Tsunami run-up and inundation simulations using LexADV_EMPS solver framework on Fujitsu FX100

Masao Ogino (Nagoya University), Hongjie Zheng (Toyo University), Kohei Murotani (The University of Tokyo), Seiichi Koshizuka (The University of Tokyo), Ryuji Shioya (Toyo University), Lijun Liu (Nagoya University)

1. Development of the parallel expliciti MPS solver framework "LexADV_EMPS"

1.1 Objectives

We have been developing the open source software library, named **LexADV** [1], which is for the extreme-scale numerical simulations in continuum mechanics. Our aim in this research is to develop a scientific library for the particle method.

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.

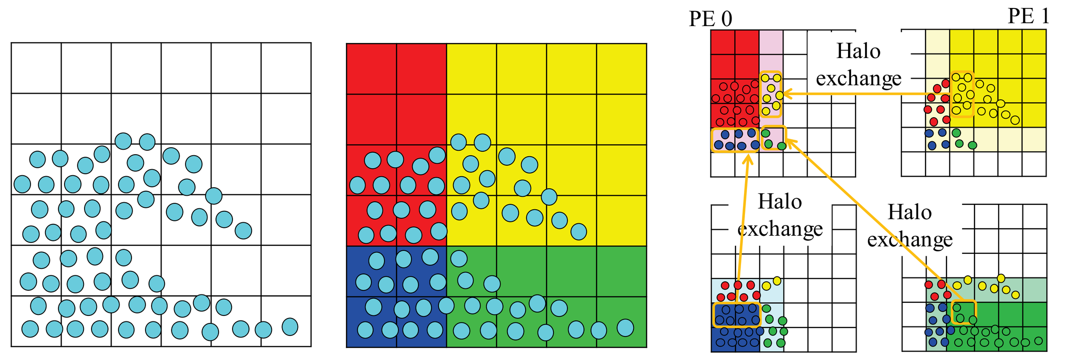
We adopted the moving particle simulation (MPS) method [2] which is one of the most popular particle method as target solver and have been developing the **LexADV_EMPS** of the explicit MPS solver framework which supports two-level domain decomposition, halo exchange pattern of communication, and dynamic load balancing on the distributed-memory parallel computers.

[1] LexADV website. http://adventure.sys.t.u-tokyo.ac.jp/lexadv/
[2] Koshizuka, S. and Oka, Y. *Nuclear Science and Engineering*, 123 (1996), pp.421-434.

1.2 Software overview

- Target problem size
  - 10 million - 1 billion particles or more than
- Functions for parallel computing
  - Two-level domain decomposition
  - Dynamic load balancing
  - Halo exchange pattern of communication
- Hierarchical bucket structure of three levels
  - Domain decomposition on 1st level bucket
  - Halo exchange on 2nd level bucket
  - Neighboring particle search on 3rd level bucket
- Development and operating environment
  - OS : UNIX, Linux
  - Compiler : C
  - Communication library : MPI

1.3 Bucket-based domain decomposition



(a) Embedding particles in buckets. (b) Domain decomposition for buckets by using the ParMETIS library. (c) Generation of halo exchange pattern.

Fig.1 Domain decomposition based on the bucket and generation of the halo exchange pattern.

1.4 Dynamic load balancing

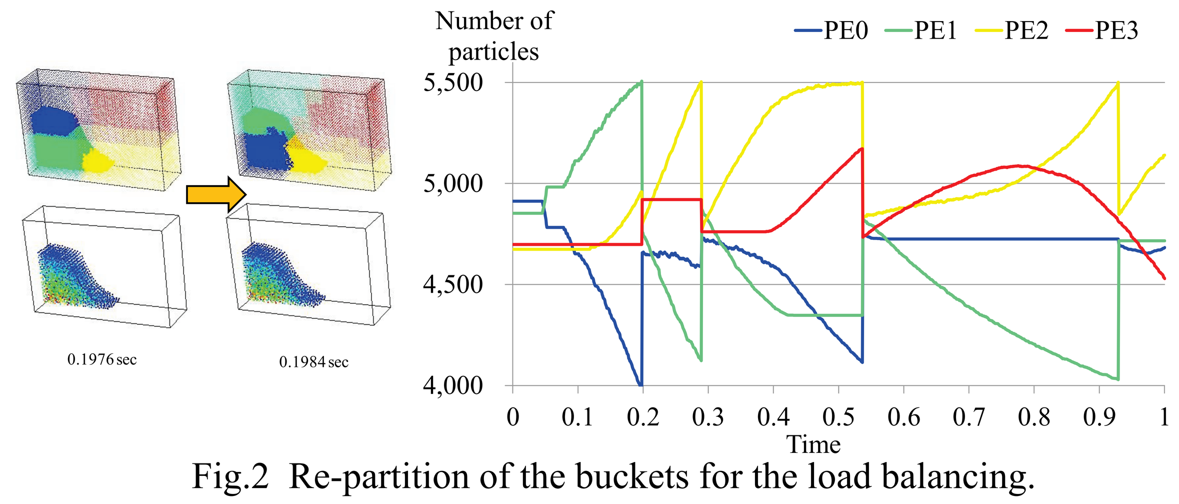


Fig.2 Re-partition of the buckets for the load balancing.

1.5 Basic performance measurement

Table 1 Computational performances of LexADV_EMPS and its kernel functions in a dam breaking problem with 130 million particles on 192 computer nodes of FX100.

	Eff. to peak FLOPS [%]	SIMD [%]	L1I miss [%]	L1D miss [%]
all	13.81	74.33	0.0010	2.7815
- calc. viscosity & gravity	18.32	74.72	0.0005	3.3223
- calc. collision	17.30	68.36	0.0004	2.8964
- calc. pressure	21.95	74.58	0.0009	1.7643
- calc. pressure gradient	19.30	75.61	0.0005	2.7371

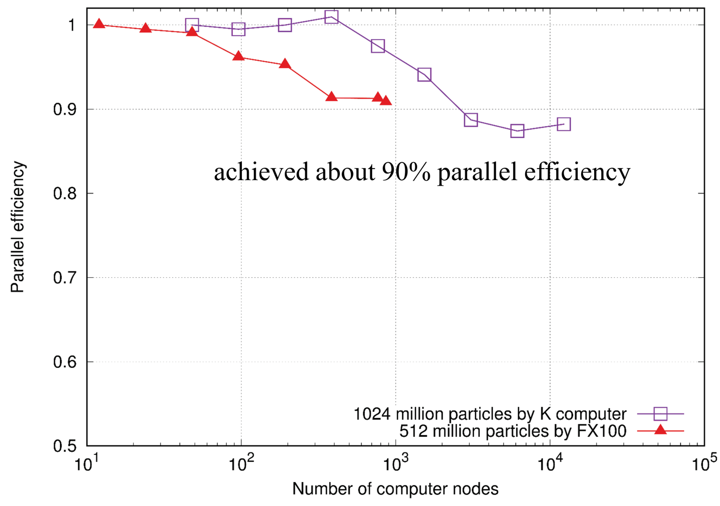
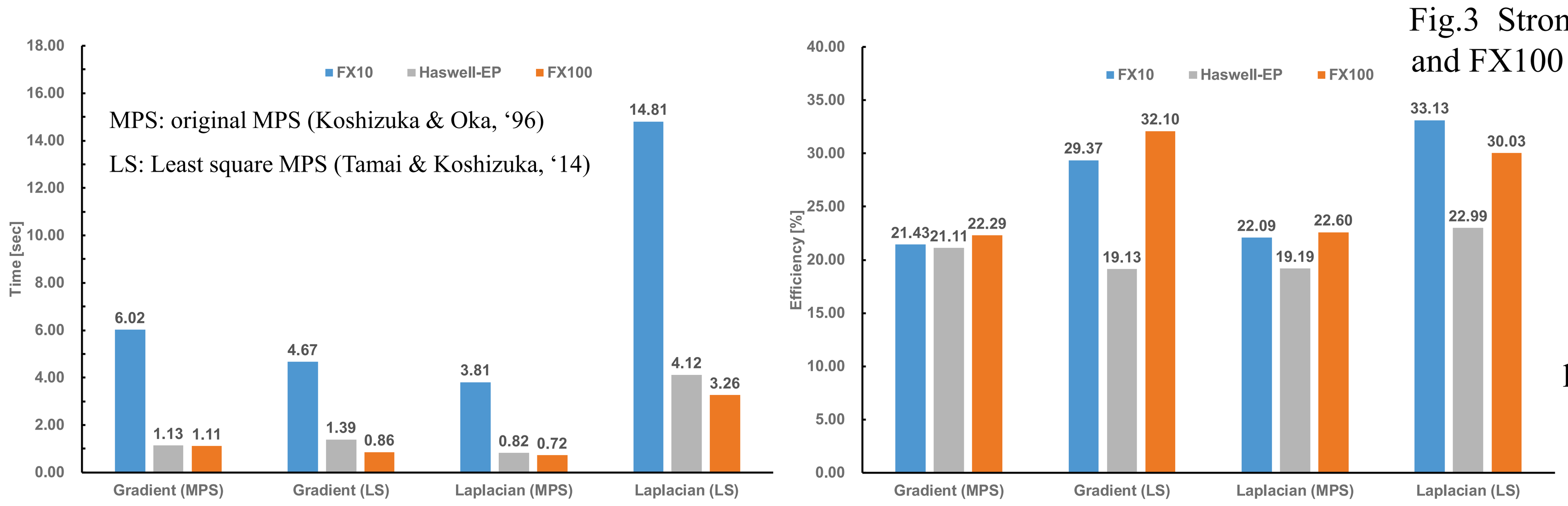


Fig.3 Strong scaling efficiency by K computer and FX100 for a hydrostatic pressure problem.



(a) Computation time (b) Efficiency (% of peak performance)

Fig.4 Computational performances with different spatial derivative schemes in a hydrostatic pressure problem with 16 million particles on 12 computer nodes of FX10 (2.84 TFLOPS), Haswell-EP (13.98 TFLOPS), and FX100 (13.52 TFLOPS).

1.6 Example of implementation of distributed-memory parallel explicit MPS solver by LexADV_EMPS

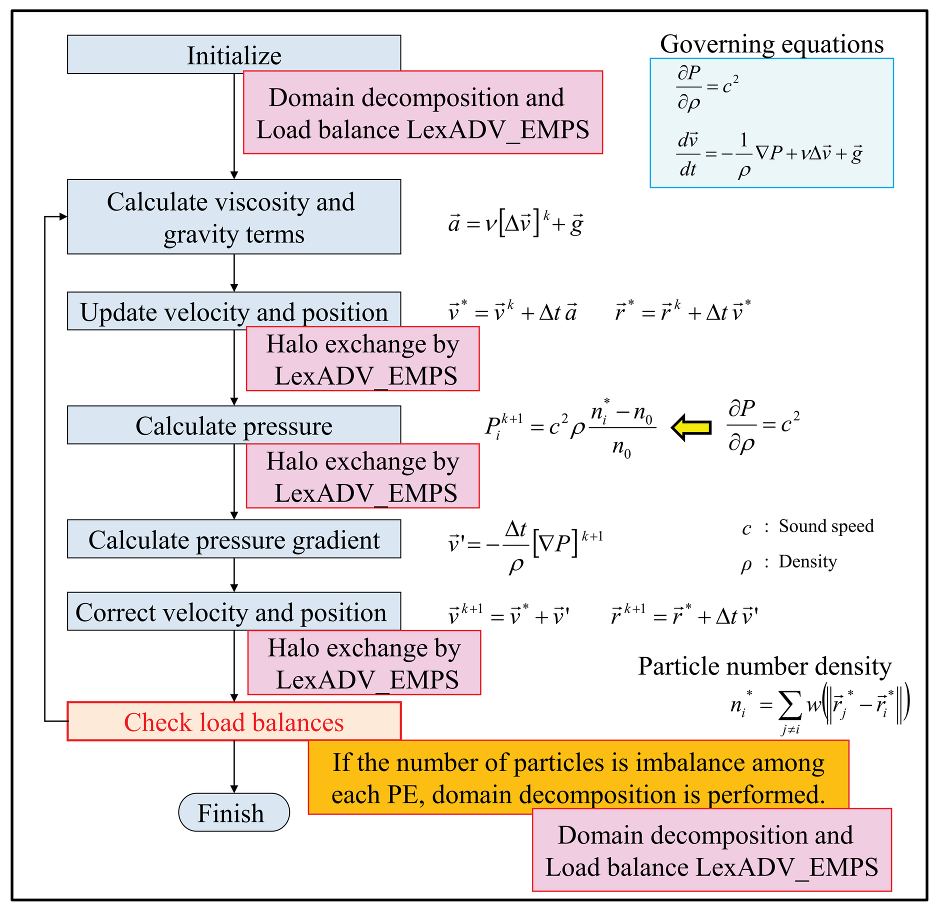


Fig.5 Flowchart of explicit MPS method.

1) 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.

2) Communication for many rigid bodies

- Communication for many rigid bodies occurs in only summation operation. (most important point)
- Summations are done three times 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 is (9 double type components) x (# of rigid bodies) . (not # of rigid body particles.)
- For example, if 100 rigid bodies, communication size is only 8(size of double)x9x100 byte.

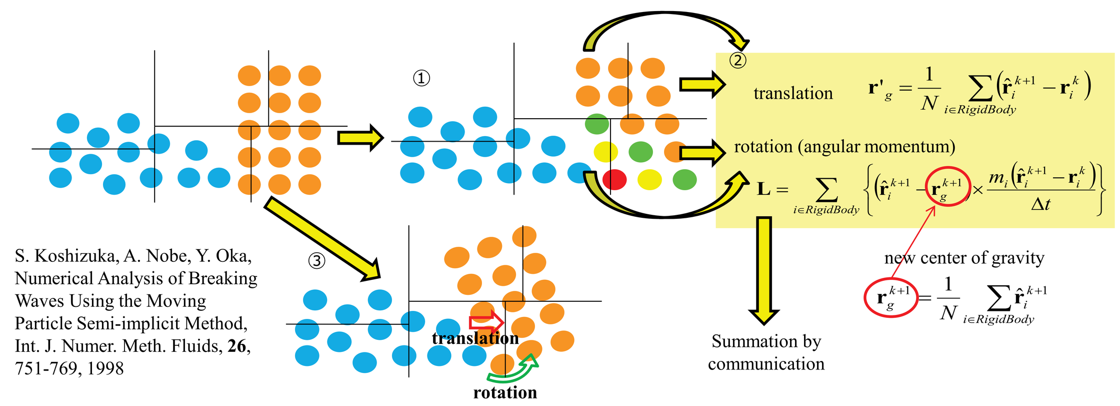


Fig.6 Computational procedure of the rigid body.

2. Tsunami run-up and inundation simulations

2.1 Overview

Tohoku area was severely damaged by the tsunami of the Great East Japan Earthquake on 2011. To simulate tsunami effects on coastal area, we analyzed tsunami inundation area of Kesennuma city, Ishinomaki city, and Fukushima Daiichi Nuclear Power Staion. In this research, to solve target area efficiently, we use results of the 2-dimensional shallow-water analysis for the area of about 1000 km^2 as boundary conditions of coastal area.

In case of Kesennuma city, tsunami run-up analysis of 130 million particles (1-meter particle spacing) has been done in about 6 days using 192 nodes of FX100. On another case, tsunami run-up simulation of Fukushima Daiichi Nuclear Power Station with 250 million particles has been successfully performed in about 150 hours using 192 nodes of FX100. Since solving same problem on K computer used computer resource of 12,000 nodes multiply 30 hours, with consideration of the difference of peak FLOPS, we succeeded in reduction of required computer resource drastically by using LexADV_EMPS on FX100.

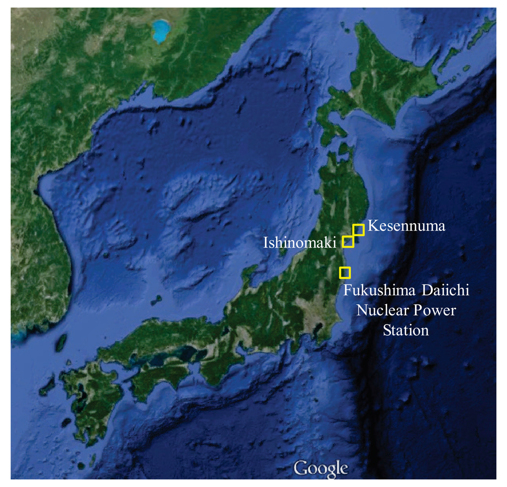


Fig.7 Target areas

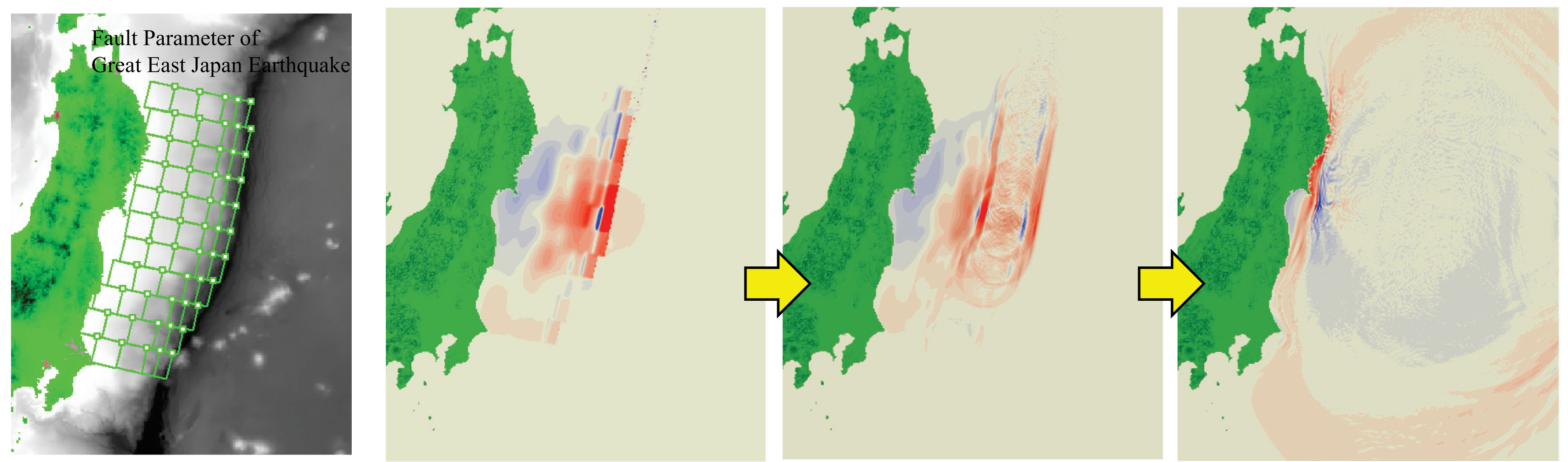
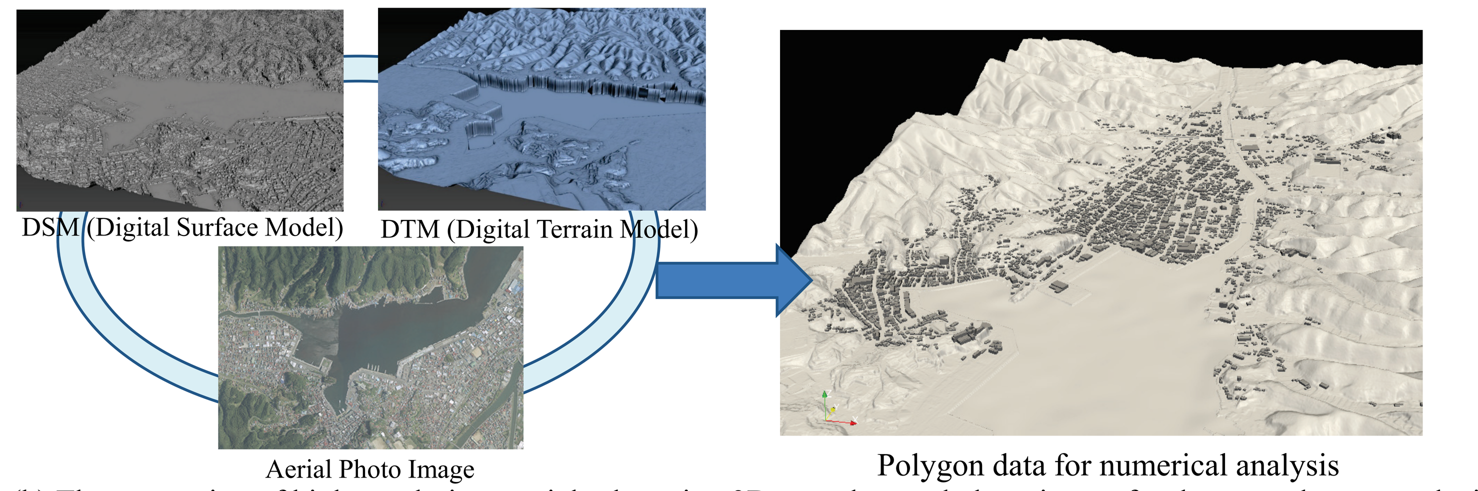


Fig.8 First stage analysis by using the 2D shallow-water analysis from epicenter to coast areas.

2.2 Kesennuma city

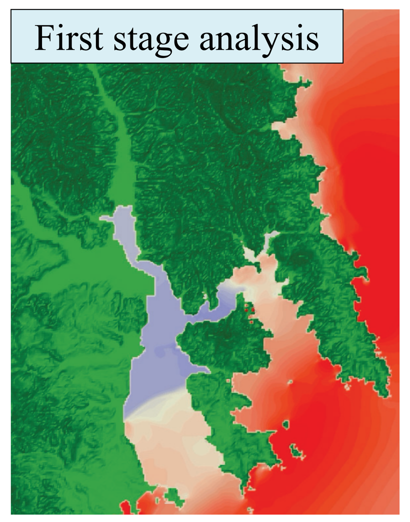
(a) Kesennuma city.



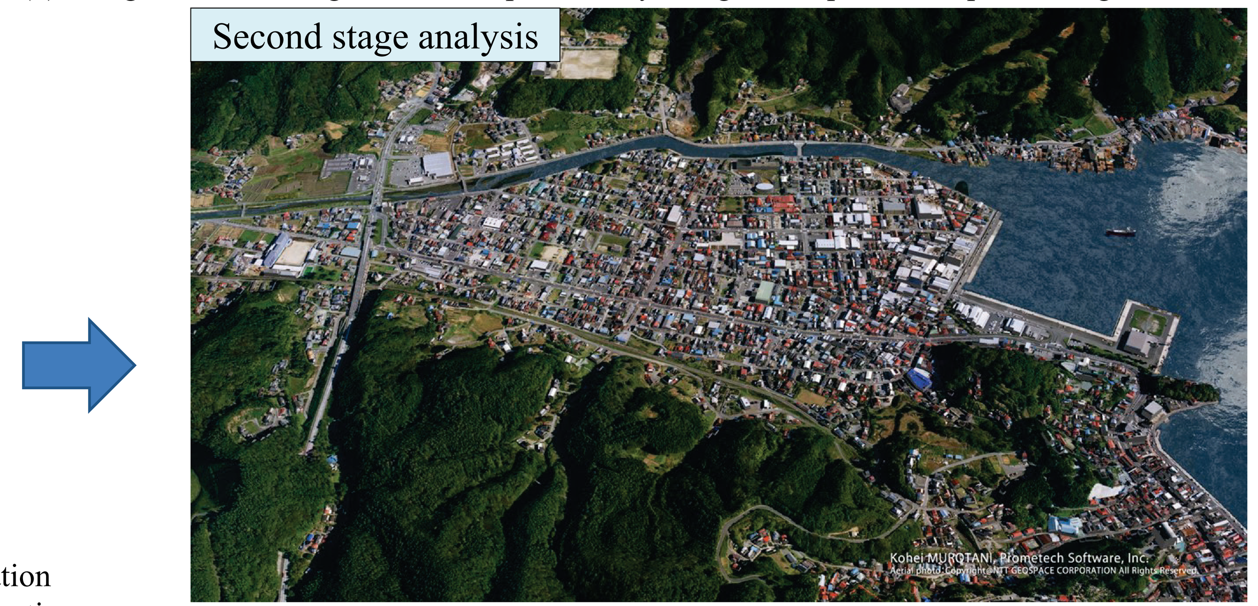
(b) The generation of high-resolution particles by using 3D map data and photo image for the second stage analysis.

Table 2 Computational performances of LexADV_EMPS at the second stage analysis. The number of particles is about 130 million, and the analysis time is 1,800 seconds.

System	FX100, Nagoya Univ.
Nodes	192
Elapsed time [day]	6

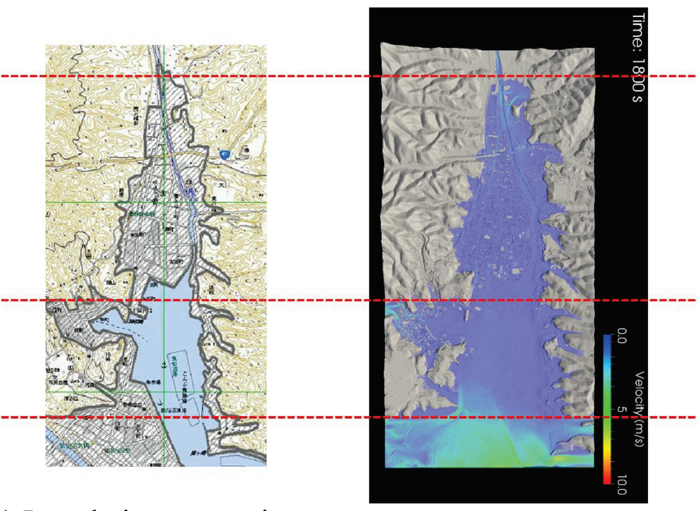


(c) Results of the tsunami wave propagation analysis by using the 2D shallow-water equation.



(d) Photo-realistic visualization of tsunami running up analysis with a 330 ton fishing vessel.

Fig.9 Tsunami running-up and inundated simulation for Kesennuma city.

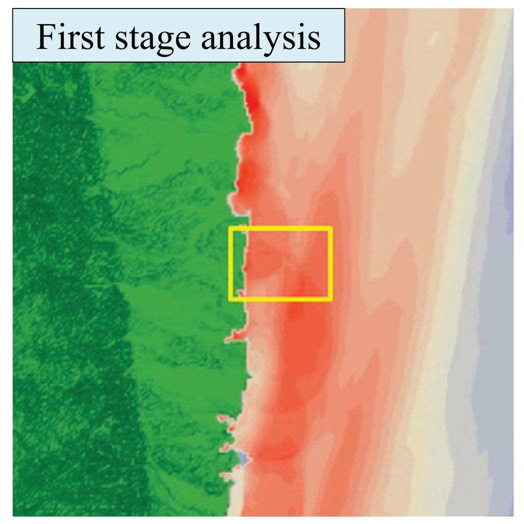


(a) Inundation area using aerial photo (b) Computer simulation by LexADV_EMPS

Fig.10 Validation of inundation map between aerial photo and computer simulation results.

2.3 Fukushima Daiichi Nuclear Power Station

(a) Fukushima Daiichi Nuclear Power Station.



(b) Results of the tsunami wave propagation analysis by using the 2D shallow-water equation.

Table 3 Computational performances of LexADV_EMPS at the second stage analysis. The number of particles is about 250 million, and the analysis time is 1,800 seconds.

System	K computer, RIKEN	FX100, Nagoya Univ.
Nodes (peak TFLOPS)	12,000 (1,536)	192 (216)
Elapsed time [h]	30	150
Computer resource usage [node x hours]	360,000	28,800

92% reduction

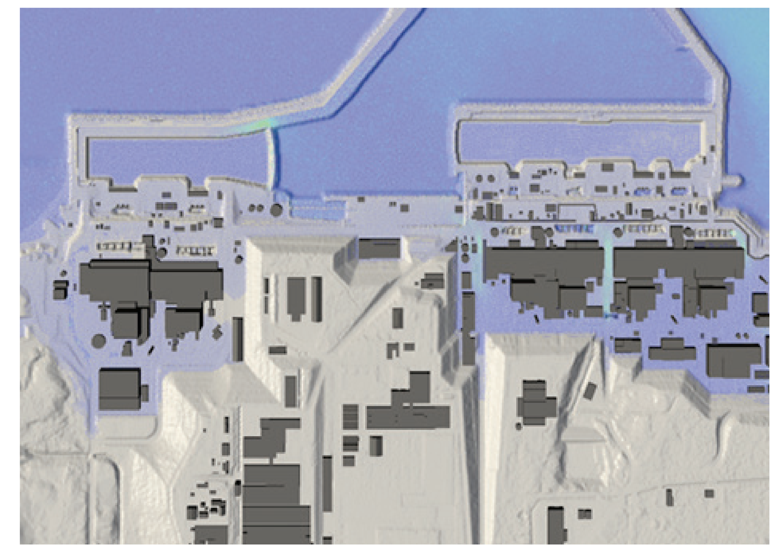
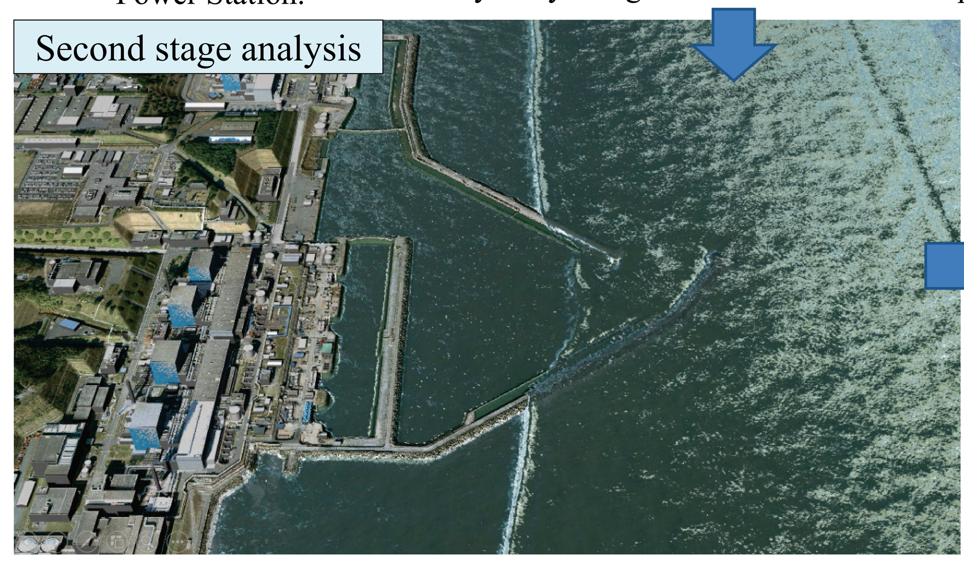
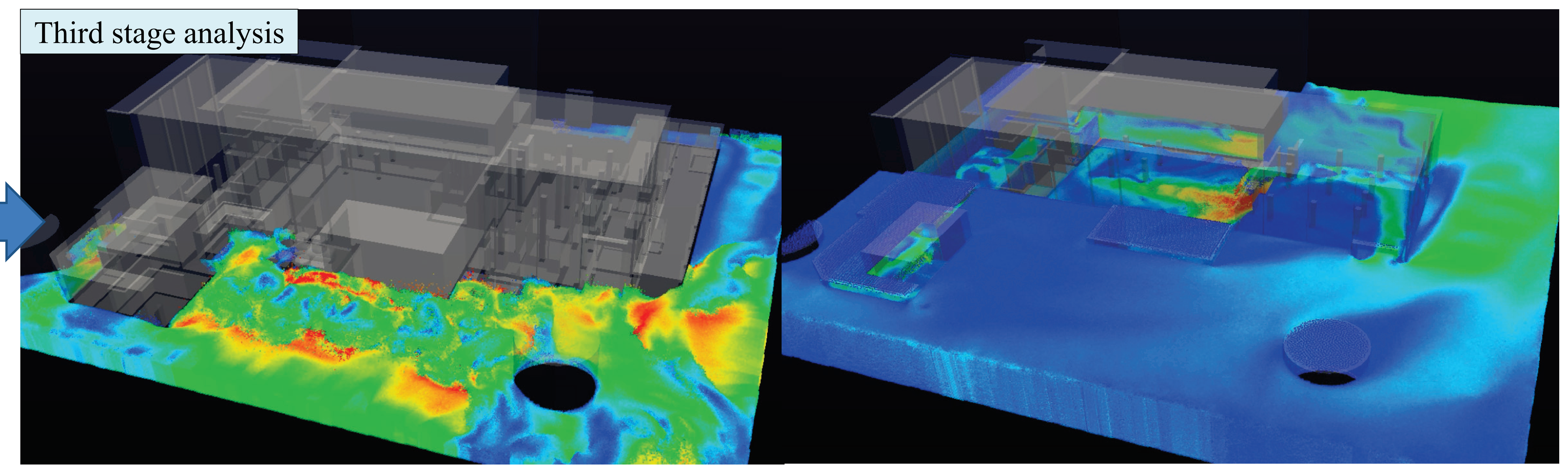


Fig.12 Inundation area by using the LexADV_EMPS.



(c) Photo-realistic visualization of tsunami running up analysis by using the explicit MPS method.



(d) Results of tsunami inundated analysis for the interior of the turbine building by using the explicit MPS method.

Fig.11 Tsunami running-up and inundated simulation for Fukushima Daiichi Nuclear Power Station.