

jh180040

大規模津波浸水被害推計シミュレーションの マルチプラットフォーム向け最適化手法の研究

撫佐 昭裕（東北大学サイバーサイエンスセンター）

概要

本津波浸水被害推計シミュレーションは、内閣府の総合防災システムとして、南海トラフ地震発生時には SX-ACE を用いて 5 分以内に静岡県から鹿児島県までの津波被害推計を行うものである。本研究では、Intel Xeon, Xeon Phi, GPU などのスーパーコンピュータに本プログラムを移植・高速化し、将来、有事発生時に SX-ACE だけでなく、複数のスーパーコンピュータで被害推計が行えるようにするものである。今年度は、Intel Xeon と Xeon Phi へのプログラム移植と高速化を実施し、5 分以内にシミュレーションを行うことが可能となった。また、本プログラムは階層型格子において多角形格子を採用しており、演算のロード・インバランスが存在していた。本年度はこのロード・インバランスを削減したモデル開発し、実行時間を最大 50%削減した。さらに、GPU においては OpenACC を用いて本プログラムの移植を実施した。

1. 共同研究に関する情報

(1) 共同研究を実施した拠点名

大阪大学, 東北大学

(2) 共同研究分野

□ 超大規模数値計算系応用分野

(3) 参加研究者の役割分担

所属：東北大学サイバーサイエンスセンター		
1	氏名	撫佐 昭裕
	役割	全体統括, プログラム高速化検討
2	氏名	小松 一彦
	役割	Xeon Phi システムチューニング, GPU 高速化検討
3	氏名	下村 陽一
	役割	プログラム高速化検討・実装, システム構成検討
所属：東北大学災害科学国際研究所		
4	氏名	越村 俊一
	役割	副代表, モデル検討
所属：東北大学大学院情報科学研究科		
5	氏名	小林 広明
	役割	高速化の指針検討
6	氏名	佐藤 雅之
	役割	Xeon 高速化検討
7	氏名	岸谷 拓海

	役割	Xeon Phi システムチューニング, GPU 高速化検討・実装
所属：大阪大学サイバーメディアセンター		
8	氏名	伊達 進
	役割	システム構成検討, Octopus 高速化検討

2. 研究の目的と意義

2011 年の東日本大震災では、大規模な津波により北海道から千葉県の大範囲に津波浸水被害がおよび、その全貌把握には数日を要した。そのため、救援、物資支援の遅れによる被害が拡大した。津波災害では、被害の全貌をいち早く把握し、すばやく対応することが求められる。東日本大震災後、政府は災害発生直後に航空機等による被災地調査の体制を整備している。しかし、夜間や悪天の場合には対応できないことや写真画像などの処理に時間がかかり、被害把握には未だ時間を要している。この解決手段の一つとしてコンピュータシミュレーションの活用がある。短時間で津波の伝搬、遡上および建築物被害を推定することができれば、大規模地震発生直後に津波浸水範囲や建築物被害の全貌を把握することができる。そして、その推定結果をもとに救援、物資支援の対応計画をいち早く立案するこ

とが可能になる。

本研究グループは、大規模地震発生時に津波被害の全貌を即時に把握することを目標として、津波浸水被害推計を高精度に行う、リアルタイム津波浸水被害推計シミュレーションの開発を行っている。現在、当研究グループのシミュレーションプログラムは、内閣府総合防災システムの津波被害推計処理に採用され、南海トラフ地震発生時には、東北大学と大阪大学のスーパーコンピュータ SX-ACE を用いて、静岡県から鹿児島県までの沿岸約 6,000Km を 30m の格子サイズで、5 分以内にシミュレーションを行うことになっている（地震データ受信からシミュレーション結果の送信までを 30 分以内）。しかしながら、以下の 3 つの課題を有している。①大規模地震では広い範囲で建屋の倒壊・損壊や停電などの地震被害をもたらす、東北大学と大阪大学の冗長化だけでなく、他サイトのスーパーコンピュータシステムとの冗長化や分散配置が必要であること。②30m の格子サイズでは、地方自治体が整備している津波ハザードマップの 10m 格子サイズより精度が低いこと。③太平洋沖での巨大地震では、北海道から鹿児島まで津波が襲来する可能性があり、延べ 10,000Km 以上に及ぶ海岸部の津波浸水被害推計を行う必要があること。

これら課題を解決するためには、有事発生時に被災を免れたスーパーコンピュータを用いてシミュレーションを実行するマシン環境の整備（①の対策）と、10m 格子サイズによる全国規模のシミュレーションを短時間で実行するためのプログラム高速化（①, ②の対策）が必要となる。

本研究の目的は、複数のスーパーコンピュータにおいて、津波浸水被害推計シミュレーションプログラムの高速化を行い、全国規模のシミュレーションを実現するための要件、具体的にはモデルの規模、スーパーコンピュータ規模数、サイト数を明らかにし、今後の有事発生時のシミュレーション環境の指針を策定するものである。ターゲットとするスーパーコンピュータは、Intel 製 Xeon, Xeon Phi, NVIDIA 製 GPU 及び NEC

製 SX-ACE のスーパーコンピュータである。

3. 当拠点公募型共同研究として実施した意義

当研究グループは、津波工学の専門家によるシミュレーションモデルの研究と計算機科学の専門家による高速化研究を融合し、スーパーコンピュータ SX-ACE において効率良く動作する津波浸水被害推計シミュレーションを開発してきた。

本研究は、この実績をもとに複数の異なるアーキテクチャのスーパーコンピュータにおいて、津波浸水被害推計シミュレーションを効率良く動作させるための高度化と大規模並列化を行うものである。これまで最適化を実施してきた SX-ACE の知見を活用して、Xeon, Xeon Phi, GPU を対象とした高速化の検討、開発を行う。

このため、それぞれのシステムにおける高速化実績を有する研究者との共同研究体制が必要になってくる。さらに、本研究では、全国規模の津波浸水被害推計シミュレーションを実現するため、大規模なシミュレーションを実行できるスーパーコンピュータの計算環境も必要である。また、全国に跨がるシステムの冗長化や分散配置について、データ転送の効率化を含めた検討も必要となる。

以上より、本公募型共同研究を通して共同研究体制の構築を行い、大規模スーパーコンピュータを活用した研究を実施するものである。

4. 前年度までに得られた研究成果の概要

本研究は、本年度新規課題である。

5. 今年度の研究成果の詳細

(1) プログラムのベクトル(SIMD)化

近年の汎用プロセッサはベクトル機構（SIMD 機構）を搭載することによって高い性能を実現している。そのため、汎用プロセッサの高速化においてベクトル化は重要な高速化技術となっている。

本プログラムは、ベクトル型スーパーコンピュータ SX-ACE においてベクトル化を行っており、そのコードで Intel Fortran コンパイラでベクトル化できるか調査を行った。ここで、表 1 が SX-ACE と Intel コンパイラで使用したオプションである。

表 1 コンパイルオプション

コンパイラ	オプション
SX-ACE	-Chopt -pi
Intel	-O3 -ipo

表 2 がプログラムのベクトル化状況を表している。「No. of loops」が DO ループの個数、「VEC loops」がベクトル化すべきループの個数、「SX」が SX-ACE の Fortran コンパイラがベクトル化を行ったループの個数、「Intel」が Intel コンパイラがベクトル化を行ったループの個数、「Intel Directive」が Intel コンパイラ指示行を用いてベクトル化したループの個数である。

表 2 DO ループのベクトル化状況

	No. of loops	VEC loops	SX	Intel	Intel Directive
個数	91	35	43	20	26

SX-ACE のコンパイラはベクトル化できるループをすべてベクトル化している。さらに多重ループの最内ループとその外側のループをベクトル化しており、「VEC loops」よりベクトル化したループ数が多くなっている。

一方、Intel コンパイラでは表 1 のオプションではベクトル化できないループがあった。それらは、①Intel コンパイラがベクトル化対象として認識しないループと、②ベクトル化した場合の実行時間を推測してベクトル化の効果がないと判断したループである。①に対してベクトル化を指示するコンパイラ指示行「vector always」、「simd」を挿入したが、ベクトル化は行われなかった。②に対しては、コンパイラ指示行「vector always」の挿入、またはコンパイルオプション「vec-threshold0」でベクトル化でき、性能向上することを確認した。Intel コンパイラは、ベクトル

化によるコスト見積もりを行い、ベクトル化を行うか判断しているが、誤っている場合があり、コンパイラの判断だけでなく、実際に実行して効果を確認する必要がある。

(2) プログラム性能

SX-ACE, Xeon Gold, Xeon Phi における 1 コアの性能を示す。表 3 に各システムの諸元、表 4 にコンパイル条件、表 5 にシミュレーション条件を示す。

図 1 が SX-ACE, Xeon Gold, Xeon Phi の 1 コアにおける実行時間 (分) である。なお、Xeon Phi はクラスタモードとして all-to-all、メモリモードとして FlatMCDRAM で計測を行った。

表 3 システム諸元

	SX	Gold	Phi
CPU 型番	-	6126	7210
周波数 (GHz)	1	2.6	1.3
コア数	4	12	64
コア性能 (単精度) (Gflops)	64	108.8	83.2
CPU 性能 (単精度) (Gflops)	256	1305.6	5324.8
メモリバンド幅 (CPU 当たり) (GB/s)	256	128	115.2
CPU 当たりの B/F	1	0.098	0.021

表 4 コンパイル条件

CPU	項目名	コンパイル環境
SX-ACE	-	-Chopt -pi (指示行なし)
Xeon Gold	Def	-O3 -ipo (指示行なし)
	Direc.	-O3 -ipo (指示行あり)
	AVX512	-O3 -ipo -xCORE-AVX512 (指示行あり)
Xeon Phi	Def	-O3 -ipo (指示行なし)
	Direc.	-O3 -ipo (指示行あり)
	AVX512	-O3 -ipo -xMIC-AVX512 (指示行あり)

表 5 シミュレーションの条件

対象地域	高知県
範囲	1288 km x 1025 km
階層格子	多角形格子
格子サイズ	810 m, 270 m, 90m, 30m
格子数	4.8×10^6
Δt	0.5 秒
再現時間	6 時間

Xeon Gold, Xeon Phi とともにベクトル化の指示行挿入によって、それぞれ実行時間を 27%と 49%削減している。このことは Xeon Gold, Xeon Phi とともに高速化においてベクトル化が有効であることを示している。また、AVX512 命令の使用によって、さらにベクトル化した D0 ループの時間が短縮され、Def に比較して実行時間が 1/2 と 1/5 になっている。マルチコアにおける実行時間を図 2 に示す。現在、リアルタイム津波浸水被害推計システムとして、シミュレーション時間 5 分以内を目標としているが、それぞれ 5 分以内にシミュレーションを完了している。

本プログラムの B/F 値はソースコードから 1.8 である。SX-ACE ではハードウェアカウンタを用いることで、実行時の B/F 値 (実行 B/F 値) を採取することができる。実行 B/F 値は 0.66 であり、これはコンパイラによる最適化とキャッシュによるメモリロードの削減により 1.8 から 0.66 に減少している。この実行 B/F 値と SX-ACE の B/F 値 1 から、SX-ACE でのプログラム実行では演算器がボトルネックになっていることがわかる。一方、Xeon Gold と Xeon Phi では、実行時の B/F 値は採取できていないが、SX-ACE と同等と想定

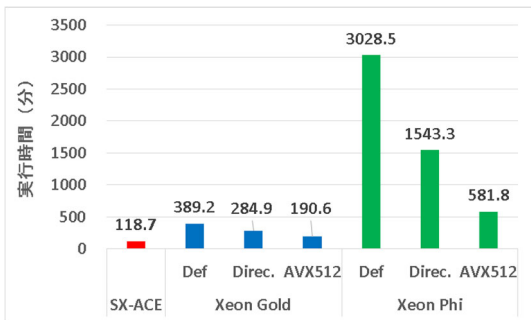


図 1 1 コアでの実行時間

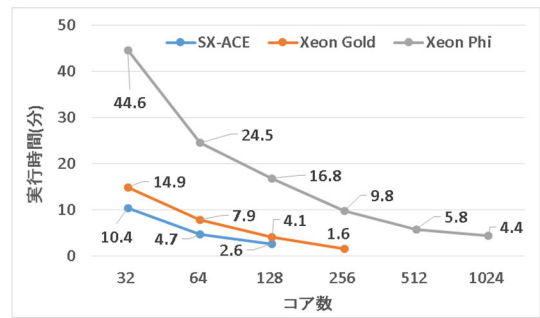


図 2 マルチコアにおける実行時間

すると、Xeon Gold と Phi で B/F 値が 0.098 と 0.021 であることからメモリバンド幅がボトルネックとなっている。今後の高速化では、SX-ACE は演算器の効率的利用、Xeon Gold と Xeon Phi ではメモリ負荷の低減が必要となる。

また、SX-ACE 128 コア実行時における各プロセスの演算量 (浮動小数点データの個数) の最大値と最小値を表 6 に示す。二つのプロセスで演算量が約 8 倍異なっている。そのため、演算のロード・インバランスが発生しており、表 6 の Proc. 124 (演算量が最小のプロセス) ではアイドル時間が処理時間の 85%を占めている。

表 6 128 コアにおける演算量とアイドル時間

	Proc. 46	Proc. 124
経過時間(秒)	128.4	128.4
演算量(ギガ個)	4565	576
アイドル時間(秒)	5.9	109.7

(3) プログラムの高速化

前節で本プログラムには演算のロード・インバランスがあることを示した。本コードは階層型格子を採用しているが、シミュレーションの時間刻み (Δt) を大きくするため、海洋の深い領域を避けるように格子を組んでいる。図 3 が

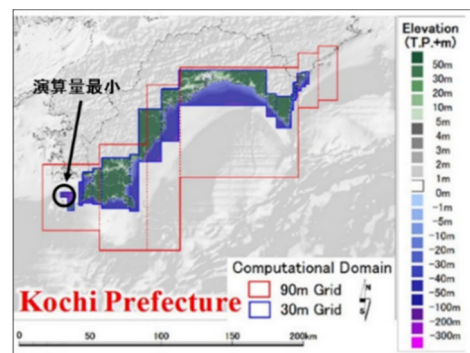


図 3 格子構造 (多角形格子)

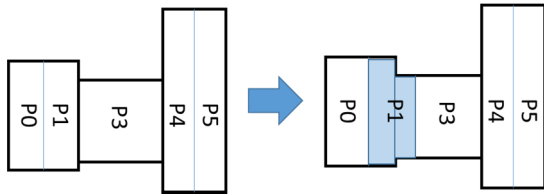


図 4 プロセスの割当手法

高知県の格子構造を示したものであり、90m と 30m の格子が深水の深いところを避けるように複数の四角形（ブロックと呼ぶ）を組み合わせる階層格子を作っている（多角形格子構造）。図中の黒丸で示したブロックが最も演算量が少なくなっている。各ブロックに一つ以上のプロセスを割り当てて並列化を行っているが、演算量の少ないブロックにより、演算のロード・インバランスが発生している。

この演算のロード・インバランスを解消するため、図 4 に示したようにブロック間を跨いでプロセスを割り当てられるように並列化を行った。図 5 が SX-ACE における従来の並列化と本並列化の実行時間である。この並列化により、演算のロード・インバランスが減少し、図 5 のように 32 コアで実行時間が半分になっている。また、従来の 128 コアの実行時間より、本並列化による 64 コアの実行時間の方が短くなっている。この高速化で津波シミュレーションをより短時間に、また、シミュレーション領域を拡大できるようになった。尚、Xeon Gold と Xeon Phi については 2019 年度に移植する予定である。

(4) プログラムの GPU への移植

本研究ではマルチプラットフォームでのプログラム実行を目的にしているため、プログラムの GPU への移植は OpenACC を用いて行った。GPU

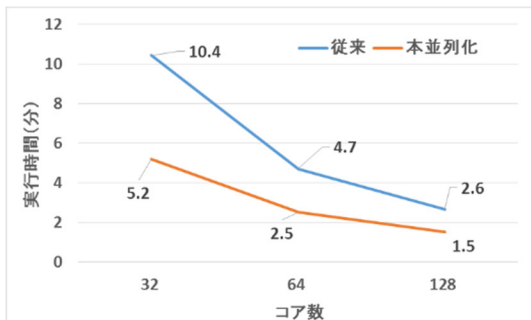


図 5 SX-ACE での実行時間

```

サブルーチン 例
SUBROUTINE sub_G_1
  . . .
  !$sacc kernels copyin( . . . ) copyout( . . . )
  DO J=JSTA, JEND
    DO I=ISTA, IEND
      . . .
    END DO; END DO
  !$sacc end kernels
  . . .
RETRUN; END
    
```

図 6 予備実験でのプログラム実装例

は NVIDIA Tesla P100 を使用した。本プログラムは二つのサブルーチンでプログラム全体の 96% のコストを占めている。予備実験として DO ループごとに演算に必要なデータを転送する実装を行った（図 6）。この実装ではデータ転送により Xeon Gold の 9 倍以上の実行時間がかかることが判明した。

次に、二つのサブルーチンで演算を行うデータを予め GPU へ転送を行い、ホスト側で演算を行うために必要なデータとホスト側で更新されたデータのみをデバイス・ホスト間で転送する

```

メインプログラム 例
MODULE com_data
  REAL ,DIMENSION(:,:),ALLOCATABLE:: . . .
  INTEGER ,DIMENSION(:,:),ALLOCATABLE:: . . .
  !$sacc declare create( . . . )
END MODULE com_data

(初期化処理 等)

!$sacc enter data create( . . . )
!$sacc update device( . . . )
DO T=1,X (時間発展のループ)
  CALL sub_G_1 (GPU上で計算処理)
  !$sacc update host( . . . )
  CALL sub_H_1 (HOST上で計算処理)
  !$sacc update device( . . . )
  CALL sub_G_2 (GPU上で計算処理)
  !$sacc update host( . . . )
  . . .
  CALL sub_H_n (HOST上で計算処理)
  !$sacc update device( . . . )
END DO

(終了処理)
STOP; END
    
```

```

サブルーチン 例
SUBROUTINE sub_a
  USE com_data
  . . .
  !$sacc data present( . . . )
  !$sacc para;llel
  !$sacc loop gang
  DO J=JSTA, JEND
    !$sacc loop vector
    DO I=ISTA, IEND
      . . .
    END DO; END DO
  !$sacc end parallel
  . . .
RETRUN; END
    
```

図 7 プログラム例

ように実装を行った。図 7 がプログラム例である。サブルーチン間でデータを共有するため、OpenACC の declare create を使用し、enter data

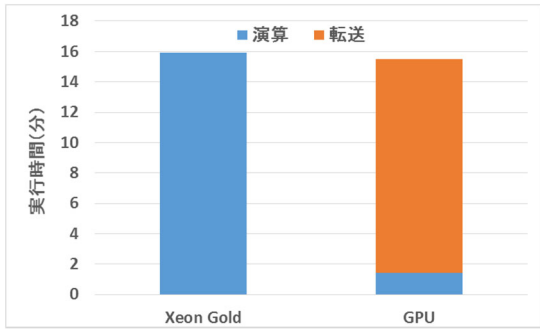


図 8 GPU での実行時間

create でデバイス上にメモリ領域を確保した。また、初期化処理後にデバイスへデータを転送し、それ以後、ホストとデバイスで必要となるデータのみを update 命令で転送するようにした。この実装によりデータ転送量を削減した。

図 8 が 30 分間の津波現象を Xeon Gold 1 コアと GPU でシミュレーションしたときの実行時間である。GPU の演算時間は Xeon Gold の約 1/10 になっているが、ホストとデバイス間の転送時間が 14 分もかかっている。この実装においてもデータ転送時間が大きく、また、現状の配列データの持ち方では粒度が小さく GPU のコア性能を十分に活かしていない。2019 年度には、時間発展の処理をすべて GPU で実行し、かつ粒度を大きくするデータ配置を検討する予定である。

(5) Xeon Phi システムチューニング手法

Xeon Phi プロセッサは 38 個のタイルから構成され、最大 72 コアからなり、ハイパースレッディングにより最大 288 スレッドまでの並列実行が可能である。また、高メモリバンド幅の MCDRAM を搭載し、DDR メモリと階層型メモリシステムになっている。これらタイルとメモリを最適に利用するために複数のシステムパラメータでシステム構成を変更することができる。表 7 に選択可能なパラメータを示す。これらパラメータの組み合わせは 300 通りあり、実行するプログラムごとに最適な設定を選択する必要がある。

本研究では、STREAM ベンチマーク等の著名なベンチマークプログラムの Xeon Phi での性能情

報とプログラムの特性から 300 通りの性能評価を行わなくてもパラメータの絞り込みが行える手法を開発した。

表 7 システムパラメータ一覧

項目	設定できるパラメータ
クラスタモード	all-to-all, Hemisphere, Quadrant, SNC-4, SNC-2
メモリモード	Flat DDR, FlatMCDRAM, Cache, Hybrid25, Hybrid50
スレッドアフィニティ	Compact, Scatter, Balanced
スレッド数	72, 144, 216, 288

一般的にシミュレーションプログラムは実行時のメモリ容量は既知であり、プログラムの特性としてメモリ律速か演算律速も既知である。これらの情報と著名なベンチマークの性能情報からパラメータの絞り込みを行う。ここでは本プログラムにおけるパラメータの絞り込み方法を示す。図 9 と図 10 は STREAM ベンチマークにおける Xeon Phi の性能情報である。図 9 がス

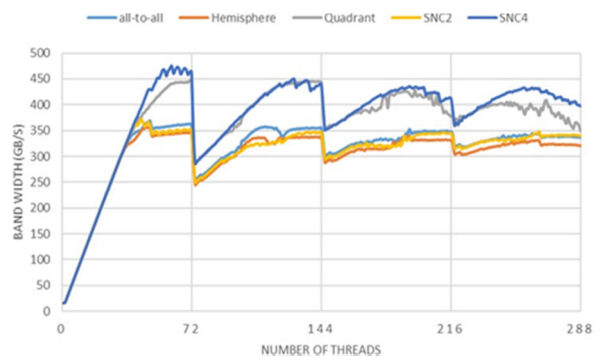


図 9 STREAM におけるメモリバンド幅性能 (スレッド数とスレッドアフィニティ)

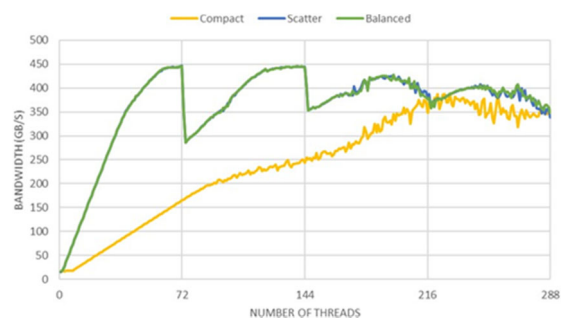


図 10 STREAM におけるメモリバンド幅性能 (スレッド数とクラスタモード)

レッド数とスレッドアフィニティを変えた場合のメモリバンド幅性能である。図 10 がスレッド数とメモリモードを変えた場合のメモリバンド幅性能である。

本プログラムはメモリ容量が MCDRAM の容量を超えないので、メモリモードは FlatMCDRAM となる。次に図 9 と図 10 からメモリバンド幅が高くなるパラメータを選択する。具体的にはスレッド数は 72 か 144, スレッドアフィニティは Scatter か Balanced, クラスタモードは Quadrant か SNC-4 になる。以上より本プログラムに最適なパラメータは表 8 の組み合わせの 8 通りとなり, 8 回の試行で最適なパラメータを決めることができる。この 8 通りの実測を行い, 最終的に実行時間が最短になるパラメータは表 8 の赤字で示したパラメータの組み合わせであった。

表 8 最適なパラメータ候補

項目	パラメータ
クラスタモード	Quadrant, SNC-4
メモリモード	Flat MCDRAM
スレッドアフィニティ	Scatter, Balanced
スレッド数	72, 144

(6) 全国規模に向けたシミュレーション

本研究の目的は, 10m の格子サイズで全国規模の津波シミュレーションをリアルタイムで行うことである。ここで, (3) の高速化したプログラムにおける高知県 10m 格子の SX-ACE における実行時間を図 11 に示す。512 コア (64 ノード) で 4 分以内にシミュレーションを完了している。

南海トラフ地震の津波被害が予測されている西日本の太平洋沿岸の演算量は高知県の 27 倍, 日本全国の太平洋沿岸は 81 倍になる。SX-ACE で

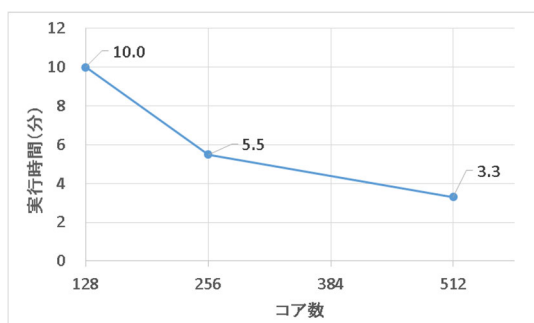


図 11 SX-ACE 10m 格子の実行時間

シミュレーションを行う場合には表 9 のノード数が必要になる。4 分以内に全太平洋沿岸をシミュレーションする場合には HPCI-JHPCN の全 SX-ACE を超えるノード数が必要になる。全国規模のシミュレーションのためには SX-ACE 以外のスーパーコンピュータの利用が必要になることがわかる。2019 年度には Xeon Gold と Xeon Phi による必要資源の見積と分散配置の検討を行う。

表 9 シミュレーションに必要なノード数

SX-ACE	高知県	西日本	全国
6 分以内	64	1728	5184
4 分以内	128	3466	10368

6. 今年度の進捗状況と今後の展望

今年度の計画は, ①プログラムの Xeon Gold, Xeon Phi への移植と最適化, ②GPU への移植, ③大規模並列化の対応, ④システムの冗長化と分散配置の検討であった。

①については, 5 章(1), (2), (3)に示したように Xeon Gold, Xeon Phi とともにベクトル化を行い, マルチコア実行で SX-ACE と同等の実行時間を達成した。②については, 5 章(4)に示したように OpenACC による GPU への実装を行った。しかし, 現在のプログラム構造ではデータ転送が多く, GPU の性能を十分に使い切れていない。2019 年度は, データ転送の削減と粒度を大きくしたモデルを開発する。③については, 演算のロードバランスを改善し, SX-ACE で 1.7 倍から 2 倍の性能向上を達成した。2019 年度は, 本プログラムを Xeon Gold と Xeon Phi へ移植を実施するとともに, MPI 時間の削減を検討する。④については, SX-ACE における全国規模のシミュレーションに必要なノード数を算出した。2019 年度は Xeon Gold, Xeon Phi, GPU を含めた計算資源での全国規模のシミュレーションの資源量を算出する。

以上より, 本年度予定していた研究はほぼ達成し, 2019 年度は高速化したプログラムを用いて全国規模のシミュレーションの実現性を議論する。

7. 研究成果リスト

(1) 学術論文

・ T. Inoue, T. Abe, S. Koshimura, A. Musa, Y. Murashima, and H. Kobayashi, “Development and Validation of a Tsunami Numerical Model with the Polygonally Nested Grid System and its MPI-Parallelization for Real-Time Tsunami Inundation Forecast on a Regional Scale,” *J. Disaster Res.*, Vol.14, No.3, pp. 416-434, 2019.

(2) 国際会議プロシーディングス

・ K. Komatsu, T. Kishitani, M. Sato, A. Musa, H. Kobayashi, “Search Space Reduction for Parameter Tuning of a Tsunami Simulation on the Intel Knights Landing Processor,” 117-124, doi:10.1109/MCSoc2018.2018.00030, 2018.

・ K. Komatsu, S. Momose, Y. Isobe, M. Yokokawa, O. Watanabe, T. Aoyama, M. Sato, A. Musa, H. Kobayashi, “Performance Evaluation of a Brand-New Vector Supercomputer SX-Aurora TSUBASA,” In: *The International Conference on High Performance Computing, Networking, Storage and Analysis (SC18)*, pp. 54:1-54:12, IEEE Press, Dallas, USA (November 2018), doi:10.1109/SC.2018.00057.

(3) 国際会議発表

・ A. Musa, T. Kishitani, T. Inoue, H. Hokari, M. Sato, K. Komatsu, Y. Murashima, S. Koshimura, H. Kobayashi, “Performance Evaluation of a Real-Time Tsunami Inundation Forecast System on Modern Supercomputers,” 15th Annual Meeting Asia Oceania Geoscience Society, Honolulu, USA, June 4-8, 2018.

・ K. Komatsu, T. Kishitani, M. Sato, A. Musa, H. Kobayashi, “Search Space Reduction for Parameter Tuning of a Tsunami Simulation on the Intel Knights Landing Processor,” *IEEE 12th International Symposium on Embedded Multicore/Many-core Systems-on-Chip Special session Auto-Tuning for Multicore and GPU*, 2018.

・ K. Komatsu, S. Momose, Y. Isobe, M. Sato, A. Musa,

H. Kobayashi, “Early Evaluation of a New Vector Processor SX-Aurora TSUBASA,” *ISC High Performance 2018*, Frankfurt, Germany, June 24-28, 2018.

・ K. Komatsu, S. Momose, Y. Isobe, M. Yokokawa, O. Watanabe, T. Aoyama, M. Sato, A. Musa, H. Kobayashi, “Performance Evaluation of a Brand-New Vector Supercomputer SX-Aurora TSUBASA,” In: *The International Conference on High Performance Computing, Networking, Storage and Analysis (SC18)*, Dallas, USA, (November, 2018).

・ T. Kishitani, K. Komatsu, A. Musa, M. Sato, H. Kobayashi, “An Application Parameter Search Method Based on the Binary Tree Algorithm for Performance Tuning,” *ISC High Performance 2019*, Frankfurt, Germany, June 16-20, 2019. (Accepted)

・ A. Musa, T. Abe, T. Kishitani, T. Inoue, M. Sato, K. Komatsu, Y. Murashima, S. Koshimura, H. Kobayashi, “Performance Evaluation of Tsunami Inundation Simulation On SX-Aurora TSUBASA”, 9th International Workshop on Advances in High Performance Computational Earth Science in conjunction with ICCS 2019, Portugal, June, 2019. (Accepted)

(4) 国内会議発表

・ 岸谷 拓海, 小松 一彦, 撫佐 昭裕, 佐藤 雅之, 小林 広明, “メニーコアプロセッサのためのパラメータチューニング時間削減手法”, 2018 年並列/分散/協調処理に関するサマー・ワークショップ, 熊本, 7 月 30-8 月 1 日, 2018.

(4) その他 (特許, プレス発表, 著書等)

・ 第一回オープンイノベーション大賞 総務大臣賞, “リアルタイム津波浸水被害予測システムの開発と運用”, 越村俊一, 太田雄策, 村嶋陽一, 撫佐昭裕, 加地正明, 2019 年 4 月.