Advances in Engineering Software xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

Parallel implementation of hyper-dimensional dynamical particle system on CUDA

Jan Mašek[*,a], Miroslav Vořechovský[b]

[a] Institute of Structural Mechanics, Brno University of Technology, Veveří 331/95, Brno 602 00, Czech Republic
[b] Institute of Structural Mechanics, Brno University of Technology, Veveří 331/95, Brno 602 00, Czech Republic

ARTICLE INFO

Keywords:
Particle dynamical system
Parallel implementation
NVIDIA CUDA
On-chip memory
Global memory
Atomic operations
Serialization of threads

ABSTRACT

The presented paper deals with possible approaches to parallel implementation of solution of a hyper-dimensional dynamical particle system. The proposed implementation approaches are generally applicable for similar particle systems of interest in various research and engineering fields. The original motivation for the present work was a simulation of particles that represent a space-filling design to be optimized for further use in design of experiments. Due to the underlying purpose of this particle system, the dimension of the particle system of interest is considered to be entirely arbitrary. Such a hyper-dimensional space is further folded into a periodically repeated domain.

The theoretical background of the proposed particle system is provided along with the derivation of equations of motion of the dynamical system. As the complexity of the system is not limited by the number of particles nor the number of dimensions, the possibilities of utilizing the GPGPU platform are more restricted in comparison with today's fast parallel implementations of common particle systems.

Two distinct approaches to parallel implementation are presented, one aiming at a generalized usage of the fast on-chip resources, the other entirely relying on the GPU's on-board global memory. Despite unambiguous mutual differences in their performance, both parallel implementations deliver major speedup compared to the single-thread CPU solution as well as a better scaling of execution time when increasing both the number of particles and dimensions.

1. Introduction

During the recent decade, researchers dealing with simulation of particle systems acquired a rather powerful computing platform with the development of general purpose computing on graphic processor units (GPGPU). With the great computational power offered by the GPGPU architecture, increasing number of formerly non-calculable problems are now being solved. Particle systems are nowadays simulated in numerous engineering and research fields. Molecular dynamics, material and mechanical engineering or astrophysics are only a few examples of these.

Namely the astrophysics simulations of vast scenarios of forming galaxies with tens of thousands of planets-particles are now possible to compute, see e.g. [1–3]. Unlike the systems of mutually attracting celestial bodies simulated by the astrophysicists, there exist similar particle models without a direct physical analogy to the purpose of their simulation. This is also the instance of the dynamical particle system as considered further in the presented paper.

The proposed system of *mutually repelling* particles is assembled to serve as an optimization tool for obtaining *uniformly* distributed point samples. Such optimized samples may find utilization in dozens of research problems, an interesting instance of which is the statistical sampling for numerical integration of an arbitrary function – Monte Carlo sampling.

Numerical integration of the Monte-Carlo type requires sampling of points that are uniformly distributed within a *design domain*. The design domain is considered to be the domain of sampling probabilities (values of the joint distribution function - the domain of copulas) which is a unit hypercube $[0, 1]^{N_{\text{var}}}$, where N_{var} is the dimension of the design domain and also the number of random variables of the integrated function.

The problem of using an ideally distributed set of finite number of integration points rises also in numerous engineering and research fields. While sampling from a random vector or integrating an unknown function, achieving a uniform layout of integration points is the only possible way for minimization of the lower bound of the resulting error, see e.g. the Koksma–Hlawka inequality [4–6].

Many criteria for "uniformity" have been put forth over the past

* Corresponding author.
E-mail addresses: jan.masek1@vut.cz (J. Mašek), vorechovsky.m@vut.cz (M. Vořechovský).

https://doi.org/10.1016/j.advengsoft.2018.03.009
Received 18 July 2017; Received in revised form 31 January 2018; Accepted 20 March 2018

years, aiming to serve for evaluation or optimization of distribution of N_{sim}points within a unit hypercube of dimension N_{var}. These criteria often investigate mutual distances between integration points with a tendency to prefer designs with points equally distant from each other.

Some other criteria can be shown to have analogies with physical problems. As an elegant instance of these, we consider the Audze-Eglājs (AE) criterion [7] and the generalized ϕ-criterion [8], respectively. The limiting case of the ϕ-criterion is the MaxiMin criterion [9] that prefers designs maximizing the distance between the closest pairs of points.

All of these criteria serve for evaluation of quality of point layouts and can be reinterpreted as a potential energy of a system of charged particles with repulsive forces. The objective of these criteria then lies in minimization of potential energy of a particle system and the positions of the particles are considered to be the coordinates of sampling points in the unit design hypercube.

During the recent years, it has been shown that the Audze-Eglājs criterion suffers from existence of boundaries of the design space [10,11]. A remedy of this behavior was proposed [10,11], assuming periodically extended design hypercube and thus achieving a design domain without boundaries, see Fig. 1b. Since then, it has been proved that optimization of point layouts by the introduced Periodic Audze-Eglājs (PAE) criterion leads to statistically uniform designs (from design to design) and to well distributed set of points for every single point layout.

This paper is based upon the conference paper [12], but the present paper provides much broader context, complementing the already proposed implementation (which relies on the GPU's on-board memory) by the recently assembled implementation utilizing the faster but limited on-chip memory. The in-detail study of thread serialization techniques along with the performance comparison of using atomic operations is not part of this paper and remains as a reference to the conference contribution [12].

Since the initial motivation for simulation of such a particle system was in optimization of point samples using the Audze-Eglājs criterion, the criterion itself is presented in Section 2 and the paper follows with derivation of equations of motion of the physically analogical system of charged particles in Section 3.

A brief review of particle simulation algorithms is presented in Section 4 and the particle system of interest is set into the context of today's GPGPU particle implementations and solution algorithms. Section 5 follows with an analysis of requirements on the solution implementation.

Hardware limitations rising from the unrestrained dimensionality of the particle system at hand are discussed in detail and reasons for both implementation approaches are justified, see Section 6. There, two different ways of data storage and the associated algorithms are presented.

Finally, the performance of both parallel implementations is provided and further compared to the single-thread CPU implementation in Section 7.

2. Audze-Eglājs Optimization criterion

The value of the Audze-Eglājs criterion can be understood as the amount of potential energy stored within a system of mutually repelling particles. The potential energy accumulated in particle interactions depends on distances between all pairs of particles.

The Euclidean distance between points i and j in N_{var}-dimensional space, L_{ij}, can be expressed as a function of their coordinates:

$$L_{ij} = \sqrt{\sum_{v=1}^{N_{\text{var}}} (x_{i,v} - x_{j,v})^2} = \sqrt{\sum_{v=1}^{N_{\text{var}}} (\Delta_{ij,v})^2}, \tag{1}$$

where $\Delta_{ij,v} = |x_{i,v} - x_{j,v}|$ is the difference in their positions projected onto the axis v. Let us assume that the points i and j with their mutual distance L_{ij} are repelled by the force F_{ij} induced by the potential energy E_{ij}:

$$E_{ij}(L_{ij}) = \frac{1}{L_{ij}^2} = \int_{\infty}^{L_{ij}} F_{ij}(x)\mathrm{d}x. \tag{2}$$

By differentiating the energy potential with respect to the distance, L_{ij}, the *repulsive* force is obtained: $F_{ij}(L_{ij}) = 2L_{ij}^{-3}$. As the particle system contains N_{sim} interacting particles, the total potential energy of the system is a sum of contributions from all $\binom{N_{\text{sim}}}{2}$ individual pairs:

$$E_p = \sum_{i=1}^{N_{\text{sim}}-1} \sum_{j=i+1}^{N_{\text{sim}}} E_{ij} = \sum_{i=1}^{N_{\text{sim}}-1} \sum_{j=i+1}^{N_{\text{sim}}} \frac{1}{L_{ij}^2}. \tag{3}$$

The total potential energy in Eq. (3) represents the value of the Audze-Eglājs criterion to be minimized.

A simple and efficient improvement that considers a periodic extension of the design space has been proposed in [10]. After some simplification, one can derive equations for *periodic* Audze-Eglājs criterion (PAE) by replacing $\Delta_{ij,\,v}$ in Eq. (1) with its periodic variant:

$$\overline{\Delta}_{ij,v} = \min(\Delta_{ij,v},\ 1 - \Delta_{ij,v}). \tag{4}$$

With such a redefined projection, a new metric is obtained and the distance between points i and j, called the periodic length $\overline{L}_{ij}$, becomes the actual shortest distance between point i and the nearest image of point j [10], also see Fig. 1:

$$\overline{L}_{ij} = \sqrt{\sum_{v=1}^{N_{\text{var}}} (\overline{\Delta}_{ij,v})^2}. \tag{5}$$

We note that using the nearest image of point j with respect to point i does not cover a true periodic repetition of the design domain. In a complete periodic repetition, infinite number of images of the point j would interact with the point i. The presented approach is a simplification that can be shown [10] to yield identical results to the fully repeated system in case of sufficient point count, N_{sim}.

If the number of points in the original domain is too low for assembly of the desired self-similar pattern[1], considering additional periodical images of particles is advised. As argued in [13], this is due to an insufficient resolution between short and long-range forces in the system. Another remedy is also to use stronger differentiation between short and long-range forces, that is to rise the exponent upon the mutual distance L_{ij} in the energy potential, see Eq. (3), to a greater value. As shown in [13], the exponent shall be no lower than $N_{\text{var}} + 1$.

For greater particle systems, there is no practical need to consider more than one image of each particle as long as the energy potential uses a correct value of the exponent. Then, additional periodical images are not considered as a true periodic extension is not necessary for achieving an optimal space-filling design.

The AE criterion is originally meant for evaluation of uniformity of a fixed set of particles by calculating the overall potential energy. This potential energy is stored in all pairwise particle interactions considered in a radial sense.

For the purposes of dynamical simulation, however, each vector of the mutual repelling force $F_{ij} = \ddot{x}_{ij}$ induced by such an energy potential needs to be decomposed into all of its N_{var}orthogonal components which then provide the information about the actual accelerations $\ddot{x}_{ij,v}$ in each of N_{var}:directions. The derivation of equations of motion of the particle system is discussed in what follows.

[1] Simplest self-similar space-filling patterns can be assembled from simplest objects which contain volume in the particular dimension N_{var}: line in 1D (2 points), triangle in 2D (3 points), tetrahedron in 3D (4 points), etc.