and convergence is general very slow. Moreover, in non-stationary situations the impossibility of using time averages to reduce the fluctuations leads, as in the case of deterministic methods, to computationally expensive simulations, even if recently new approaches have tried to tame the problem [33,18,19,15].

To avoid the cost related to the high dimensionality of the problem, in [20,21], the authors have developed a new deterministic method to solve kinetic equations called Fast Kinetic Scheme (FKS). The method is based on the so-called discrete velocity methods (DVM) [39,36] and on the semi-Lagrangian approach [11,12] for the solution of the linear transport part of the kinetic model. Following the DVM approach [4,36,39,40], the velocity space is discretized into a set of fixed velocities. Consequently the original kinetic equation is then replaced by a set of linear transport equations plus a coupling interaction term corresponding to the collision operator. The purpose of the method described in [20,21] was to drastically reduce the cost of the transport part of the discrete velocity model using a semi-Lagrangian technique, allowing to exactly solve the transport part on the entire domain at a negligible cost: the shape of the distribution function is fixed once for all in the space domain and no reconstructions of the distribution function are needed anymore to find the feet of the characteristic. Hence, the cost of the solution of the original kinetic equation is almost entirely due to the projection of the solution onto the grid to compute the collision operator. The resulting scheme shares many analogies with classical semi-Lagrangian methods [23,11,12,30,31,10] and particle schemes [1,3]. However, contrarily to the latter, the scheme is as fast as a particle method and provides a fully deterministic numerical solution exempt of any source of statistical error.

Although the authors have shown that the full six dimensional problem (3D in space and 3D in velocity space) can be solved in one day or less on a single processor laptop on acceptable meshes ($100 \times 100 \times 100$ in space and $15 \times 15 \times 15$ in velocity) for a simple kinetic equation consisting of transport plus a relaxation BGK operator [29], the cost of such approach remains still too high if one wants to deal with more realistic numerical simulations. Thus, it appears to be natural to take advantage of high performance computing. In literature there exist several attempts about the parallelization of codes for kinetic models. We recall the works of Frezzotti and co-authors on the extension of finite difference and Direct Simulation Monte Carlo methods for the BGK equation on GPU [24–26], the extension of the deterministic spectral method of Gamba and Tharkabhushaman [27] to CPU-based parallelization using OpenMP and MPI by Haack [28,32], the work of Malkov and Ivanov on GPU [35] and the work of Dumbser et al. on the parallelization of implicit solver using MPI [44]. Parallelization of particle method or Lattice Boltzmann method has been explored under GPU architecture for instance by de Vuyst et al. [13,14].

The purpose of this paper is to explore the possibility of adapting the Fast Kinetic Scheme to parallel architectures and to show its performance. Here we have adopted the following philosophy:

- A parallel version must be simple enough to be (re)usable with nowadays available personal parallel computers. In other words we do not address, for the moment, the question of a massively parallel version of our scheme running under thousands of processors.
- A parallel version must use at the best the available computer resources (GPU, multi-core, etc.) without demanding a whole revamp of the sequential version.

As a consequence we have developed and then tested two FKS 3D $\times$ 3D parallel codes: one ultra-light parallel version using classical CPU under OpenMP, and, one more advanced version taking advantages of Graphical-Processor-Unit (GPU) under CUDA language.

The article is organized as follows. In Section 2, we present the model and Fast Kinetic numerical Scheme (FKS). Section 3 is devoted to the specific treatments developed for the two parallel architectures along with the algorithms. Several 3D $\times$ 3D test problem results and a scalability study are presented in Section 4 to assess the efficiency of the parallel versions of the method. Conclusion and future developments are finally exposed in Section 5.

2. Kinetic equations and Fast Kinetic Scheme

In this section we present the model and the numerical method originally designed in [20,21]. We refer the reader to the cited papers for more details.

2.1. Boltzmann-BGK equation

Let us consider the six-dimensional kinetic equation [9,29]

$$\partial_t f + \boldsymbol{V} \cdot \nabla_{\boldsymbol{X}} f = \frac{1}{\tau}(M_f - f), \tag{1}$$

with the initial condition

$$f(\boldsymbol{X}, \boldsymbol{V}, t = 0) = f_0(\boldsymbol{X}, \boldsymbol{V}). \tag{2}$$