UNIUNEA EUROPEANĂ GUVERNUL ROMÂNIEI Fondul Social European POSDRU 2007-2013 Instrumente Structurale 2007-2013 MINISTERUL EDUCAŢIEI NAŢIONALE OIPOSDRU UNIVERSITATEA TEHNICĂ CLUJ-NAPOCA

Investeşte în oameni!

Proiect cofinanţat din Fondul Social European prin Programul Operaţional Sectorial pentru Dezvoltarea Resurselor Umane 2007 – 2013
Axa prioritară: 1 „Educaţia şi formarea profesională în sprijinul creşterii economice şi dezvoltării societăţii bazate pe cunoaştere"
Domeniul major de intervenţie: 1.5 „Programe doctorale şi postdoctorale în sprijinul cercetării"
Titlul proiectului: „Parteneriat inter-universitar pentru excelenta in inginerie - PARTING"
Cod Contract: POSDRU/159/1.5/S/137516
Beneficiar: Universitatea Tehnică din Cluj-Napoca

eng. **Adrian SABOU**

PH.D.THESIS

-Summary-

PARALLEL SIMULATION OF THREE-DIMENSIONAL PARTICLE-BASED MODELS

Scientific supervisor

Prof. **Dorian GORGAN, Ph.D.**

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA
FACULTY OF AUTOMATION AND COMPUTER SCIENCE

2014

Summary

1. Introduction, context and motivation

Modeling and simulating three-dimensional dynamic surfaces represents one of the main research areas of Computer Graphics and probably one of the most fascinating. The most prominent techniques for such simulations are physically-based methods, which have become an important part of the entertainment industry and are regularly employed in computer games, feature films and various computer based animations, for visual effects involving interacting fluids, textile materials, deformable and rigid bodies. One of the most important sub-domains of physically-based modeling is particle-based modeling, where surfaces are approximated through sets of discrete points having various physical properties such as mass, volume, speed, acceleration, and the behavior of such surfaces along with their interaction with the environment is governed by the laws of physics, more specifically by the forces that act upon particles. Particle-based modeling is a natural choice since in the real world interactions concerning deformable surfaces occur at a molecular level and, in theory, given a sufficient number of particles, any such surface can be accurately modeled. Such modeling techniques outstrip by far traditional key frame animation in terms of level of realism and complexity.

Soft, deformable bodies exist all around us and we can no longer imagine a computer based simulation of the real world without emulating the behavior of such surfaces as realistically as possible. Physically-based models have been successfully used ever since they were introduced in order to develop computer generated visual content for an ever more difficult to impress audience. However, the strive for realism brought forth a new challenge, namely the strive for computational power and efficiency. As sequential computing hardware such as the Central Computing Unit (CPU) evolved, algorithms and techniques utilized suffered little or no change, the increase in performance being mainly obtained from higher CPU clocks, but as CPU speeds topped 4 GHz, the dreaded thermal barrier impeded further advancement in this direction. However, for some users this was not enough, thus the ``parallel era'' began. CPUs evolved into multicore architectures and new specialized hardware for parallel computing emerged in the form of Graphic Processing Units (GPUs). However, with this new computing direction came the need for efficient parallel techniques in order to optimally utilize existent resources.

Among scientific applications to benefit from GPU acceleration are physically-based simulations, the inherently parallel nature of physically-based models making them perfectly suited for GPU-based implementations. Thus, in the last decade, research into this domain produced highly efficient parallel methods and techniques for accelerating the most time consuming phases of the simulations, namely the numerical integration and the collision detection. However, as the resolution of simulated models increases in the strive for realism, single GPUs can no longer cope with ever more demanding scenarios, especially due to memory limitations. Therefore, in order to continue pushing the boundaries of parallel physical-based simulation, we need to turn our attention towards more scalable computing architectures such as graphics clusters.

It is our belief that particle-based simulation could largely benefit from utilizing such powerful computing architectures and thus the motivation for this thesis is centered on this idea, although challenges exist, mainly at application development level. Firstly, due to physical design and operating mode, both shared memory and distributed memory paradigms apply when designing applications that will run on GPU clusters. Adding the real-time attribute that such simulations usually require, along with the need to transform raw processed data into a form visually meaningful to the users, we end up with probably one of the most complex class of applications attempted to be implemented on such architectures. Main challenges include parallelization of physically-based models, implementing parallel numeric integration algorithms, parallel collision detection, distributed model processing on more cluster nodes, remote visualization and interaction with the simulated scene and, most importantly, ensuring communication and synchronization between all these processes. Thus any approach of physically-based simulation on GPU clusters must not limit itself to tackling each of the above challenges and optimizing these processes independently, but rather discover their interdependencies and treat them as modules contributing to efficient simulation, taking into account the aforementioned particularities of the underlying hardware. Furthermore, this new type of HPC computing should be made available to a broader audience through higher level programming interfaces. For example, powerful algorithms derived from numerical analysis may be developed for physically-based modeling by researchers in domains other than computer science, who may not want to be encumbered by all the intricate low level details of cluster workings. In this context, our work aims at researching the particularities of an optimized, modular and highly scalable application development architecture for designing physically-based simulation scenarios on GPU clusters.

2. Research Objectives

The main objective of this research is the optimization of interactive visual HPC applications such as physically-based simulations for running on GPU clusters. The objective is focused on researching the particularities of an optimized, parallel, distributed, modular and highly scalable application development architecture for designing particle-based simulation scenarios.

Other important objectives, derived from the main objective, are:

- to identify possible modular structures for the application development architecture;
- to identify communication patterns between individual modules;
- to propose a conceptual architecture based on the optimum modular design and communication interfaces identified;
- to implement and optimize individual modules based on adapting currently employed parallel techniques in physically-based modeling for usage in GPU clusters;
- to extend particle-based models and the application development architecture in order to use them outside the domain of computer animation;
- to test and evaluate the proposed architecture;

3. Thesis outline

The thesis is organized in eight chapters, with 151 bibliographic references. The contributions highlighted throughout this paper were published in 1 B+ scientific journal, 5 articles in international conferences indexed as ACM or IEEE and 3 presentations at different conferences, workshops and trainings.

Chapter 1 is an introduction to the research domain and this thesis, describing the background and motivation of the current research, along with the main research objectives.

Chapter 2 offers an analysis of parallel and distributed computing architectures. It focuses on shared-memory and distributed-memory systems, emphasizing advantages and disadvantages that each category holds, with regard to particle-based modeling. Modern shared-memory architectures such as the Graphics Processing Unit are analyzed and, at the end, we conclude that, in order to provide a fast and scalable solution for particle-based simulators, hybrid architectures such as graphics clusters may offer most advantages, by combining strong points from both shared-memory and distributed-memory systems.

Chapter 3 is a critical survey of models and techniques employed in physically-based simulations. It gives an overview of existing and related research on physically-based simulations, as well as attempting a high-level formal description of particle-based models. Several models for simulating cloth and fluids are analyzed from a parallelization potential point of view, such as particle-based and models based on continuum mechanics, as well as most utilized explicit and implicit numerical integration techniques and collision detection methods. The formal description of particle-based models abstracts the concept of particle in order to facilitate a possible extension of particle-based simulators outside the scope of computer animation. An initial modular structure for an application development architecture, based on the main steps required for particle-based simulations is described at the end of the chapter.

Chapter 4 discusses in detail the parallel techniques developed for accelerating particle-based simulations on graphics clusters. It covers the most important issues for parallel particle-based simulations such as model decomposition and distribution at GPU level and at graphics cluster level, hybrid CPU/GPU parallelism and parallel numerical integration. For model partitioning we decided on a static domain decomposition method that ensures minimal length frontiers and thus minimizes network traffic required for synchronization. We also proposed a technique that ensures minimal modification to the kernels used for the single machine approach by keeping an extended model on each processing node. CPU/GPU parallelism was employed in order to minimize idle times for the CPU while waiting for GPU tasks to complete. By employing a technique which decouples several steps of the simulation process, we were able to run the simulation in an out-of-phase manner and carry out CPU-based synchronization tasks while the GPU executed the complex computation tasks. Parallel explicit numerical integration is achieved using a two-pass data parallel approach, while, for implicit integration, we rely on a parallel version of the Conjugate Gradient algorithm. To further accelerate computation and reduce memory requirements, we developed an efficient technique to update the large sparse matrices involved in implicit integration directly into the Compressed Sparse Row storage format, by exploiting their regular structure.

Chapter 5 introduces *ParTSim*, a parallel, distributed, modular application development architecture for running particle-based simulations on graphics clusters. The architecture is based on a client-server approach and, at server level, is organized based on the main modules that were identified as indispensable for this kind of simulations, along with modules introduced by the particularities of the computation process on these distributed and remote computing architectures. Based on open standards like OpenCL and OpenGL, this architecture uses interoperability between general purpose computing and rendering on the same GPU, thus reducing memory transfers between device and host and greatly improving performance. The modular structure allows increased flexibility in combining various models with various numerical integration techniques and collision detection and resolve methods. This chapter also treats the issues that arise in the context of graphics clusters due to the distributed and remote nature of the processing. To obtain performant centralized visualization, we present an optimized parallel rendering technique. Based on a sort-last approach to parallel rendering and coupled with a Region-of-Interest algorithm, this technique greatly improves simulation performances when visualization of the entire scene and model is required on a single display device. Remote visualization and interaction is ensured by using a solution based on the VNC protocol that offers great performances for running interactive remote graphics applications even when lacking physical display devices as is the case of most GPU clusters.

Chapter 6 aims at extending particle-based models with regard to their initial scope. Based on the formal description in chapter 3, we identify possible correspondences between particle-based models employed in computer graphics and models belonging to other scientific domains, such as sociophysics and econophysics. This allows us to use the *ParTSim* architecture in order to accelerate simulations outside the field of soft-body dynamics. As a case study, we simulate an existing model for technology diffusion using a visual particle-based approach. We identify correspondences between the main elements of a traditional particle-based model and the elements of the sociophysics model and we run the simulation interactively using the *ParTSim* architecture.

Chapter 7 describes the experimental validation and evaluation of the *ParTSim* architecture and of all parallel simulation techniques presented in previous chapters. For performance evaluation we use metrics specific both to parallel computing and to computer graphics, such as execution time, speedup and frame rate.

Chapter 8 highlights the conclusions together with the main contributions and future work.

4. Contributions

The main contribution of this thesis is the development of powerful parallel techniques that allow particle-based models to be interactively simulated on graphics clusters. Other original contributions that have direct impact on achieving this major objective are listed below:

- defining a formal description of particle-based models, based on their constitutive elements, that facilitates a possible extension with regard to their initial scope, outside the field of computer graphics [2];

- the description of a parallel model decomposition technique for running particle-based simulations on GPUs and graphics clusters that minimizes border lengths and preserves global ordering of particles [5];
- the description and implementation of a hybrid CPU/GPU parallel technique that improves simulation performance through minimization of CPU idle times [4];
- the description and implementation of an out-of-phase simulation method that allows decoupling GPU kernel execution and CPU-based synchronization [4];
- the description and implementation of a parallel two-pass implementation of the velocity Verlet explicit integration technique for graphics clusters [4] [5];
- the description and implementation of a parallel update technique for the large sparse matrices that are involved in implicit integration that allows fast, efficient update directly into the Compressed Sparse Row format [3];
- the description and implementation of *ParTSim*, an optimized, parallel, distributed, modular and highly scalable application development architecture for designing particle-based simulation scenarios on graphics clusters [1];
- identifying the communication patterns between individual modules of the *ParTSim* architecture [1];
- the description and implementation of a GPGPU/Rendering pipeline interoperability mode based on OpenCL/OpenGL that avoids unnecessary memory transfers between host and device in order to render particle-based models [5];
- the description and implementation of an optimized sort-last approach to parallel rendering for particle-based simulations that uses a region-of-interest algorithm to reduce the amount of network traffic needed for synchronization [4];
- adapting remote visualization and interaction techniques and tools existent in the research literature for interactive particle-based simulations on graphics clusters [5];
- expanding particle-based models with regard to their initial scope, as to allow the simulation of models outside the domain of computer graphics, based on their formal description [2];
- validating and evaluating the performance gain obtained by using the proposed parallel, distributed techniques for GPU cluster-based simulations [3] [4] [5];
- validating and evaluating the *ParTSim* architecture by performing a visual performance comparison of explicit and implicit integration [1];
- validating and evaluating the particle-based simulation of sociophysics models [2];

5. Publications

Journal articles

- D. Copândean, A. Sabou and D. Gorgan, „Tehnici de interacţiune utilizator cu obiecte 3D modelate prin particule”, in Revista Română de Interacțiune Om-Calculator, vol. 7, no. 1, 2014, pp. 71-88;

Conference papers

[1] Sabou and D. Gorgan, „A parallel, distributed, high-performance architecture for simulating particle-based models”, in 2014 16th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), Sept 2014, in press;

[2] Sabou and D. Gorgan, „Interactive particle-based simulation of sociophysics models”, in 2014 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), Sept 2014, pp. 411--416.

[3] Sabou, D. Gorgan, and I. R. Peter, „Parallel implicit time integration for particle-based models on graphics clusters”, in 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO),May 2014, pp. 336--341;

[4] Sabou and D. Gorgan, „Physical simulation of 3d dynamical surfaces on graphics clusters”, in 2013 36th International Convention on Information Communication Technology Electronics Microelectronics (MIPRO), May 2013, pp. 292--297;

[5] Sabou, C. Mocan, and D. Gorgan, „Particle based modeling and processing of high resolution and large textile surfaces”, in 2012 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), Sept 2012, pp. 355—360;

Presentation at Conferences, Workshops and Trainings

- Sabou, „Parallel simulation of three-dimensional physically-based models”, in The Second graduate school within the EU COST Action IC0805 ``Heterogeneous computing - impact on algorithms'', 3-7 June 2013, Uppsala, Sweden;
- Sabou and D. Gorgan, „Parallel Real Time Simulation of Mass-Spring Models on Graphic Cluster Architectures”, in 3rd Workshop of COST 0805 - Open Network for High-Performance Computing on Complex Environments, 17-18 April 2012, Genova, Italy;
- Sabou and D. Gorgan, „Exploring a graphic cluster based solution for real-time virtual surgery”, in 2012 International Conference on Medical Education Informatics, 6-7 April 2012, Thessaloniki, Greece;