Simulation of tsunami propagation

Xing Cai
(Joint with G. Pedersen, S. Glimsdal, F. Løvholt, H. P. Langtangen, C. Harbitz)

Simula Research Laboratory

Dept. of Informatics, University of Oslo

2nd eScience meeting
Geilo
January 21–22, 2008

Outline

1. Introduction
2. Challenges
3. Strategy

List of Topics

1 Introduction

2 Challenges

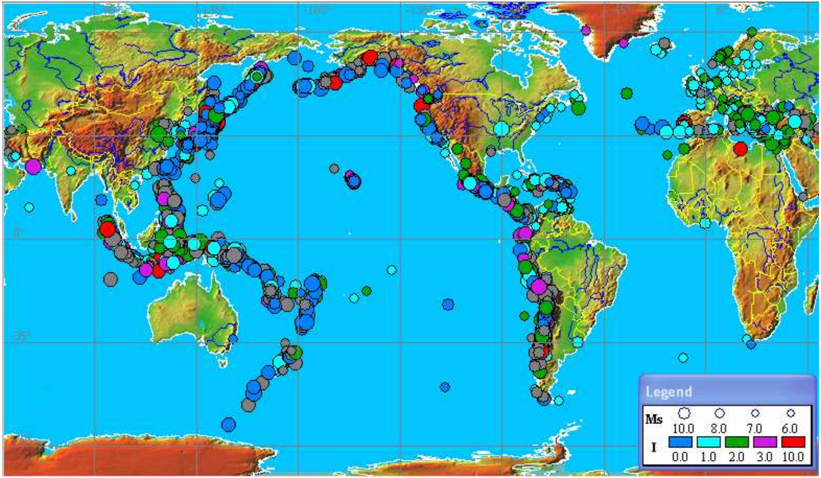
3 Strategy

About tsunami

Tsunamis: large waves formed by rapid mass movements

- Induced by subwater earthquake, volcanic eruption
- Induced by asteroid impact
- Induced by landslide/rockslide (of great importance to the Norwegian fjords)

Historical tsunamis



Total 1965 tsunami events from 1628BC to 2004
Statistics and figure by Tsunami Laboratory, Novosibirsk, Russia

Indian Ocean Tsunami Dec. 2004

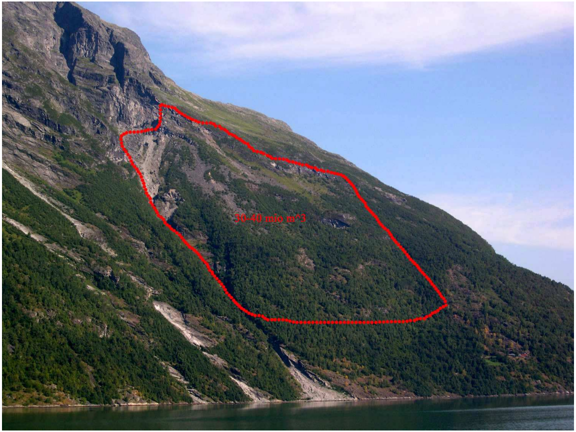
More than 225,000 people died

Tafjord, Norway, 1934

Rockslide induced tsunami, 62 meter runup, 40 people dead

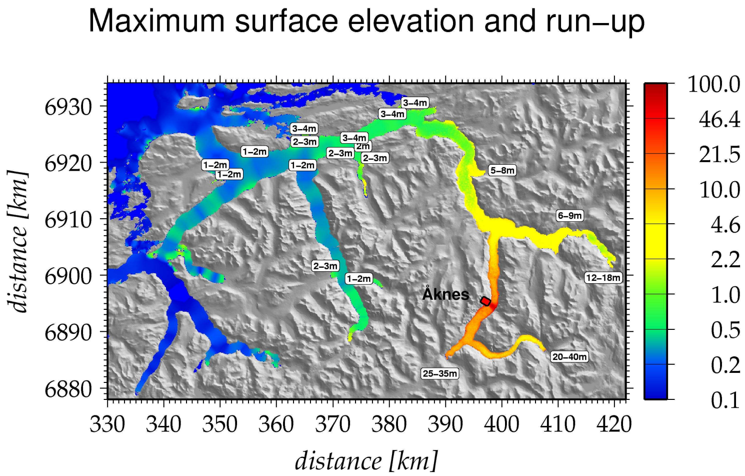
More than 170 tsunami-related deaths in Norway in last century

Tsunami danger at Åknes



- Potential rock avalanche
- Different scenarios need to be simulated $\Rightarrow$ lots of computations

Preliminary Åknes simulation



Courtesy of S. Glimsdal NGI/ICG, the "Åknes-Tafjord project"

Three phases of tsunami

- Tsunami generation
  - modeling is complex
  - in connection with geological source modeling
- Tsunami propagation
  - long distance
  - huge area
- Costal impact
  - wave amplification, breaking, runup and inundation on shore
  - different physics

Our objective: high-resolution, high-accuracy and efficient simulation of tsunamis

List of Topics

1. Introduction
2. Challenges
3. Strategy

Challenge 1: mathematical modeling

- Different physics
  - dispersion
  - wave breaking
  - rotational effects
  - nonlinear effects
  - wave runup and innudation
  - sedimentation
  - turbulence
- Different mathematical models
  - Navier-Stokes equations
  - potential-flow model
  - Boussinesq wave equations
  - shallow water long wave equations

Challenge 2: numerical aspects

- A wealth of discretization strategies:
  - finite difference
  - finite element
  - finite volume
  - spectral methods
- Stability
- Moving computational boundaries
- Nonlinearity
- Fast solution of linear systems
- ...

Challenge 3: computational amount

- Huge solution domain
  - for example an entire ocean
  - (ambition: the whole globe)
- Very high resolution is needed regionwise
  - near shore zones
  - source modeling
  - sharp local topographical changes
- Sufficiently high resolution is needed elsewhere
- High temporal resolution

Challenge 4: software

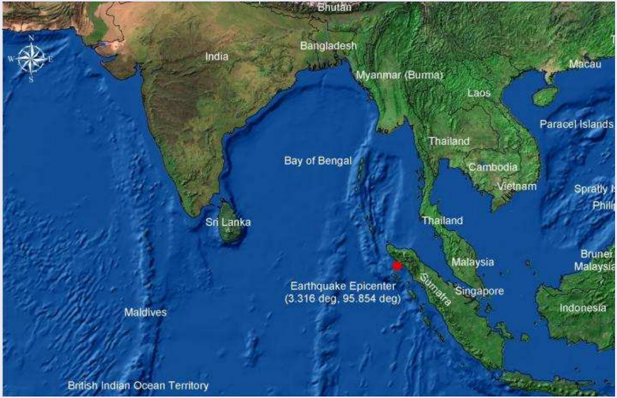
- A wealth of existing software codes
  - public domain
  - commercial
  - in-house
- Each code is targeting a particular mathematical model
- Each code is bound with a particular numerical strategy
- Each code has advantages and disadvantages
- Adaptive meshing and time stepping not yet common

There exists no software code good enough for our objectives

Challenge 5: high-performance computing

- Huge amount of computation
- Short turnaround time
- Parallel computing is essential
  - must incorporate adaptivity at different levels
  - must preferably make use of existing (serial) codes
  - must efficiently use modern parallel computers

Computational resolutions (example: Indian Ocean)



- 1km$\times$1km resolution overall: about 40×10^6 mesh points
- 200m$\times$200m resolution overall: 10^9 mesh points

Computational resolutions (cont´d)

- 1km resolution ok for deep water, insufficient everywhere
- 200m resolution $\longrightarrow$ too large meshes, and still too coarse near shore
- In the Malacca Strait, e.g., up to 10m resolution necessary
- We need "smart computing":
  - High resolution only in areas where necessary
  - Simple mathematical model in vast areas
  - Advanced mathematical model (due to complicated physics) in small areas
- Desirable resolution requires
  - number of mesh points $\sim 100 \times 10^6$
  - number of time steps $\sim$ many thousands

List of Topics

1. Introduction
2. Challenges
3. Strategy

Overview

- Wave propagation simulation is essential to tsunami studies
- Need to seamlessly couple with coastal impact simulation
- Build parallel simulators from re-using serial codes
- Plug and play
- Objective: high-resolution, high-accuracy and efficient simulations
- Vision: real-time simulation and assessment of transoceanic tsunamis

Strategy: adaptivity at different levels

- Adaptivity in mathematical models
  - advanced models in "demanding regions"
  - simpler models elsewhere
- Adaptivity in numerical strategies
  - finite elements and unstructured meshes in "demanding regions"
  - simpler and more efficient methods elsewhere
- Adaptivity in resolution
  - very high resolution in "demanding regions"
  - sufficiently high resolution elsewhere
- Adaptivity in software
  - sophisticated software in "demanding regions"
  - simpler software elsewhere

Purpose: economic high-performance parallel computing

Plug and play

- Not certain which mathematical model is best suited in a particular region
- Not certain which numerical strategy is best suited in a particular region
- Not certain which software code works best in a particular region

We want the flexibility of "plug and play"

Subdomain-based parallelization

- The entire solution domain is decomposed into subdomains
- Each subdomain can choose between
  - different mathematical models
  - different numerical methods
  - different mesh types and resolutions
  - different serial software codes
- Parallelism arises from concurrent computations on the subdomains
- Mathematically inspired by the additive Schwarz algorithm

The numerical foundation

Additive Schwarz algorithm

- Small amount of overlap between subdomains
- Simple algorithmic structure
- Originally as a parallel numerical strategy for solving large linear systems
- We apply domain decomposition at "software level"
- Subdomains are more independent
- Many components of additive Schwarz are generic and can be implemented as library

Programming effort

- Wrap up each existing serial code with a unified generic interface of all subdomain solvers
- Write a relatively simple main program coordinating all the subdomain solvers
- Use library for tasks such as domain partitioning and inter-subdomain communication
- Users are not directly exposed with low-level parallel programming details
- Limited user programming effort due to extensive code re-use
    - generic library of additive Schwarz components
    - re-use of existing serial codes

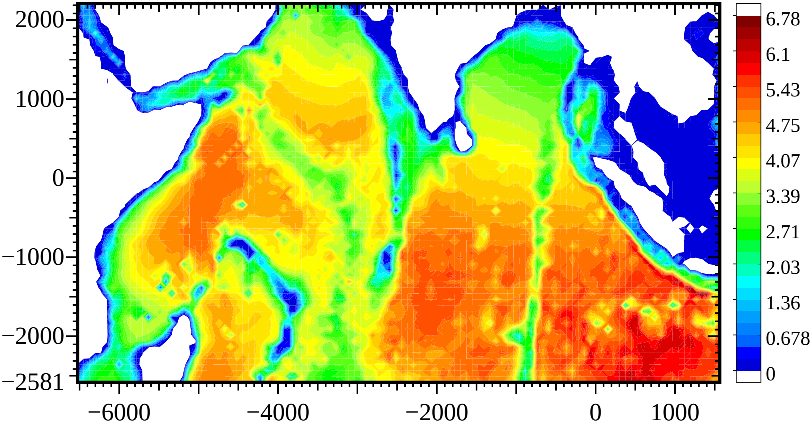
Proof of concept

Parallel tsunami simulation using two serial codes together

- Starting point
  - C++ Boussinesq solver using FEM with adaptivity
  - Legacy F77 code using FDM
  - Direct parallelization of either code requires too much work
- High-level parallelization
  - Easy programming using the generic Schwarz framework

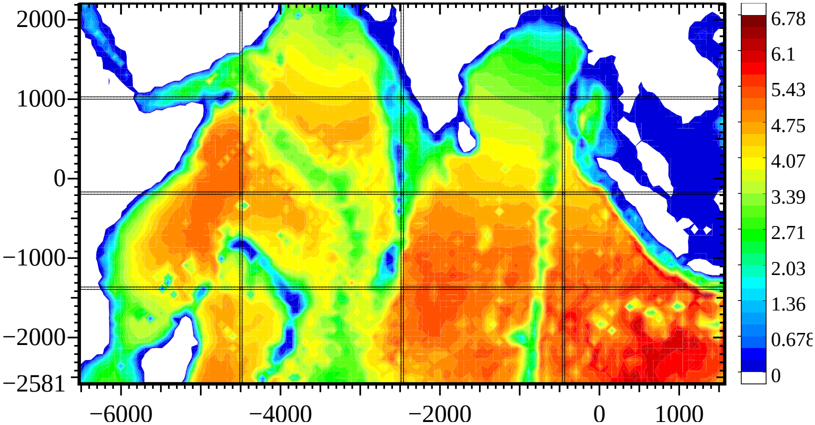
Result: hybrid parallel tsunami simulator

Simulation of Indian Ocean Tsunami



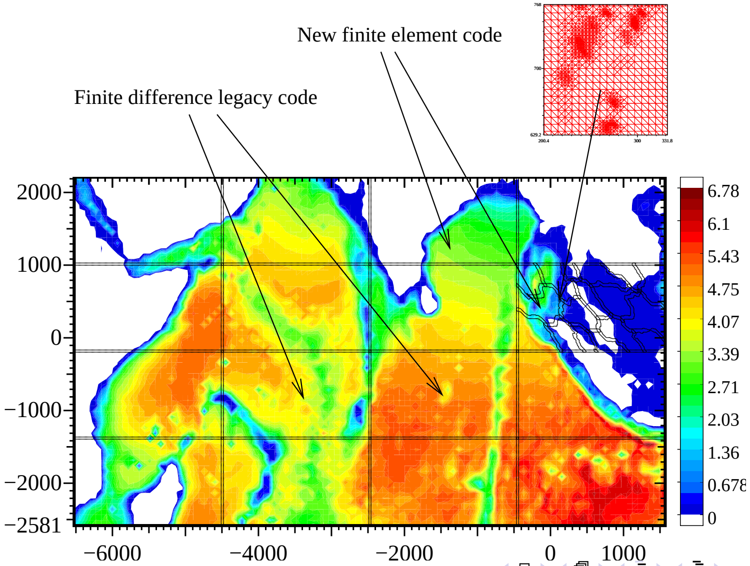
Bathymetry

Subdomain preparation (I)

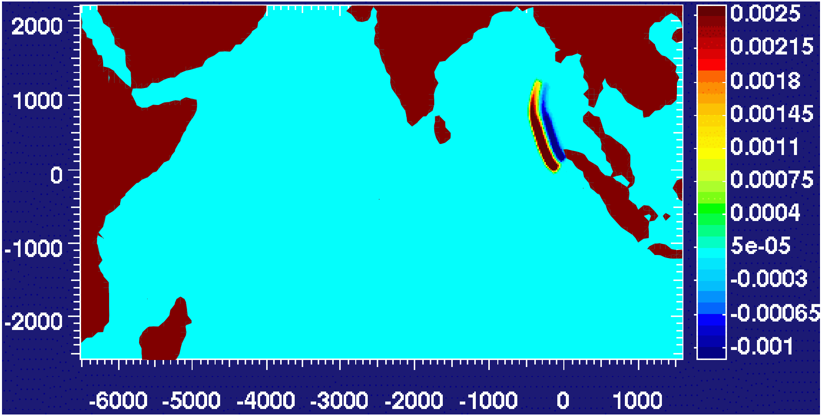


Uniform FDM meshes and regular domain partitioning

Subdomain preparation (II)

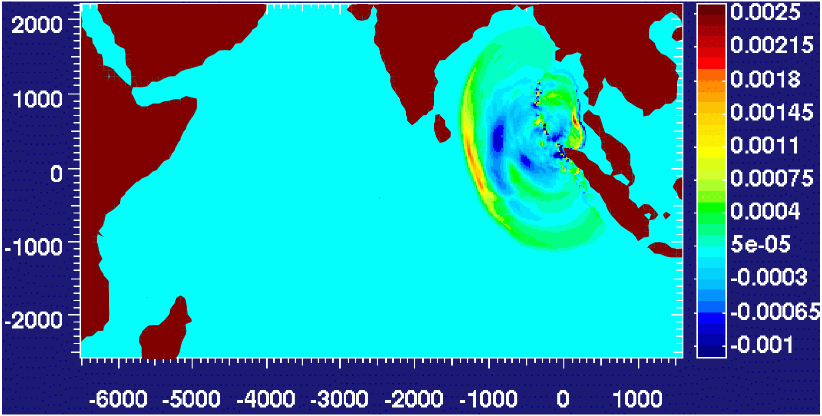


Preliminary test simulation



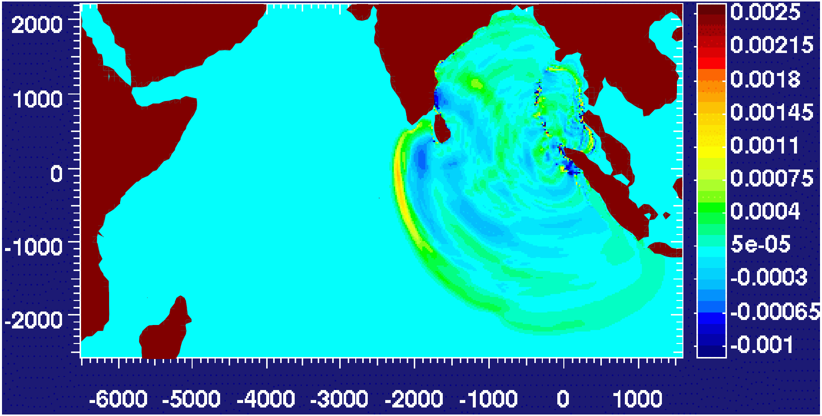
Initial wave elevation after the earthquake

Snapshot 1



After 1.4 hours

Snapshot 2



After 2.8 hours

Concluding remarks

- Economic high-performance computing due to adaptivity at different levels
- Subdomain-based parallelization using a generic framework
- Numerical foundation in additive Schwarz algorithm
- Extensive re-use of existing serial codes
- Possibility of plug-and-play