Abstract

We present a preliminary release of MCAFramework, an extensible system for the analysis and interactive visualization of lava flows simulations. The core of the system is SCIARA - fV2. It is the latest release of the SCIARA Cellular Automata family, and resides in a remote multi-GPU node which provides a multilayered GPU implementation in order to compute single or multiple simultaneous simulations. Experiment results are interactively visualized in real-time by means of a 3D graphics engine implemented in C++ and VTK and integrated in Qt GUI.

System Overview

- MCAFramework provides a scalable and modular architecture
- Interaction between the GUI and the 3D/2D Engine Module is guaranteed by Qt’s Signal/Slot communication mechanism
- The Graphics Engine module establishes a connection with the SCIARA model via socket protocol

3D/2D Graphics Engine

- Implemented in C++ and VTK and integrated in Qt Engine for generic Cellular Automata visualization
- Coordinates the rendering process and manages lights, cameras and scene tridimensional objects

Modelling Tools

- 3D models: structures with topology and geometry (triangle strip mesh)
- 2D models: color mapping strategy
- Contour, textures mapping, generics models

Visualization Tools

- Double visualization system:
  - 3D View: prospective phenomena simulation visualization
  - 2D View: orthogonal phenomena simulation visualization

Real-Time Interaction Tools

- Scene Interactor Style: cam management (zoom, rotation, pan)
- Cell picking: cell selection, cell state viewing and editing
- Simulation execution control (play, stop, pause)

Multi-GPU CUDA CA

- Provides a double CUDA Sciara - fV2 implementation layer:
  - CUDA SCIARA - fV2 simulator
    - Single simulation (Modelling Analysis, Data Analisys)
  - CUDA SCIARA - fV2 multisimulator
    - Large number of simultaneous simulations (Risk Maps, Genetics Algorithms)

SCIARA - fV2 model

SCIARA - fV2 is the latest release of the SCIARA Cellular Automata (CA) lava flows family:

$$\text{CA} = (Q, \Gamma, \psi, \delta_x, \delta_y, \delta_z)$$

- \(Q\) is the set of points defining the bi-dimensional cellular space
- \(\Gamma\) specifies the lava source cells (i.e. craters/vents)
- \(\psi\) identifies the pattern of cells that influence the cell state change
- \(\delta_x\), \(\delta_y\), and \(\delta_z\) are the finite set of global parameters (invariant in time and space)
- \(\delta: Q^3 \rightarrow Q\) is the deterministic cell transition function
- \(\gamma: Q \times N \rightarrow Q\) specifies the emitted lava thickness from the source cells

Simulating lava flows using CUDA

Single simulation strategies (SS)

- Whole cellular pace implementation
  - One thread-one CA cell
  - Two-dimensional static kernel grid
  - Memory hybrid approach
    - Global/Shared Memory
  - Dynamic grid implementation
    - One thread-one CA active cell
    - Two-dimensional dynamic kernel grid
    - Rectangular bounding box based

Conclusions

We have presented MCAFramework, an efficient Visualization System for Cellular Automata Lava Flows Simulation. A modular system architecture solution was adopted for guarantee a clear separation between the interactive GUI process (client) and computation process (server). Starting from the problem of accelerating the real-time visualization of complex phenomena, we implemented several approaches for single and multiple simultaneous running of lava flow simulations using CUDA and GPUL. The achieved results compared to CPU-based implementations, in terms of parallel speedup (FFS), were very significant. Thanks to CUDA the System can run the combined rendering and simulations at interactive frame rates.

Acknowledgments

This work was partially funded by the European Commission - European Social Fund (ESF) and by the Regione Calabria (Italy).