

通信障害耐性を備えた分散協調型シミュレーションフレームワーク SilkRoad の 3 拠点連携実験

森 眞一郎[†]
福井大学[†]

南里 豪志[‡]
九州大学[‡]

1. はじめに

我々は遠隔地のサーバ間を連携させ高性能なシミュレーションを可能にする分散協調型のシミュレーションを容易に実現するためのフレームワークの開発を行ってきた。過去の研究では、遠隔地の大規模サーバ上で実行中のシミュレーションに対し、実時間ステアリングを行う対話型スーパーコンピューティング環境の構築を行ってきた。これまでの研究では、中央の大規模サーバに、遠隔地の中小規模サーバを従属的に連携させる協調形態をとっていた。

これに対し、主と従の関係を逆転させた新たな連携フレームワーク SilkRoad[1]の開発を現在行っている。SilkRoadは、地理的に異なる地域拠点に置かれ本来独立して動作可能な複数の中規模サーバ（リージョナルサーバ:RSと呼ぶ）が行うシミュレーションを遠隔地の大規模サーバ（セントラルサーバ:CS）を介して連携させることで、個々の地域拠点での単独の計算のみでは得られない高精度な計算結果を得ることを目的としている。具体的には、特定のRSのみでは得られない境界条件の時間変化をCSから入手する一貫性制御(DQC)や、その逆にRSでのみ得られる高精度な計算結果をCSへ反映させる一貫性制御(UQC)を行うことで、RSならびにCSともに計算精度を向上させる機構を実装した。

この際、複数のRSとCSとの間の連携を強くしすぎると特定のサーバで発生した障害が全体に波及しシステム全体のブラックアウトが発生してしまう。そこで、サーバ間を弱連携としたうえで、サーバ間の通信障害が全体に波及しない通信障害耐性を備えたフレームワークの開発を行った。近年多発する災害や通信事業者のトラブルに起因する通信障害への耐性をも備えたシステムといってもよい。



図1. 分散協調シミュレーションの拠点毎の担当計算領域の配置例(2階層の例)

An Experiment of Three remote Regions Cooperative Real-Time Simulation on the Distributed Cooperative Simulation Framework SilkRoad

[†] Shin-ichiro Mori, University of Fukui

[‡] Takeshi Nanri, Kyushu University

現在は、SilkRoadのCSとRSの関係を階層的に繰り返す多階層フレームワークへの拡張を行っており、昨年度、RSの下位層としてエッジサーバ(ES)を導入した3階層モデルの動作検証を行った[2][3]。しかしながら、昨年度の動作検証では、AWS EC2サーバ(東京Region)と福井大学の2台のサーバを連携させた論理的な3拠点連携(地理的には2拠点での連携)で動作の検証であった。

今回、一カ月の期間を定め、九州大学、AWS EC2サーバ(大阪Region)、福井大学の地理的に異なる3拠点を連携させる機会が得られたので、地理的に異なるなる3拠点連携の実証実験を行った結果を報告する。

2. 実験環境

今回の実験では九州大学のスーパーコンピュータ ITOの1ノード、AWS EC2サーバ、福井大学設置のサーバをVPNで接続し相互にMPI通信可能な環境を用意した。すべての拠点でOSはITOに揃えてCentOS7を用いた。ITOのノードはノード固定タイプ(フロントエンド)を利用しリアルタイム利用を可能とすると共に、期間限定の共同研究の一環としてAWS ES2サーバとの間のVPN接続を特別に許可していただいた。

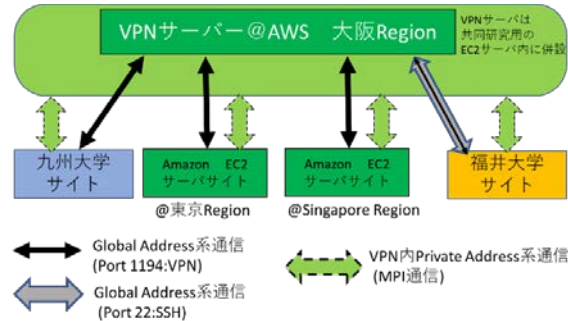


図2 3拠点連携実験時のネットワーク構成

動作検証で使用したシミュレーションは複数の熱源を有し、それらの位置と熱量が時間変化する2次元熱拡散シミュレーションで、各拠点の計算グリッドは512x512の正方格子とし、1グリッドに対応する物理距離はCSからESに向かって階層が一つ下がるごとに1/2となるよう設定した。今回の実験ではシミュレーション自体の計算負荷は高くないので各拠点でのCPU性能は実験結果にさほど影響を与えていないと考えるが、CS,RS,およびESそれぞれIntel XeonGold6140 2.4GHz, Intel XeonPlatinum8180 2.5GHz, および, Intel Core2 Duo E8500 3.1GHzを使用した。また、CS,RSは10Gbps, ESは1GbpsのNICを利用した。

3. 実験結果

3.1 3拠点連携実験

実証実験が一カ月という制約から、SilkRoadの全ての機能の動作検証にはいたらなかったが、通信障害の自動検出・自動復帰の機能、ならびに入力データ集約機能の

一部を制限した状態で、九州大学、AWS EC2 サーバ（大阪 Region）、福井大学の 3 拠点間を連携したシミュレーションを 12 時間連続して動作させる実験に成功した。またこの実験中、早朝の時間帯に福井大学設置のサーバで偶発的な障害が発生したが、九州大学および AWS EC2 サーバ（大阪 Region）の 2 拠点連携動作は継続しており単一障害によるブラックアウトが発生せず自動で縮退運用に切り替わることが確認できた。また、障害発生約 3 時間後に福井大学のサーバを手動で復帰したところ、正常な 3 拠点連携シミュレーションに復帰した。障害からの自動復帰機能を停止していたため、予期せずして発生した偶発障害からの自動復帰のテストはできなかったが、手動復帰の動作は確認できた。

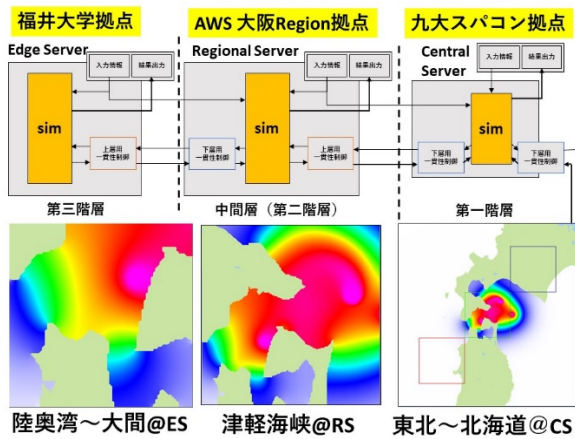


図3 3 拠点連携実験時の連携フレームワークの構成（概要）と各拠点での可視化結果

なお、一カ月の実験期間終盤に、九州大学スパコンの定期保守が入ったためこの期間を利用して、地理的に異なる Region に AWS EC2 サーバを複数用意し、AWS EC2 サーバ 2 拠点と福井大学の 3 拠点連携の実験も行った。すべてのケースにおいて RS は AWS EC2 大阪リージョン、ES は福井大学に固定している。それぞれの拠点ごとに 3 拠点連成が成功した際のシミュレーションの実行間隔を表 1 に示す。理論上の実行間隔の最小値は拠点間の往復遅延時間と通信速度で決まるが、これらの値はシミュレーションの実行中にも大幅な変動が発生する。表 1 はこの通信時間の変動を考慮したうえで実行前にフレームワークに設定するシミュレーション間隔である。なお、九州大学スパコン以外との連携では十分なパラメータチューニングを行っていないので 3 拠点連成可能なシミュレーション間隔はこの値より小さいものと考えられる。

表 1 各 CS 拠点で動作確認できた Simulation 間隔

Central Server 設置場所	Simulation 間隔
九州大学	50ms
AWS 大阪 Region	44ms
AWS 東京 Region	50ms
AWS Singapore Region	150ms
AWS Sydney Region	300ms

3.2 拠点間の通信特性の測定

AWS 大阪 Region に設置した RS と他の拠点との間の往復遅延時間 RTT を ping コマンド(-c 100 オプション)を用いて測定した結果を表 2 に示す。この計測は 4.1 の 3 拠点

連携シミュレーションを実行中に測定した。

表 2 RS と各拠点間の往復遅延時間

Central Server 設置場所	RTT(min/avg/max/mdev)[ms]
九州大学	13 / 31 / 75 / 16
AWS 大阪 Region	5 / 20 / 56 / 11
AWS 東京 Region	11 / 26 / 66 / 12
AWS Singapore Region	79 / 150 / 230 / 41
AWS Sydney Region	117 / 179 / 307 / 54
ES	5 / 21 / 108 / 17

次に、ベンチマークプログラム iperf3[4]を用いて通信速度の測定を行った。RS で iperf3 をサーバモードで起動し、各拠点との間の通信速度を測定した結果を表 3 に示す。iperf3 の実行時オプションは -c RS -V -t 5 である。この測定は連携シミュレーションを実行していない時間帯に行っている。

表 3 RS と各拠点間の通信速度

Central Server 設置場所	通信速度[Mbit/s]
九州大学	Tx 671/Rx 659
AWS 大阪 Region	Tx 703/Rx 694
AWS 東京 Region	Tx 668/Rx 657
AWS Singapore Region	Tx 37.9/Rx 33.5
AWS Sydney Region	Tx 3.13/Rx 2.74
ES	Tx 42.7/Rx 37.6

なお、AWS 大阪 Region と AWS 東京 Region 間で VPN over SSH 環境を用意して通信遅延、通信速度を計測したところ通信遅延時間はほぼ変化がなかったが、通信速度は Tx 451/Rx 441Mbps となり 2/3 倍の速度となった。

5. まとめ

これまで、地理的に異なる 2 拠点のサーバで仮想的に 3 階層の連携実験をおこなってきたが、今回の実験で実世界の 3 拠点間連携実験が成功し我々の分散協調シミュレーションフレームワーク SilkRoad の動作検証ができた。

九州大学のスパコン以外では十分な最適化を行っていないので、シミュレーション間隔の下限と通信特性との関係に関する議論は難しいが、今後は、これらの数値をもとにフレームワークの最適化を行うとともに、今回検証できていない機能の検証を行う機会があれば検証を行いたいと考える。

謝辞：

本研究は一部 JSPS 科研費 JP20K11744 の助成により実施した。本研究は九州大学情報基盤研究開発センター研究用計算機システムの共同研究制度を利用した。

3 拠点連携実験の機会提供、ならびに実験中のサポート体制を提供いただいた AWS ジャパン合同会社、株式会社 Fusic の関係各位に心より感謝いたします。

参考文献

- [1] Jiachao Zhang, Shinji Fukuma, Shin-ichiro Mori : Silk Road: A framework for distributed collaborative simulation, Journal of Information Processing, Vol.26, No.3, pp. 237-246, 2018.
- [2] 三宅 武将, 福岡 慎治, 森 眞一郎 : 分散協調型シミュレーション連携フレームワーク SilkRoad の多階層化モデル 0 の提案, 電気・情報関係学会北陸支部連合大, F1-2, 2021.09.04
- [3] 三宅 武将 : 分散協調型シミュレーションフレームワーク SilkRoad の多階層化, 福井大学大学院工学研究科修士論文, 2022.02.
- [4] iPerf - The ultimate speed test tool for TCP, UDP and SCTP, <https://iperf.fr/>