

地球シミュレータを活用した即時津波浸水予測のためのデータベースの構築

課題責任者

馬場 俊孝 徳島大学 工学部

著者

馬場 俊孝 徳島大学 工学部

三神 厚 徳島大学 ソシオテクノサイエンス研究部

坂東 淳 徳島県危機管理部

地方自治体の地震や津波などの災害対応においては、まず対応の基礎となる「被災情報の確保」が大きな課題である。特に大規模災害であればあるほど、断片的な情報から全体を推計することは困難を極める。このため、徳島大学と徳島県は、防災関係機関の対応判断の支援を目的として、津波浸水シミュレーションによる「面」的な災害推定システムを開発している。本システムでは気象庁から提供される震源位置とマグニチュードに基づいて、津波浸水データベースを即時検索することによって行われる。アルゴリズムは単純だが、南海トラフで発生する可能性がある津波を網羅し、かつ高分解能な津波浸水データベースを構築する必要があり、その実現には膨大な計算資源が必要となる。本研究では、津波浸水データベースの構築について、地球シミュレータを用いて検討した。その結果、津波計算コードの地球シミュレータへの最適化を経て、実行可能な計算時間でデータベースを構築できることが分かった。

キーワード：津波, 南海トラフ, 予測システム

1. はじめに

南海トラフ巨大地震などによる大規模地震・津波災害の場合、人命救助のために、災害応急体制を迅速かつ適切に構築し、対応を開始しなければならない。このため、地震動による被害や津波の浸水範囲などの情報を可能な限り早く把握することが求められる。半日から1日程度経過すれば、ヘリコプターなどによる調査によって被害の全貌が明らかになってくるものの、災害発生後およそ半日間は断片的な被害情報しか得られない。

このように災害対応業務においては、まず対応の基礎となる「被災情報の確保」が大きな課題である。特に大規模災害であればあるほど、断片的な情報から全体を推計することは困難である。このため、従来「点」で確認していた観測情報だけでなく、シミュレーションによる「面」的な災害情報の推定が求められる。徳島大学は徳島県と連携して、津波災害発生時において、上記を実現するシステムの開発している。このシステムでは震源情報（位置と規模）に基づいて、データベース検索し、津波浸水の予測を提供し、さらにはそれに基づいて、通行可能道路を判断する。そのシステムで必要となる津波浸水データベース（DB）の構築を、今回、地球シミュレータにより試作した。

2. 津波シミュレーションの手法

まず、本研究で利用した津波シミュレーションモデルについて説明する。地震は地殻変動を伴い、地盤の水平移動や昇降が発生する。これが海底において生じた場合、海底の昇降によって津波が励起される。震源の情報を元

に作成した波源を初期条件とし、地形情報を境界条件として入力して、水位と流量の時間発展を評価することで津波伝播を計算する。本計算では、津波シミュレーションコード（JAGURS）（Baba et al, 2015; 2016[2][3]）を利用した。JAGURSは、オイラーの運動の式と連続の式を長波として近似した式（長波方程式）を使用し、その時間発展を有限差分法により解くことで、津波の波源から沿岸までの伝搬、および陸地への浸水や河川への遡上を計算することができる。移流項（空間の移動に起因する速度の変化量）および海底面の摩擦を考慮した非線形式、またはこれらを考慮しない線形式の2つに対応しており、また太平洋を伝搬するような津波計算も扱えるように座標系は球面座標を採用している。JAGURSで用いている支配方程式は次のように書ける。

$$\begin{aligned} \frac{\partial M}{\partial t} + \frac{1}{R \sin \theta} \frac{\partial}{\partial \varphi} \left(\frac{M^2}{d+h} \right) + \frac{1}{R} \frac{\partial}{\partial \theta} \left(\frac{MN}{d+h} \right) \\ = - \frac{g(d+h)}{R \sin \theta} \frac{\partial h}{\partial \varphi} - \frac{gn^2}{(d+h)^{7/3}} M \sqrt{M^2 + N^2} \end{aligned} \quad (1)$$

$$\begin{aligned} \frac{\partial N}{\partial t} + \frac{1}{R \sin \theta} \frac{\partial}{\partial \varphi} \left(\frac{MN}{d+h} \right) + \frac{1}{R} \frac{\partial}{\partial \theta} \left(\frac{N^2}{d+h} \right) \\ = - \frac{g(d+h)}{R} \frac{\partial h}{\partial \theta} - \frac{gn^2}{(d+h)^{7/3}} N \sqrt{M^2 + N^2} \end{aligned} \quad (2)$$

$$\frac{\partial h}{\partial t} = - \frac{1}{R \sin \theta} \left[\left(\frac{\partial M}{\partial \varphi} + \frac{\partial (N \sin \theta)}{\partial \theta} \right) \right] \quad (3)$$

式(1)、(2)が運動の式、式(3)が連続の式である。ここでM,Nは緯度、経度方向の単位幅流量、Rは地球半径、

θ は余緯度、 ϕ は経度、 d は静水深、 h は水位、 f はコリオリ係数、 t は時間を示す。また、長距離にわたって伝播する遠地津波の場合には、分散項やコリオリ力を考慮する必要があるが、本研究では近地地震を対象としているため、それらは考慮しない。

差分計算のための格子として、スタaggerド格子を採用し、水位 (h) および静水深 (d) を格子中央、流量 (M, N) を $1/2$ 格子分ずらし格子境界上にとっている。基礎方程式はリーブフログ法のスキームを用いて差分化する。数値不安定を避けるため移流項のみ1次の風上差分で解いている。1次風上差分では、数値粘性の影響により変数値(ここでは流量と水位)の減衰が想定されるが、津波の波長は格子分解能に比べて非常に長く、なだらかに変化するため、数値粘性の強い1次風上でも、検潮記録の再現や浸水域の予測においては問題にならない。しかしながら、浅水変形、海岸での反射や地形による屈折、沿岸近傍で短波長成分が卓越してきた場合において、1次風上による数値粘性に影響が無視できなくなるため、計算目的によっては、数値粘性の少ない高次の差分スキームを利用する必要があるので注意しなければならない。

JAGURS は多階層のネスティングにより、計算領域内の特定の領域について格子解像度を上げた計算が可能となっている。ネスティング境界において水位および流量が非連続とならないように双方向の通信を行っており、親ドメインから子ドメインに対しては親ドメインのデータをネスティング境界で内挿して生成し、子ドメインに渡す。子ドメインから親ドメインに対しては、親ドメインと重なっている子ドメインの格子のデータをコピーする。MPI 並列化については、ネスティング階層それぞれに対して、全プロセスでの東西、南北方向2次元領域分割を行う。ネスティング階層内の分割領域間の通信は1対1の袖通信、またネスティング階層間の通信は全プロセスによる集団通信を利用する。JAGURS における並列津波計算のフローチャートを図1に示す。

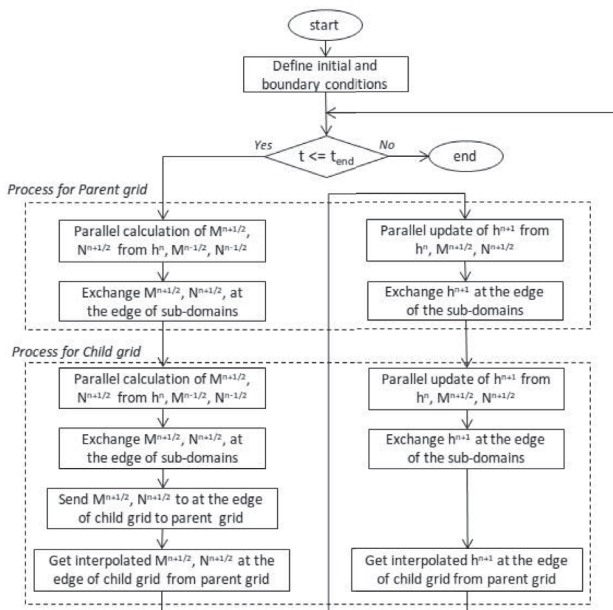


図1 津波計算のフローチャート(2重ネスト)の例。

3. 地球シミュレータへの最適化

津波シミュレーションコード(JAGURS)は、スーパーコンピュータ「京」及びIntel CPUを搭載したPCクラスタでの動作実績はあったが、当初、ESでの動作実績はなく、ES向けにチューニングが施されていない。ここでは、性能向上の効果が見られた2つのチューニング(本稿ではtune1、tune2と記す)とマルチジョブコントロール機能について報告する。実計算で使用する南海トラフ地震を想定したデータセットで津波計算を実施する。並列手法はFlatMPIとし、ノード内4プロセスで測定を実施した。地形データ等の領域分割は、大部分の処理で東西方向のループが最内ループとなっているため、東西方向の分割数を可能な限り小さくし、ループ長(ベクトル長)をより長くすることにより、ベクトル化の効率を高める必要がある。そこで東西方向に2分割、南北方向に32分割とし、64MPIで測定した。

3.1 非線形流量計算ルーチンのベクトル化(tune1)

計算の大部分は非線形流量計算ルーチンが占めている。この非線形流量計算ルーチンについて、主要演算ループがベクトル化されていなかったため、ループ分割を適用しベクトル化を実施した。当該ルーチンはループボディが大きく、条件分岐が複数あり、ベクトル化を阻害していた。表1の条件1を満たすケースが大部分を占めているため、条件1の条件分岐を削除し、全条件に対して、処理1のベクトル演算処理を行った後、その他条件に適合する場合のみ変数を更新するように変更を加えた。その結果、非線形流量計算ルーチンの実行時間が大幅に短縮されており、それに伴って本ルーチンにおけるインバランスを含む子ドメインから親ドメインへのネスト間通信、及び袖領域の隣接通信の実行時間も短縮された。全体として、平均ベクトル演算率が75.9%から97.9%に、平均ADB要素ヒット率が36.4%から67.8%に向上し、計算時間が6.1倍高速化した。

表1 非線形流量計算ルーチンの最適化

最適化前	最適化後
<pre> do i = ist, ind do j = jst, jnd if (条件1) then 処理1 else if (条件2) then 処理2 else if (条件3) then 処理3 else 処理4 end if end do </pre>	<pre> do i = ist, ind do j = jst, jnd 処理1 end do do i = ist, ind do j = jst, jnd if (条件2) then 処理2 end do : : </pre>

3.2 ネスト間通信手法の変更 (tune2)

津波シミュレーションコード (JAGURS) におけるネスト間通信の手法としては、MPI_Allreduce を用いる手法と MPI_Alltoallv を用いる手法が選択できる (図2)。MPI_Allreduce を用いる場合は、全プロセスがネスト間通信の対象となるバッファを一様に保持する。このバッファの全域を0で初期化した後、各プロセスが自身の送信する要素のみについて値を入れ、その後 MPI_Allreduce 通信を行うことで全プロセスが結果を共有する。この手法は、並列数に拠らず通信バッファサイズが常に一定であり、各プロセスがアクセスする必要がない要素についても共有のための通信が行われるため、通信量に無駄が多いというデメリットがある。一方、MPI_Alltoallv を用いる場合は、各プロセスが、自身が送受信すべき最低限の要素について送受信を行う。この手法では、並列数の増加に伴って各プロセスの通信量が減少し、必要最低限の通信が行われるというメリットがある。

また、ネスト間通信における子ドメインから親ドメインの内挿処理については、「親ドメインの領域端のみを対象とする」か「親ドメインの全領域を対象とするか」を選択できる。後者の場合、前者と比較して MPI 通信量が増加する。本研究では、親ドメインの領域端のみを内挿対象とした。

スーパーコンピュータ「京」における高並列計算では、MPI_Alltoallv の3次元論理トーラス形状に特化した手法 (安藤ほか、2014[1]) が効果的だったが、新 ES で行う本研究では MPI_Alltoallv よりも MPI_Allreduce の方が計算時間で1.8倍高速であった。これは並列規模が小さく、通信量が少ないため、MPI_Allreduce の方が高速となったと考えられる。

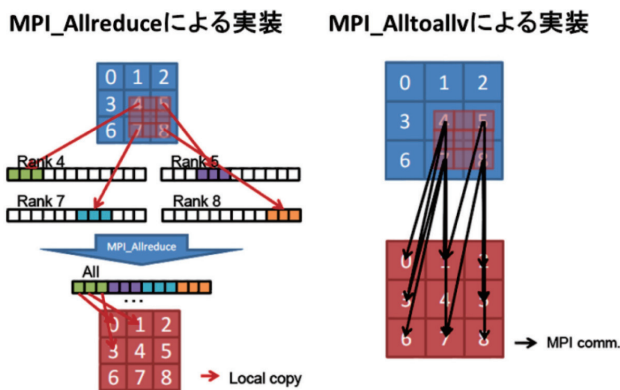


図2 MPI_Allreduce と MPI_Alltoallv の比較。

3.3 マルチジョブコントロール

津波シミュレーションコード (JAGURS) には入力データが異なる複数ケース (複数のシミュレーション) を単一の MPI ジョブとして実行する機能を有している。全ケースのシミュレーションが完了するまでケース間の通信、同期は一切発生せず、完了したケースは他の全ケースが完了するまで待つため、ジョブ全体として実行時間が最

も実行時間が長いケースのものとなる。本研究で使用するデータは、どのケースでも実行時間はほとんど変わらず、ケース数が増えることによるオーバーヘッドもほとんどなかった。

システムの利用状況 (ジョブスケジュール) にあわせて、1ジョブ当たりのケース数を工夫することで計算効率性が向上する。例えば、既に実行中のジョブが多い状況では、1ジョブ当たりのケース数を少なくした方がジョブスケジュールの空きを効率良く利用することが出来る。一方、実行中のジョブがない、もしくは少ない状況では、1ジョブ当たりのケース数を増やすことで同時に複数ケースの結果を得ることができ、システムの最大同時ジョブ実行数が少ない場合に有効である (図3参照)。新 ES は、複数のクラスタ (高速ネットワークが接続されている単位) で構成されており、2,048 ノードといった大規模クラスタもある。大規模クラスタを占有すれば、1ケース16ノード64並列実行の場合、一度に128ケースを実行することも可能である。

本研究では、スケーラビリティの検証結果とシステムの利用状況から、2ケース32ノードジョブを採用した。また、新 ES のシステム制限値である同時実行数を緩和することで、大量ジョブを効率良く実行できるように調整した。

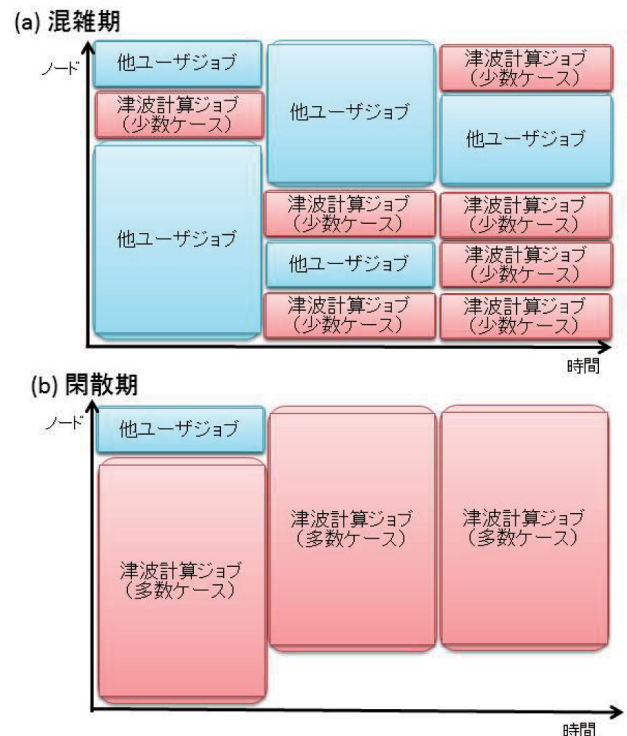


図3 複数ケースジョブにおけるジョブスケジューリング。

4. 津波浸水データベースの構築

地球シミュレータに最適化されたコードを用いて、南海トラフで発生する様々な津波にを予め計算し、徳島県全沿岸部での津波浸水予測をデータベース化する。津波の計算に必要な震源断層モデルは、南海トラフの海溝型地震を対象に絞って、約200通り構築した。震源断層モデル群においては、有意な津波を励起するマグニチュー

ド (M) 6.5 程度から最大クラスの M9 を対象とし、各断層は矩形断層で近似し、内閣府 (2012[5]) のプレート境界面深度モデルの上面に配置した。断層長や幅、滑り量分布の設定には、スケーリング則 (e.g. Murotani et al., 2013[4]) や、東北地方太平洋沖地震、昭和東南海・南海地震など過去の海溝型巨大地震のモデルを参考とし、断層すべりの空間的不均質さもある程度考慮した (図 4)。

各震源断層モデルに対し、徳島県沿岸部全域で予想される津波高と浸水分布を地球シミュレータにより計算した。計算については、当該モデルによる地殻変動を半無限均質弾性モデル (Okada, 1985[6]) により求め、その鉛直成分を津波の初期水位分布として与えた。ネスティング手法を用いて、沿岸部に近づくにつれて分解能が細くなるモデルを採用し、沿岸部での地形分解能は 5m とし

た。津波発生から 6 時間分の津波を計算した。地形データには、平成 24 年徳島県委託業務「津波解析用データ作成業務」により整備されたものを用いた。計算における津波防御施設の扱いは、南海トラフ地震 3 連動以上 (およそ M8.5 以上) では津波防御施設は破壊するとして、それ以下では健全とした。潮位は T.P.=0 として解析した。

本システムでは気象庁から提供される即時震源情報を基に、上述の地震動・津波 DB を検索・マッチングさせて、最適なシナリオを特定し、その震度分布と浸水深分布を表示する。シナリオ検索は、震源の位置、マグニチュードの 3 つという少ない情報に頼るため、ロジックツリー (図 5) を作成し、それに基づいてシナリオを特定するアルゴリズムを構築した。

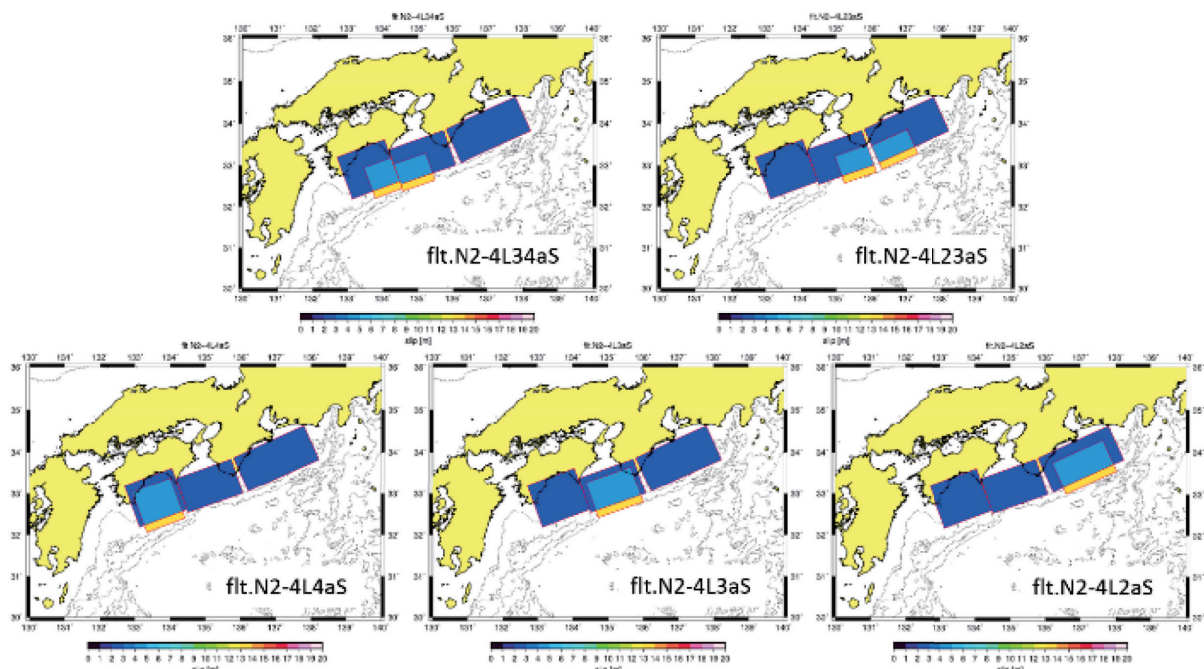


図 4 震源断層モデルの例。青色の部分は背景すべり域、水色は大すべり域、黄色は超大すべり域を示す。

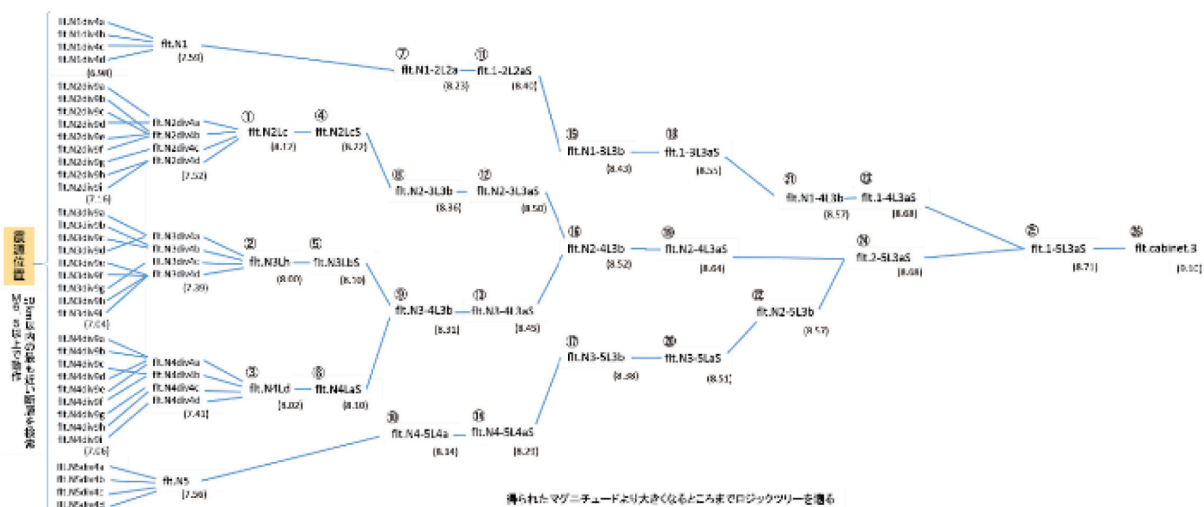


図 5 津波データベース検索のためのロジックツリー。M6.5 以上で起動し、震源位置と比較し最も左側の列にある断層のうち近接断層を特定、気象庁の提供するマグニチュード (括弧内の数字) を超えるとシナリオまでツリーを右側に移動し、止まったところのシナリオが最適として選ばれる。

5. まとめ

本稿では、地球シミュレータの社会貢献の一例としての取り組みを紹介した。多くのシナリオに対して、高い空間分解能で津波予測を実施するには、多くの計算資源が必要である。地球シミュレータへの津波計算コードの最適化、および地球シミュレータを用いた大規模計算により、本課題の実用化に一定の方向性を示した。

謝辞

本研究の実施にあたり、海洋研究開発機構地球情報基盤センターより、多大なご尽力と賜りました。特に上原均氏、今任嘉幸氏には業務の枠を超えて、本研究の実施にご協力いただきました。記して感謝いたします。

文献

- [1] 安藤和人, 馬場俊孝, 松岡大祐, 加藤季広, 京コンピュータによる大規模津波シミュレーション, 情報処理, 55, pp.4-9, 2014.
- [2] Baba, T., N. Takahashi, Y. Kaneda, K. Ando, D. Matsuoka, and T. Kato, Parallel implementation of dispersive tsunami wave modeling with a nesting algorithm for the 2011 Tohoku tsunami, *Pure appl. Geophys.*, doi:10.1007/s00024-015-1049-2, 2015.02.
- [3] Baba, T., K. Ando, D. Matsuoka, M. Hyodo, T. Hori, N. Takahashi, R. Obayashi, Y. Imato, D. Kitamura, H. Uehara, T. Kato, and R. Saka, Large-scale, high-speed tsunami prediction for the great Nankai trough earthquake on the K computer, *Inter. Jour. of High Per. Comp. App.*, 30, 71-84, doi:10.1177/1094342015584090, 2016.02.
- [4] Murotani, S., K. Satake, and Y. Fujii, 2013, Scaling relations of seismic moment, rupture area, average slip, and asperity size for M~9 subduction-zone earthquakes. *Geophys. Res. Lett.*, 40:5057-5074. doi:10.1002/grl.50808.
- [5] 内閣府, 2012, 南海トラフ巨大地震モデル検討会, <http://www.bousai.go.jp/jishin/nankai/model/>, last access on 2016.05.12.
- [6] Okada, Y., 1985, Surface deformation due to shear and tensile faults in a half-space, *Bull. Seism. Soc. Am.*, Vol. 75, pp. 1435-1154, 1985.

A Tsunami Database Created by Using Earth Simulator for Real-time Prediction

Project Representative

Toshitaka Baba Faculty of Engineering, Tokushima University

Authors

Toshitaka Baba Faculty of Engineering, Tokushima University

Atsushi Mikami Tokushima University

Makoto Bando Tokushima Prefecture

Emergency disaster operations should be started rapidly and properly to save lives after the great tsunami disasters happen. In order to enhance the emergency disaster operation, we are developing a prototype system predicting tsunami inundation to the coastal area in Tokushima prefecture. Basic algorithm of the prototype is similar with that of the national tsunami early warning system in Japan that selects an appropriate earthquake scenario from pre-computed tsunami database based on the epicentral location and magnitude. A difference between the JMA's system and this study can be seen in prediction target. They predict tsunami height at the coast line, but our system will predict tsunami inundation on land. In creating the tsunami inundation database, huge computational resources are required so that we at first optimized tsunami simulation code on Earth Simulator. Then, we defined about 220 earthquake scenarios possibly occurred in the Nankai subduction zone with a range of magnitude from 6.5 to 9.0. All tsunamis generated the created earthquake scenarios were calculated to construct a tsunami inundation database. Earth Simulator successfully accomplished constructing the tsunami database within a reasonable time, which was embedded to the prototype of tsunami prediction system.

Keywords: Tsunami, Nankai trough, tsunami early warning

1. Introduction

The Nankai earthquakes are anticipated to occur accompanied by large tsunamis. Emergency disaster operations should be started rapidly and properly to save lives after the great tsunami disasters happen. In order to enhance the emergency disaster operation, we need to provide a possible tsunami inundation area as soon as possible. Although a site survey will be conducted after a half day or a day by using a helicopter for example, numerical tsunami predictions using the real-time seismic and tsunami observation are solely available until the first 12 hours to draw a big picture of the disaster. We accordingly are developing a prototype system predicting tsunami inundation to the coastal area in Tokushima prefecture based on numerical simulation. The system applies a database-driven algorithm so that we have to prepare a tsunami inundation database containing all tsunami scenarios that could occur in the Nankai trough subduction zone. In this study, Earth Simulator was used in constructing of the database. But we at first optimized tsunami simulation code on Earth Simulator to carry it out efficiently. We describe the optimization methods and created tsunami database in this manuscript.

2. Optimizing on Earth Simulator

We used a tsunami calculation code called JAGURS that solves linear/nonlinear long-wave/Boussinesq equations with

a nested algorithm on a staggered, leap-frog finite differential scheme (Baba et al., 2015[1]). This code was paralleled by using OpenMP and MPI.

Velocity update routine in JAGURS originally contained many IF statements that caused long waiting periods for floating-point instructions to be completed. We therefore introduced the following code enhancements (Table 1). In the original code, propagating waves of dry grid cells (i.e. on land, where there was no propagating wave) were not processed in the velocity update loop. However, we modified the code to apply the propagating wave processing to all grid cells regardless of whether they were dry or wet. This allowed vector optimization and software pipelining. In grid cells located on land, velocities were overwritten with zeros after the first pass through the velocity update loop. Because of the complicated IF statement structure, vector optimization and software pipelining were not applied during this overwrite, but the time cost of this operation was negligible. To further improve vector optimization and software pipelining of propagating wave processing, we divided the inner loop into several subloops, each with only one IF statement. We also used an option that enabled the compiler to generate masked instructions.

As result of the optimization, averaged vector operation rate and ADB element hit rate were improved from 75.9% to 97.9%, and from 36.4% to 67.8%, respectively. Calculation speed

became to be 6.1 times faster.

Table 1 Optimization in velocity update process

original	optimized
do i = ist, ind	do i = ist, ind
do j = jst, jnd	do j = jst, jnd
if (branch 1) then	process 1
process 1	end do
else if (branch 2) then	end do
process 2	do i = ist, ind
else if (branch 3) then	do j = jst, jnd
process 3	if (branch 2) then
else process 4	process 2
end if	end do
end do	end do
end do	:
	:

3. Multi-scenario control

The tsunami code, JAGRUS, has a function to calculate multi-scenario as a simultaneous run. All scenarios are completely independent, no data communications and process synchronism between the scenarios are needed. Processor assigned by a small computation waits another processor assigned by large computation so that the total wall time for computation is the same with the longest job in the multi-scenarios. However, scenarios used in the study incurred almost the same calculation loads. Overhead time was negligible in our cases.

Depending on scheduling of all jobs submitted to Earth simulator, we changed a number of scenarios per a submission. For example, when many nodes are occupied by other jobs, a multi-scenario job consisting a small number of scenarios was submitted to slip in empty nodes among occupied nodes. If no other job or a small number of jobs are submitted to Earth simulator, a submission of multi-scenario job consisting a large number of scenarios is efficient. Although Earth simulator is consist of plural clusters, the biggest cluster has 2,048 computational nodes. We can calculate our 128 scenarios at the maximum, in which each scenario uses 16 nodes, at once on Earth Simulator.

4. Creating tsunami inundation database

We created a tsunami inundation database for Tokushima prefecture by using the optimized code on Earth Simulator. We defined about 220 earthquake scenarios possibly occurred in the Nankai subduction zone with a range of magnitude from 6.5 to 9.0. We here assumed heterogeneous slips on the fault planes. We repeatedly calculated tsunamis by changing the earthquake scenarios with the multi-scenario control to evaluate tsunami inundation on land with spatial resolution of 5 m interval. All data were stored in a tsunami database. A logic tree was constructed to select only one appropriate scenario from the tsunami database based on the epicentral location and magnitude

provided by the JMA's disaster information XML.

5. Conclusion

We introduced a practical use of Earth simulator in the paper though a cooperative work between a university and a local government. Earth simulator can contribute a tsunami assessment in real time in Tokushima prefecture. The tsunami prediction system is expected to be used in operational in the near future.

However, in terms of total performance of the tsunami prediction system, there are points need to be improved. The algorithm doesn't take into account earthquake rupture extent, which is not negligible in prediction of tsunami with high-accuracy. Also, it is not good at tsunami earthquake which generates a large tsunami with weak seismic shaking. We accordingly need to add a function to upgrade (re-select) scenario based on tsunami height observations off shore and at the coast, which are provided by follow-up information of the JMA's XML or direct connection to seafloor observatory such as Dense Oceanfloor Network system for Earthquakes and Tsunamis (Kaneda et al., 2015[2]). We will also consider to improve the prediction accuracy by increasing number of the earthquake scenarios. In that case, we should use more computational resources of Earth simulator.

Acknowledgement

We received kind supports from Center for Earth Information Science and Technology. Especially, we thank Mr. Yoshiyuki Imato and Dr. Hitoshi Uehara for continuous technical supports.

References

- [1] Baba, T., N. Takahashi, Y. Kaneda, K. Ando, D. Matsuoka, and T. Kato, Parallel implementation of dispersive tsunami wave modeling with a nesting algorithm for the 2011 Tohoku tsunami, *Pure appl. Geophys.*, doi:10.1007/s00024-015-1049-2, 2015.
- [2] Kaneda Y., N. Takahashi, T. Baba, K. Kawaguchi, E. Araki, H. Matsumoto, T. Nakamura, S. Kamiya, K. Ariyoshi, T. Hori, M., Hyodo M. Nakano, J.-K. Choi, S. Nishida, and T. Yokobiki, Advanced Real Time Monitoring System and Simulation Researches for Earthquakes and Tsunamis in Japan, in V. S.-Fandiño et al. (ed.) : *Post-Tsunami Hazard: Restoration and Reconstruction*, SPRINGER, pp. 179-189, doi:10.1007/978-3-319-10202-3_12, 2015.

