LexADV_EMPS : Explicit Moving Particle Simulation Framework

Objective and Motivation
We have been developing the open source CAE software, named ADVENTURE, which is a general-purpose parallel finite element analysis system and can simulate a large-scale analysis model with various supercomputers. Our aim in this research is to develop scientific libraries for the post-peta-scale simulation by the “particle methods” for the continuum mechanics.

The particle method regards a continuum as a set of particles, discretizes the physical laws governed by differential equations using interactions between the particles, and calculates the states and motions of the particles. Since the particles, as the calculation points, move on the time-marching processes, the particle method is superior to grid methods in terms of solving dynamic physical phenomena such as free surfaces and large deformations. However, the motion of particles makes it difficult to parallelize the particle method in distributed memory parallel computers.

We adopted the Moving Particle Simulation (MPS) method which is one of the most popular particle method as target solver and have been developing the LexADV_EMPS of the Explicit MPS framework which supports “Domain decomposition”, “Halo exchange” and “Dynamic load balance” as functions for parallel computing.

LexADV is free and open source software for large-scale numerical simulations of continuum mechanics problems. Currently, some beta version software was released.

Development of Explicit Moving Particle Simulation Framework and Zoom-Up Tsunami Analysis System
Kohei MUROTANI, Seichi KOSHIZUKA (University of Tokyo), Masao OGINO (Nagoya University), Ryuji SHIOYA (Toyo University)

Overview

Sample Solver : Distributed Memory Parallel Explicit MPS Implemented by LexADV_EMPS

Overview

- Development and operating environment
  - OS : UNIX, Linux
  - Compiler : C
  - Communication library : MPI

Dynamic load balance

HPC systems used in this research

<table>
<thead>
<tr>
<th>Machine name</th>
<th>Site</th>
<th>Processor</th>
<th>GFLOPS / Node</th>
<th>CoreNode</th>
<th>Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>K computer</td>
<td>RIKEN</td>
<td>Fujitsu V800</td>
<td>128</td>
<td>8</td>
<td>81,128</td>
</tr>
<tr>
<td>FX100</td>
<td>Nagoya Univ.</td>
<td>Fujitsu SIPL 50S</td>
<td>1,126</td>
<td>32</td>
<td>2,880</td>
</tr>
<tr>
<td>FX10</td>
<td>Univ. of Tokyo</td>
<td>Fujitsu SIPL 50S</td>
<td>236</td>
<td>16</td>
<td>4,800</td>
</tr>
<tr>
<td>CX400</td>
<td>Ogasawara Univ.</td>
<td>Intel Xeon</td>
<td>1,164</td>
<td>28 (2 cpus / node)</td>
<td>552</td>
</tr>
</tbody>
</table>

Halo area is generated.

The number of particles in each domain is equally partitioned.

Machine/ Site/ Processor/ GFLOPS / Node/ CoreNode/ Nodes

Parallel system (For distributed memory parallel computing)

- Two kind of sample solvers
  - Dambreak analysis of incompressible fluid analysis with free surface
  - Two floating objects analysis by interaction between fluid and rigid bodies
- Functions for Parallel computing by LexADV_EMPS
  - Domain decomposition
  - Dynamic load balance
  - Halo exchange
- Other function
  - Surface tension model by potential model
  - MPS gradient and Lagrange models
  - 1st, 2nd, 3rd and 4th order spatial derivative models
- Coming soon
  - Semi-implicit MPS method
  - Poisson equation solver with Neumann boundary condition by higher order spatial derivative models

Interaction between fluid and rigid bodies

1. Calculate all particles as fluid particles
2. Calculate a translation and a rotation of a rigid body from rigid body particles.
3. Perform the translation and the rotation for the rigid body from the original position.
   This algorithm corresponds to calculating a volume integral of forces for the rigid body.

Communication for many rigid bodies

- Communication for many rigid bodies occurs in only summation operation. (most important part)
- The data exchange is done for obtaining new center of gravity, translation and angular momentum (rotation) which has 3 components of double type respectively.
- These three values are shared in all nodes by “MPI_Allreduce()”

Halo exchange (communication among neighboring domains)

The number of particles is balanced among neighboring PE. Domain decomposition is performed.

Domain decomposition and Load balance LexADV_EMPS

[Diagram of domain decomposition and halo exchange]

- Domain decomposition and Halo exchange
  - Calculate domain decomposition and Halo exchange

Update velocity and position

- Calculate velocity and position

Halo exchange by LexADV_EMPS

Contact velocity and position

Contact velocity and position

Check load balance

- Check load balance

Poisson solver with Neumann boundary condition by higher order spatial derivative models

The data exchange is done for obtaining new center of gravity, translation and angular momentum (rotation) which has 3 components of double type respectively.

These three values are shared in all nodes by “MPI_Allreduce()”

Communication data size = (double type component) x (no. of rigid body particles) x (no. of rigid body particles)

For example, if 100 rigid body particles, communication size is only 400 (size of double) x 100 x 100 bytes.

[Diagram of halo exchange and communication]

Sample Solver : Distributed Memory Parallel Explicit MPS

Domain decomposition and Load balance LexADV_EMPS

Calculate pressure and gradient

Calculate pressure and gradient

Calculate pressure and gradient

Update velocity and position

Update velocity and position

Update velocity and position

Check load balance

Poisson solver with Neumann boundary condition by higher order spatial derivative models

The data exchange is done for obtaining new center of gravity, translation and angular momentum (rotation) which has 3 components of double type respectively.

These three values are shared in all nodes by “MPI_Allreduce()”

Communication data size = (double type component) x (no. of rigid body particles) x (no. of rigid body particles)

For example, if 100 rigid body particles, communication size is only 400 (size of double) x 100 x 100 bytes.

[Diagram of halo exchange and communication]
Development of Explicit Moving Particle Simulation Framework and Zoom-Up Tsunami Analysis System

Kohei MUROTANI, Seiichi KOSHIZUKA (University of Tokyo), Masao OGINO (Nagoya University), Ryuji SHIOYA (Toyo University)

Zoom-Up Tsunami Analysis System

Tohoku area was severely damaged by the tsunami of the Great East Japan Earthquake on 2011. Our target is to simulate impact by three kind of tsunami analyses on urban areas. First, tsunami is running-up in urban area and inundates into a building. Second, floating objects have collisions with each other and buildings. Third, stress of buildings from fluid pressure is estimated.

Zoom-up tsunami analysis system have been built in order to solve our target problems. In our system, zoom-up analysis by three stages analyses is adopted to solve a large area from an epicenter to an urban area. In the first stage, the two-dimensional shallow-water analysis is solved in the area of about 1000 km x 1000 km from the epicenter to the coastal areas. In the second and third stages, the three-dimensional tsunami run-up analyses are solved for the coastal areas using the distributed memory parallel explicit MPS method implemented by LexADV_EMPS. Today, Ishinomaki city, Kesennuma city and Fukushima Daiichi Nuclear Power Station have been successfully solved by our system. K computer of RIKEN, FX10 of the University of Tokyo, FX100 and CX400 of Nagoya University have been used in our system. For example, in case of Kesennuma city, the tsunami run-up analysis with a large ship of 130 million particles (1m particle spacing) and 1800 seconds has been done by K computer 12000 nodes and total wall-clock time 30 hours.

- Ishinomaki city
  - First stage analysis by 2D shallow water analysis
  - Second stage analysis by MPS
    - Tsunami run-up analysis
      - K computer 12000 nodes
      - Total wall-clock time: 30 days
      - 130 million particles (1m)
      - Time: 1800 s
  - Third stage analysis by MPS
    - Elastica for buildings by fluid pressure
      - Fluid analysis
        - FX10 6000 nodes
      - Total wall-clock time: 6 hours
      - 40 million particles (0.5m)
      - Time: 200 / 0.1 million step
      - Structure analysis
        - FX10 12000 nodes
        - Total wall-clock time: 2 hours
        - 10 million elements (2m)
        - Time: 200 / 400 steps

- Kesennuma city
  - First stage analysis by 2D shallow water analysis
  - Second stage analysis by MPS
    - Tsunami run-up analysis with a large ship of 60 m
      - K computer 12000 nodes
      - Total wall-clock time: 30 hours
      - 130 million particles (1m)
      - Time: 1800 s
  - Third stage analysis by MPS
    - Inundation analysis into Interior of Turbine Building of Fukushima Daiichi Nuclear Power Station Unit 1
      - FX100 360 nodes
      - Total wall-clock time: 72 hours
      - 130 million particles (0.1m)
      - Time: 200 s

- Fukushima Daiichi Nuclear Power Station
  - First stage analysis by 2D shallow water analysis
  - Second stage analysis by MPS
    - Tsunami run-up analysis
      - K computer 6000 nodes
      - Total wall-clock time: 40 hours
      - 200 million particles (1m)
      - Time: 1000 s
  - Third stage analysis by MPS
    - Inundation analysis into Interior of Turbine Building of Fukushima Daiichi Nuclear Power Station Unit 1
      - FX100 360 nodes
      - Total wall-clock time: 72 hours
      - 130 million particles (0.1m)
      - Time: 200 s

Since the beginning of inundation from the large object carry-in door,
1. The power panels is covered with water after a few second.
2. Inundation on B1 starts from the stair A after 20 s.
3. Emergency diesel generator B is covered with water after 80 s.