A GPU Accelerated Continuous-based Discrete Element Method for Elastodynamics Analysis

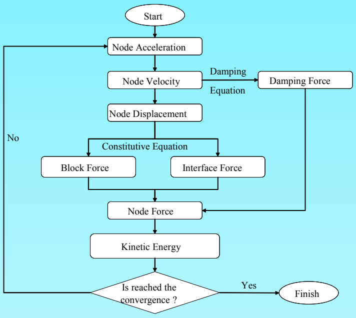
Zhaosong Ma, Chun Feng, Tianping Liu, Shihai Li
Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China
Email: marze@163.com

Introduction to CDEM

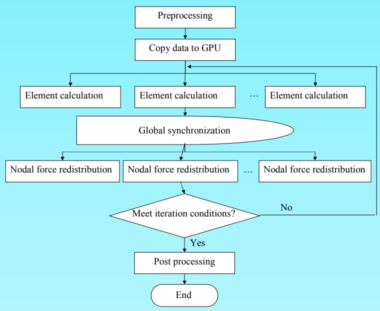
The Continuum-based Distinct Element Method (CDEM) is an approach to simulate the progressive failure of geological mass, which is mainly used in landslide stability evaluation, coal and gas outburst analysis, as well as mining and tunnel designing.

CDEM is the combination of Finite Element Method (FEM) and Distinct Element Method (DEM), It contains two kinds of elements, blocks and interfaces. A discrete block is consisted of one or more FEM elements, all of which share the same nodes and faces. An interface contains several normal and tangent springs, each connects nodes in different blocks. Inside the block the FEM is used, while for the interface, the DEM is adopted.

CDEM is an explicit time history-analysis FEM/DEM approach on finite difference principles and forward-difference approximation is adopted to calculate the progressive process through a time marching scheme. During the calculation, the dynamic relaxation method is used to achieve convergence in a reasonable period time with small time steps, and the convergence is reached when the total magnitude of the kinetic energy is minimized.



Serial Algorithm in CPU

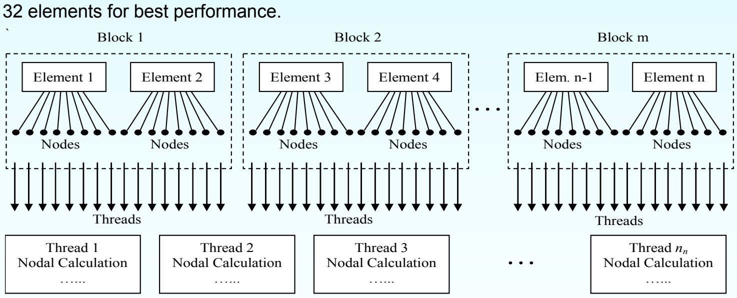


Parallel Algorithm in GPU

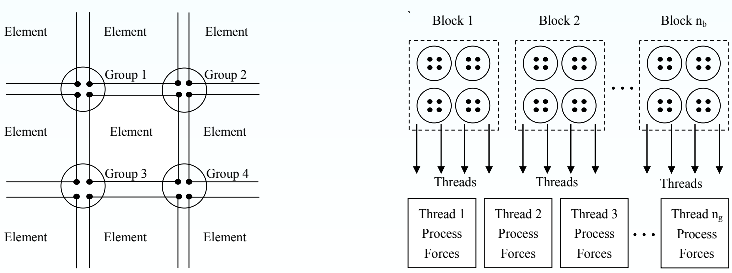
Solution Algorithm in Parallel

Unlike general CDEM algorithm which uses clone node strategy to ensure node consistency among elements, the parallel algorithm does the same work by means of a "nodal force group" strategy. The difference is out of reason of access conflict. For general CDEM, a node's force is cloned to the associated nodes just when the node's force calculation is done. This works well in serial execution, but may cause data access conflict in parallel execution, for various threads may clone their forces to a same node at the same time. Differently, the parallel algorithm uses a data structure to temporarily store the forces calculated. The nodal forces will be redistributed in a separate procedure after all of the node's force calculation is done.

The element calculation procedure is designed according to the specialty of GPU. As is known, a GPU consists of a set of SIMT (single-instruction, multiple-thread) multi-processors, which are mapped to CUDA blocks; each multi-processor has an instruction unit and several scalar processor cores, which are mapped to CUDA threads, and each scalar thread executes independently with its own instruction address and register state. Accordingly, the element calculation processes are divided into parallel CUDA blocks, each of which contains certain number of elements. In this work, a CUDA block contains 32 elements for best performance.

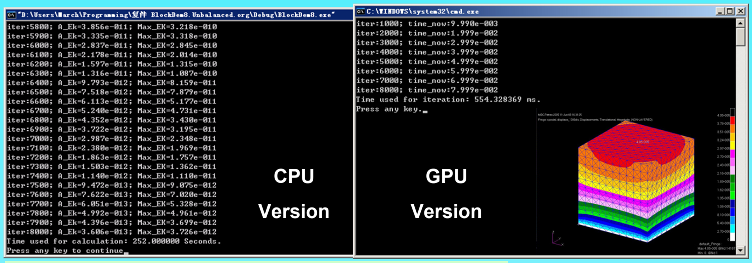


Thread Division Scheme for Element Calculation



Nodal Force Group

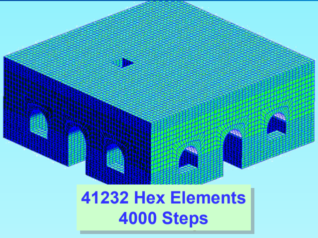
Case 1: Simple Model Comparison

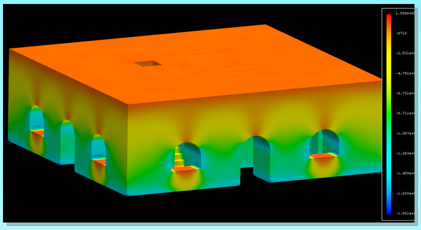


A 1000 elements brick, with bottom fixed and gravity applied

Speed Up Ratio Is 252 / 0.554 = 455 GeForce GTX285

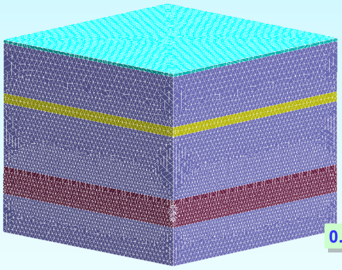
Case 2: Elastic Field Calculation of Great Wall Beacon

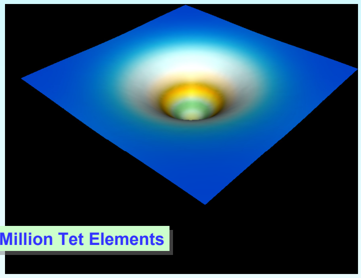




Time Cost for CDEM (GPU Version)	6.1 Sec	GeForce GTX285
Time Cost for CDEM (CPU Version)	95 Min	Core (TM) 2 1.83GHz
Time Cost for FLAC3D	520 Sec	Core Duo 3.0GHz

Case 3: Ground Settlement Induced by Deep Excavation

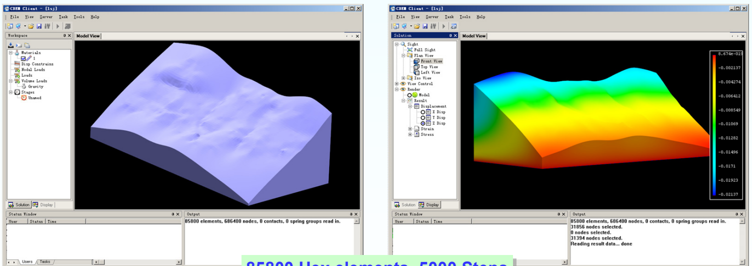




0.86 Million Tet Elements

Time Cost for Elastic Calculation	6 Min	GeForce GTX480

Case 4: Elastic Calculation of Landslide



85800 Hex elements 5000 Steps

Time Cost for CDEM (GPU Version)	12.5 Sec	GeForce GTX480
Time Cost for FLAC3D	1380Sec	Core Duo 3.0GHz