

戦略的創造研究推進事業 CREST
研究領域「ポストペタスケール高性能計算に資する
システムソフトウェア技術の創出」
研究課題「省メモリ技術と動的最適化技術によるス
ケーラブル通信ライブラリの開発」

研究終了報告書

研究期間 平成23年10月～平成29年3月

研究代表者：南里 豪志
(九州大学情報基盤研究開発センター、
准教授)

§ 1 研究実施の概要

(1) 実施概要

南里チームでは、エクサスケールでの利用に耐える通信ライブラリ、およびそれを活用したスケーラブルなアプリケーションの実現を目的として、通信ライブラリの設計と実装、通信最適化技術の開発、アプリケーションの性能解析と通信最適化、さらに性能解析ツールの開発に取り組んできた。主な研究成果は、以下の通りである。

- ・省メモリ通信ライブラリ ACP (Advanced Communication Primitives)
- ・片側通信対応の通信シミュレータ NSIM-ACE
- ・ACPと本プロジェクトの通信最適化技術を適用したアプリケーション群
- ・パターン通信の動的最適化技術

まず、ACP ライブラリは、主にプロトコルグループが中心となり、インタフェースを設計し、実装した。インタフェース設計にあたっては、片側通信とアトミック操作を前提とした通信プロトコルによる省メモリ性と低遅延性の両立を基本方針とした。ACP は基本層と中間層で構成されている。ACP 基本層は、システム全体で共有する仮想的な PGAS (Partitioned Global Address Space) 型のグローバルメモリ空間を定義し、ローカルメモリ領域のグローバルメモリ空間への登録と、登録された領域のグローバルアドレスによるメモリコピー及びアトミック操作により、プロセス間通信を表現する。ACP 基本層は並列データ構造の実装に適した仕様となっており、グローバルメモリ参照機には、読み出し側もしくは書き込み側のプロセッサがデータ転送を制御する必要がある片側通信ではなく、送信元でも宛先でもない第三者プロセスがグローバルメモリ空間上でメモリコピーを制御する。また、グローバルアドレスは、遠隔アトミック操作が可能であるように、64bit で定義している。現在、この基本層は Tofu、InfiniBand および UDP に対応している。

ACP の中間層は、データライブラリとチャネルライブラリで構成される。このうちデータライブラリは、基本層と同様にプロトコルグループが中心となって設計と実装を行った。これは、グローバルメモリ空間上で分散データ構造を操作するインタフェース群であり、グローバルメモリ上での大域的なデータ配置を最適化し、データ転送のオーバーヘッドが小さいプロトコルとアルゴリズムを提供する。データライブラリで提供するデータ構造としては、C++言語の `hy` 用順テンプレートライブラリを参考に、可変長一次元配列 `vector`、双方向リンクリスト `list`、双方向キュー `deque`、連想配列 `map`、集合 `set` を用意している。

一方、チャネルライブラリは、アプリケーショングループとインタフェースグループが中心となって設計、実装しているインタフェースであり、アプリケーション中で頻りに現れる、プロセス間の同期を伴うパターン通信を簡潔に表記する手段として、チャネルインタフェースとマルチチャネルインタフェースを提供する。チャネルインタフェースは、送信プロセスによる `send` 関数と受信プロセスによる `receive` 関数による両側通信の機能を提供するもので、送受信プロセス対毎に、チャネルと呼ぶ通信路を明示的に確保、解放する関数を用意することにより、通信バッファのためのメモリ使用量の制御を可能としている。さらに、隣接通信や集団通信などのパターン通信を、計算時間で隠蔽する通信オーバーラップを促進するインタフェースとして、マルチチャネルインタフェースの実装に取り組んでいる。これは、アプリケーショングループにおける通信解析の結果、送受信領域の事前登録による `zero copy` 通信の活用と、受信領域の上書きを許可する `ready` 関数の利用が、通信隠蔽に効果的であることが判明したため、ACP の中間層へのフィードバックとして、開発を開始したものであり、本年度末の ACP ライブラリの更新に合わせて公開する予定である。

また、ACP には、複数の MPI プログラムを ACP で連結させる ACP+MPI 機能を実装した。これは、個々の MPI プログラムの参加プロセス数を少なくして MPI による消費メモリ量を抑制しつつ、MPI プログラム間の通信を ACP で行うことにより、全体でのメモリ消費量削減を図るものである。これにより、既存 MPI プログラムをベースとした、スケーラブルなアプリケーション開発が容易となる。

NSIM-ACE は、本プロジェクトで開発する通信最適化技術やアプリケーションの性能解析を目的として、通信路制御グループが中心となって開発した通信シミュレータである。これは、MGEN プログラムと呼ぶ擬似プログラムで表現された通信パターンと計算負荷に対し、対象のネットワークのトポロジーや各種パラメータに応じてパケットの動作をシミュレートするもので、通信路の衝突による性能低下まで見積もることが出来る。NSIM-ACE では、通信インタフェースとして片側通信を利用できるため、ACP プログラムにおける通信パターンの MGEN プログラム化が容易である。アプリケーショングループによる検証で、通信パターンによる通信性能特性の違いを NSIM-ACE で再現できることを確認した。

アプリケーショングループでは、MPI で書かれた並列プログラムの ACP による実装を行った。分子軌道計算プログラム OpenFMO では、マスターワーカー型のプログラムにおいて、負荷分散のためのグローバルカウンタの操作、およびワーカー間のデータ共有を、それぞれ ACP の遠隔アトミック操作とデータライブラリを用いて実装し、メモリ消費量が削減できることを確認した。また、OpenFMO の簡易版に対して ACP+MPI 機能を用いることにより、省メモリ化が達成できることを確認した。電磁流体コードでは、ステンシル計算における Halo 通信と、その通信に依存する計算を Halo スレッドと呼ぶ専用のスレッドに任せることで通信の隠蔽を図った。さらに、Halo 通信を関数として定義し、pack/unpack を含めた最適化を適用して、通信の効率化を確認した。現在、Halo 関数の ACP 実装による、通信隠蔽のさらなる促進を図っている。重力 N 体シミュレーションでは、粒子の移動、および領域の再分割に伴って発生する不規則な通信を、ACP のメモリコピーと遠隔アトミック操作で実装することにより、通信の効率化が図れることを示した。

動的最適化技術については、インタフェースグループと通信路制御グループが中心となって、集団通信および隣接通信の動的最適化技術を開発した。集団通信の動的最適化手段としては、実行時のランク配置やトポロジー、負荷状況に応じてアルゴリズムを選択する動的アルゴリズム選択技術を実装し、有効性を確認した。また、ネットワークの混雑状況に応じてパケットの送信間隔を調整するパケットペーシング技術を集団通信に適用し、Tofu での効果をシミュレーションと実測で確認した。隣接通信については、複数の NIC を用いた多次元トラス網を対象として、NIC を有効に活用して通信速度を向上させる動的最適化技術を開発し、Tofu での効果を確認した。

(2) 顕著な成果

< 優れた基礎研究としての成果 >

1. 通信ライブラリ ACP

概要:

ACP は、省メモリと低遅延を両立させることを目的として開発した。ACP の特徴は、通信用メモリの確保と解放を明示するインタフェース設計と、使用メモリ量の管理が容易であることと、PGAS 型のグローバルメモリ空間を対象としたメモリコピーインタフェースである。このうち後者は、既存の MPI RMA インタフェースや GASNet における put/get と異なり、送信元でも宛先でもない第三者プロセスが通信を制御できるため、分散データ構造の実装に適している。

2. 通信シミュレータ NSIM-ACE

概要:

擬似的な並列プログラムの形で記述した通信パターンと計算負荷をもとに、任意のトポロジーや通信性能パラメータでのパケットの挙動をシミュレートする通信シミュレータである。通信パターンとして、片側通信での記述をサポートしているため、ACP だけでなく MPI RMA による通信パターンもシミュレート可能である。

< 科学技術イノベーションに大きく寄与する成果 >

1. Python、Ruby 向け ACP インタフェース

概要: (200 字程度)

従来の HPC 以外の分野では、プログラミングが容易な Python や Ruby 等のスクリプト言語が普及している。そこで、これらの言語に向けた ACP のインタフェースを開発することにより、HPC 分野以外のアプリケーションの分散メモリ並列化を支援する。

2. ゲノムアセンブリプログラムの並列化

概要: (200 字程度)

バイオインフォマティクスアプリケーションの並列化への取り組みとして、De novo ゲノムアセンブリプログラムを、ACP を用いて並列化した。

§ 2 研究実施体制

(1) 研究チームの体制について

①「インタフェース」グループ

研究参加者

氏名	所属	役職	参加時期
南里 豪志	九州大学情報基盤研究 開発センター	准教授	H23.10～
森江 善之	同上	研究員	H23.10～
小林 泰三	同上	研究員	H26.4～
小松 由良	同上	テクニカルスタッフ	H23.10～
高見 利也	大分大学工学部	教授	H27.9～
深沢 圭一郎	京都大学学術情報メデ ィアセンター	准教授	H27.9～
本田 宏明	九州大学情報基盤研究 開発センター	研究員	H27.9～
薄田 竜太郎	同上	研究員	H27.9～
赤尾 由加里	同上	テクニカルスタッフ	H28.5～H28.9
成林 晃	同上	大学院生	H28.5～

研究項目

- ・スケーラブルな隣接通信及び集団通信のための省メモリアルゴリズム及び動的最適化技術
- ・スケーラブルな並列アプリケーション作成のための通信ライブラリ応用技術

②「プロトコル」グループ

研究参加者

氏名	所属	役職	参加時期
住元 真司	富士通株式会社	シニアアーキテ クト	H23.10～
安島 雄一郎	同上	シニアアーキテ クト	H23.10～
三浦 健一	同上	シニアアーキテ クト	H23.10～H26.1
志田 直之	同上	シニアアーキテ クト	H23.10～

		ト	
岡本 高幸	同上		H23.10～H26.1
秋元 秀行	同上		H23.10～H26.1
佐賀 一繁	同上		H25.4～
安達 知也	同上		H25.4～H26.1
野瀬 貴史	同上		H25.4～
今出 広明	同上		H25.4～H26.1
神林 亮	同上		H25.4～H26.1

研究項目

・通信バッファを削減した通信モデルにもとづいた通信プロトコル

③「通信路制御」グループ

研究参加者

氏名	所属	役職	参加時期
柴村 英智	公益財団法人九州先端 科学技術研究所	研究員	H23.10～
薄田 竜太郎	同上	研究員	H23.10～H25.3
曾我 武史	同上	研究員	H25.3～

研究項目

・実行時の状況に応じてパケット送信間隔を動的に制御する通信最適化技術

①「アプリケーション」グループ

研究参加者

氏名	所属	役職	参加時期
高見 利也	九州大学情報基盤研究 開発センター	准教授	H23.10～H27.9
深沢 圭一郎	京都大学学術情報メデ ィアセンター	准教授	H23.10～H27.9
稲富 雄一	九州大学情報基盤研究 開発センター	研究員	H23.10～H25.3
本田 宏明	同上	研究員	H23.10～H27.9
薄田 竜太郎	同上	研究員	H25.4～H27.9

研究項目

・スケーラブルな並列アプリケーション作成のための通信ライブラリ応用技術

(2)国内外の研究者や産業界等との連携によるネットワーク形成の状況について

なし

§3 研究実施内容及び成果

3.1 インタフェースグループ(九州大学)

(1)研究実施内容及び成果

・研究の概要

インタフェースグループでは、特に、プロセス間の同期を伴うパターン通信について、通信ライブラリへの実装技術の研究開発に取り組んだ。本プロジェクトの基盤通信ライブラリとして、主にプロトコルグループが中心となって開発した ACP (Advanced Communication Primitives) は、省メモリと低遅延の実現のために、プロセス間の同期を伴わない片側通信を基本インタフェースとしている。片側通信は、プロセス間通信におけるデータ転送と同期を分離し、個別に記述することが出来るため、データ転送時の通信バッファを必要としないことから、省メモリでの実装が可能となった。また、ACP では、プロセス間の同期を、データの生成と参照の依存関係に基づいて、必要最小限の通信で実現できるように並列プログラム中に記述することが出来る。一方、MPI (Message Passing Interface) の基本インタフェースとして広く利用されている、両側通信や隣接通信、集団通信といったパターン通信は、データ転送とプロセス間同期を合わせた機能を提供する。これらは、並列プログラムにおけるデータの生成と消費の依存関係を簡潔に記述できるインタフェースとして重要であり、エクサスケール環境でも多用されると考えられる。

そこでインタフェースグループでは、これらの同期を伴うパターン通信の省メモリ低オーバーヘッド実装技術の研究開発に取り組んだ。主な研究成果は、アプリケーショングループと共同で進めたチャンネルインタフェースとACP+MPI機能の ACPライブラリへの実装、マルチチャンネルインタフェースの設計、および、各パターン通信の動的最適化技術である。また、ACPライブラリのリリース毎のパッケージ開発と公開も、本グループで行っている。

・チャンネルインタフェース

チャンネルインタフェースは、送信プロセスによる send 関数と受信プロセスによる receive 関数による両側通信の機能を提供する。MPI の両側通信と異なり、送受信プロセス対毎に、チャンネルと呼ぶ通信路を明示的に確保、解放する関数を用意することにより、通信バッファのためのメモリ使用量の制御を可能としている(図1)。また、このチャンネルは片方向であるため、各プロセスは、チャンネルでの自分の役割に応じて、送信バッファもしくは受信バッファのいずれかを配置する。さらに、タグのような、out of order でメッセージを送受信するための機構を排除することで、省メモリ低オーバーヘッドの実装が可能となった。

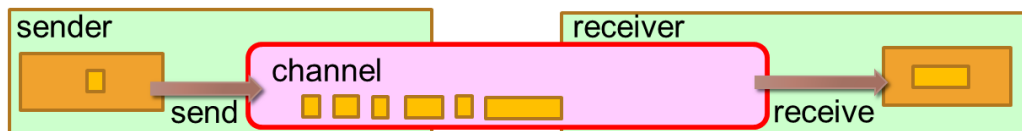


図1 チャンネルインタフェースの概要

チャンネルインタフェースの実装において、チャンネル確立の要求はチャンネルの送信側プロセスから受信側プロセスに送付する。そのため各プロセスは、不規則に到着する接続要求を受理するためのリングバッファを初期化時に用意する。このリングバッファは全プロセスからの要求を受け付けるため排他制御される。そのため、一つのプロセスに接続要求が集中した場合に性能低下が見込まれるものの、プロセス数に依存しないメモリ使用量で任意のプロセス間のチャンネル接続確立が可能となる。

通信プロトコルとしては Eager プロトコルを採用しており、送信バッファ、受信バッファとも一つ以上のスロットで構成する。スロットの大きさは送受信間共通とし、スロット数は送信側と受信側で個別に設定可能としている。スロットの大きさを超えるメッセージは、複数のセグメントに分けてパイプライン転送することにより、通信帯域幅の向上を図っている。図2に、スロットの大きさを変化させた場合のチャンネルインタフェースの通信時間と通信帯域幅を示す。スロットの大きさが 64KB で、十分な性能が得られることが分かる。

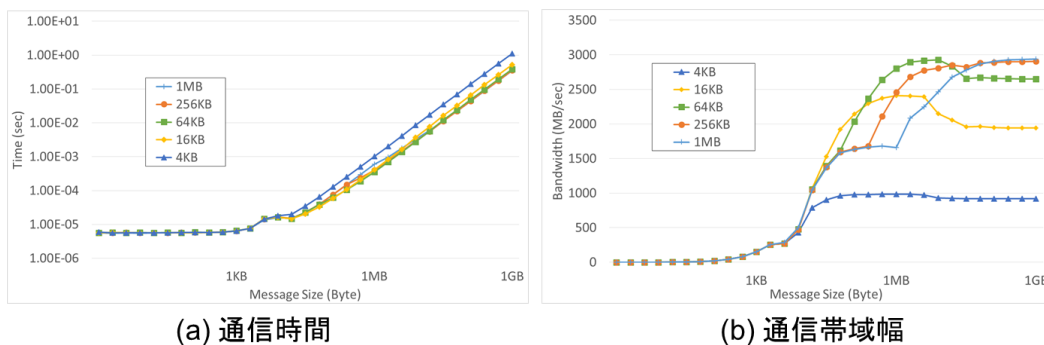


図 2 チャンネルインタフェースの性能 (InfiniBand 版)

・パイプライン転送インタフェース

ACPを用いたアプリケーションの通信効率化の取り組みの中で、隣接通信や集団通信などのパターン通信を計算と並行して進める通信隠蔽の効果を最大限に得るためには、送信領域と受信領域の片側通信領域としての事前登録、および、受信領域の上書きを許可する ready 通信が有効であることが分かった。また、これらのパターン通信では、pack 処理を含む通信や集団通信のツリー型アルゴリズムのようにパイプライン方式でデータをセグメント分割して処理することによって通信隠蔽率を上げ、スループットを向上できる場合が多い。一方、MPI には、これらを実現する通信インタフェースが定義されておらず、通信隠蔽を十分に行えない。そこで、パターン通信の効率的な通信隠蔽を可能とするため、パイプライン転送インタフェースを設計し、実装に取り組んでいる。

パイプライン転送インタフェースは、送信プロセスと受信プロセスに同じ大きさのバッファ領域を配置し、セグメント単位で送信側から受信側へデータを転送するための関数群を定義したものである。送信側は、セグメントの送信準備が終わる毎に ack 関数により受信側に転送する。受信側は、セグメントを受信後、そのセグメントが上書き可能となった時点で ready 関数により送信側に通知する。この他に、送信側で次の空きセグメントを問い合わせる関数や、受信側で新たに到着したセグメントを問い合わせる関数等を用意し、パイプライン転送を容易に記述できるインタフェースとしている。

・ACP+MPI 機能

前述の通り、現在の MPI の仕様では、通信の省メモリ化や隠蔽を十分に行うためのインタフェースが不足している。本プロジェクトでは、プロジェクト開始当初から、MPI の仕様を策定する MPI Forum に参加し、これらの問題についての議論に参加してきた。しかし、議論が収束して MPI の仕様に反映されるまでには、少なくともあと数年を要する。そこで本プロジェクトでは、現在の MPI の仕様のままで省メモリ化を達成する技術として、当初の計画で予定していた省メモリ MPI ライブラリではなく、ACP+MPI 機能を ACP ライブラリに実装した。MPI では、一つの MPI プログラムに参加するプロセス数に比例したメモリ消費が問題となっている。そこで、MPI プログラムのプロセスを複数のグループに分け、それぞれのグループ毎に別の MPI プログラムとして実行することで、グループ内の通信について、既存の MPI プログラムをそのまま利用しつつ、各プロセスのメモリ消費を削減することが出来る。一方、グループをまたがる通信は、ACP を用いることで、必要最小限のメモリで実現する。今回の実装では、既存の MPI ライブラリについては変更を加えず ACP 内部に MPI との連携機能を追加して、種々の MPI ライブラリと併用可能とした。ACP+MPI 機能を用いたプログラムの実行には、専用の macprun コマンドを用意した。このコマンドは、内部で ACP の起動コマンド acprun と MPI の起動コマンド mpirun を、使用する計算ノードの情報に基づいて適切に呼び出す。本実装の有効性を確認するため、アプリケーショングループの OpenFMO プログラムと通信パターンが同等なテストプログラムを準備し、ACP+MPI 機能を用いて実行した。また、同様のプログラムを、MPI の機能である MPI_Comm_split、および

MPI_Comm_spawn でそれぞれ作成し、メモリ消費量を最大 1025 プロセスにて比較した。その結果、メモリ消費量は ACP+MPI 機能によるものが最も少なく、本方法の有効性が確認できた。この ACP+MPI 機能の応用として、タスク並列型の MPI プログラム同士を接続し、タスク間を ACP の片側通信で記述する、等、既存の MPI プログラムの連結に利用することが考えられる。これにより、既存 MPI ユーザーに ACP 利用を促す方法の一つとなることが期待される。また、ACP+MPI 機能を利用すると、これまで MPI 並列で動作していたプログラムのメモリ内データを外部プロセスから ACP のインタフェースによって直接参照可能となるため、並列プログラムの可視化やデバッグなどにも応用可能である。

・動的最適化技術

同期を伴うパターン通信として、集団通信、隣接通信および両側通信のそれぞれを対象として、実行時に得られる情報に基づいた最適化技術を研究開発した。

集団通信の動的最適化技術：

集団通信は、科学技術計算など、様々な並列プログラムにおいて多用されており、特に大規模な並列計算機における高速化が、アプリケーションのスケラビリティ向上に向けた重要な課題となっている。集団通信には複数の実装手段があり、しかも多くの場合は、どのような状況でも必ず最適であるチャンピオンのような実装が存在しない。そのため通信ライブラリは、内部で複数の実装手段を選択肢として用意し、実行時にそれらの中から一つを選択する。従来の通信ライブラリでは、この選択を行う際に参照可能な情報としては、システムの基本的な情報とプログラム中で通信命令に渡されたメッセージサイズ等のパラメータのみであったため、資源利用効率まで考慮した積極的な最適化を適用できなかった。そこで本グループでは、実行時のプロセスの配置情報や負荷の状況、プログラム全体の構成や内部での通信の行われ方等の付加的な情報を最大限に活用し、最適な実装手段を選択する積極的な最適化技術に取り組んでいる。特に初年度と 2 年度では、ランク配置による通信性能の変動を考慮したアルゴリズム選択技術を提案し、評価した。具体的には、実行前に取得しておいたネットワークのトポロジー、ルーティング情報と、実行時に得られるランク配置の情報から、通信時の平均的なバンド幅を計算し、その値を用いて各アルゴリズムの性能を予測する。この予測結果を基に、他のアルゴリズムよりも明らかに遅いと予測されたものを選択肢から除外し、残ったアルゴリズムについて実際に実行中に一つずつ試して最速のアルゴリズムを選択する。実験の結果、提案する手法により、オーバーヘッドを低く保ちながら高い精度でアルゴリズム選択が行えることを確認した。

隣接通信の動的最適化技術：

計算機の大規模化に伴い、システム全体での通信に要する時間が問題となることから、プログラムの局所性を活かした隣接通信がより重要性を増すと考えられる。一方、大規模並列計算機のインターコネクトとしては、メッシュ/トーラスのような直接網で、各ノードに複数の通信デバイスを搭載するものが採用されることが増えてきている。隣接通信では、これらの通信デバイスを効率的に利用することで通信性能が向上できる。しかし、このようなネットワークの特徴を考慮した隣接通信アルゴリズムの研究は、あまり進んでいなかった。そこで、我々は、複数の通信デバイスを効率良く利用できる隣接通信アルゴリズムの開発を行った。

隣接通信は、アプリケーションにより隣接数が決まる。隣接数と通信デバイス数が同一であれば、通信デバイスを効率良く利用した高速な隣接通信が実行できる。一方、隣接数と通信デバイス数が異なれば、図3のように通信デバイスを効率的に利用できなくなる。そこで、我々は、メッセージを隣接数と通信デバイス数に応じて、セグメント分割して通信デバイスに割り付ける通信アルゴリズムを提案、実装した。これにより通信デバイスを効率良く利用でき、隣接通信の通信性能が向上する。

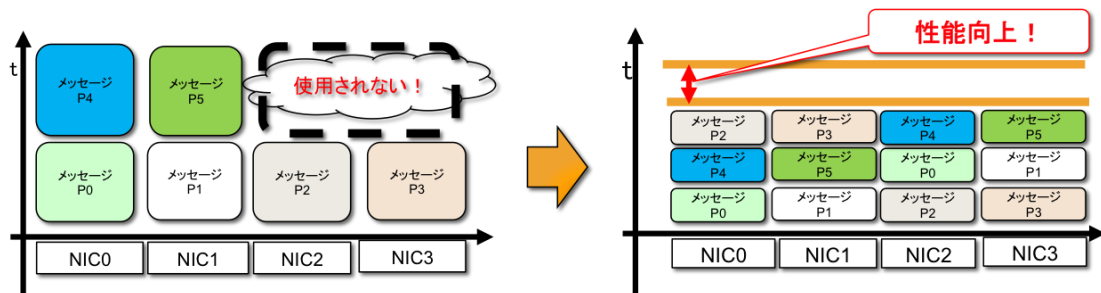


図3 隣接通信アルゴリズムによる性能向上の概要

この隣接通信アルゴリズムを、富士通拡張 RDMA インタフェースを用いて実装した。これは、このインタフェースがどの NIC を利用するか直接指定できるからである。実装した隣接通信アルゴリズムを用いて FX10 上での提案隣接通信アルゴリズムの性能評価を行った。図 4 に示すように実測では、既存の隣接通信に対して最大で25%の性能向上を得た。

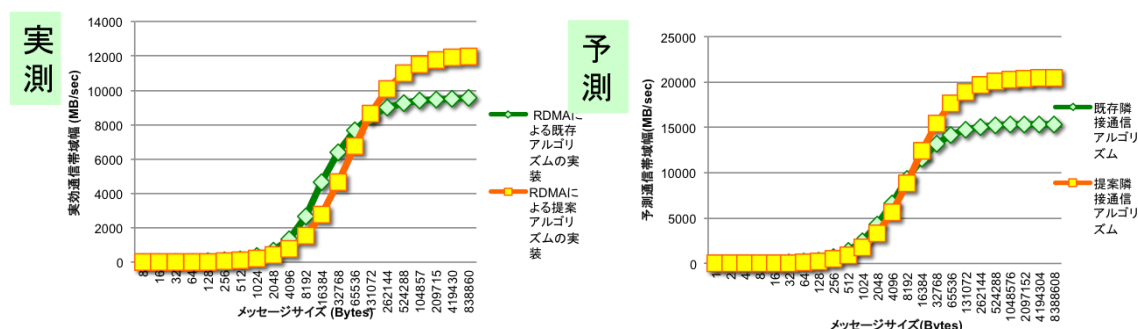


図 4 隣接通信アルゴリズムの通信性能の実測および予測

これは、作成した通信モデル式と同等であり、通信モデル式の精度が一定以上であることを示す。一方、図2で示すように通信性能の入れ替わるメッセージサイズは通信モデル式とずれており、自動最適化に向けて通信モデル式の精度向上が必要となる。

両側通信の動的最適化技術:

本グループが実装したチャンネルインタフェースは、プログラマがチャンネルの生成と削除を明示するインタフェースであった。一方、MPI のように、通信路の生成と削除を明示しないインタフェースは、プログラマの負担を軽減する上で有効である。そこで本研究では、両側通信において、2 プロセス間的高速通信に使用されるバッファの使用量を通信状況に合わせて動的に増減する技術を研究開発している。この技術は、通信状況の把握(Profile)、高速通信対象及び非対象の選択(Optimization)及び、プログラム実行中の 2 プロセス間通信の接続/切断(Application)の各機能による実装する。通信が行われると、Profile 部が持つ通信履歴情報を更新する。この時高速通信接続数が閾値を超えた場合には、Optimization 部において、直近での通信頻度を基に、バッファを開放すべき 2 プロセス間通信が選択される。Application 部では、選択された 2 点間通信の切断処理とバッファの開放処理および、新たに高速通信を行う場合には、バッファの割り当て処理と 2 点間通信の接続処理が非同期に実行される。本技術による両側通信インタフェースを、ACP ライブラリを使用して実装した。本研究では、2 プロセス間通信の通信順序の履歴をとることにより、実行中に通信頻度が低下した 2 プロセス間通信の高速通信を切断してバッファを開放して、そのバッファを新たに通信頻度が上昇した 2 プロセス間通信に割り当てて、新たな高速通信を実施可能にしている。一方、従来の MPI の実装として Open MPI 2.0.0 を解析したところ、通信回数が

閾値を越えた通信は全て高速通信に移行するため、使用頻度が低く全体性能に影響を与えない通信であってもバッファが割り当てられる。また、使用頻度が入れ替わる両側通信の間でバッファをシェアするようなこともできないので、本実装と比較した場合、性能に比したメモリ使用量は大きくなる。

• ACP ライブラリ公開と普及に向けた活動

本グループで行った、ACP ライブラリのパッケージングと公開について以下に纏める。ACP ライブラリのようなシステムソフトウェアは、開発後の一般利用とその後のメンテナンスを十分に考慮して開発設計する必要がある。本研究グループでは、Open Source ディストリビューションのデファクトスタンダードに沿って ACP ライブラリを configure; make; make install で利用可能な構造にするために、autotoolize (autoheader、 automake、 autoconf、 libtoolize) を用いてパッケージングを行った。autotoolize により、Multi MPI への対応も、組み込みの MPI の種類にユーザーが意識を払うことなく利用できるようになっている。ACP ライブラリ自体の構成は、インターコネクトのデバイスとして Tofu、InfiniBand、Ethernet などに対応させるために、デバイス依存の基本層 (Basic Layer) と非依存の中間層 (Middle Layer) をそれぞれ独立したライブラリファイルとして分離している。

ACP ライブラリの開発は、グループをまたがる複数の研究者が共同して行うため、「ACP ライブラリ開発と公開のための手順」を策定して、subversion の設置と利用方法から、開発・フィーチャーフリーズ・コードフリーズ・リリースといった、リリースまでの大まかな手順までマニュアル化した。この手順書では、ACP ライブラリ公開後の告知やユーザーからのバグ報告を含むフィードバックを得る方法も策定した。具体的には Mantis を用いたバグ管理システムとユーザーとのコミュニケーション窓口を設置した。

2014 年 10 月 15 日の version 1.0.0 リリースから、ほぼ半年ごとにリリースを行い、2017 年 4 月に version 2.2.0 をリリースする。また、プロジェクト終了後の開発継続を考慮して、よりオープンに開発をすすめることの出来る GitHub への移行を進めている。

また、ACP ライブラリの普及活動の一環として、ACP ライブラリの省メモリ性を紹介するため、Raspberry Pi クラスタ上で動作させる実証実験を行った。実験に利用した Raspberry Pi 2 Model B (図 5) はメモリが 1GB と比較的少なく、ポストペタスケールの計算機で発生するメモリ量が枯渇した状況を模擬できると考えた。また、ARM プロセッサを搭載しており、ACP ライブラリのポータビリティを示すこともできる。



図 5 ACP ライブラリが動作する Raspberry Pi クラスタの仕様構成

実験結果としては、Raspberry Pi 2 Model B の4ノードクラスタ上での ACP ライブラリの動作が確認され、そのポータビリティが示された。また、分子起動計算プログラムを4ノードクラスタ上において16プロセスで実行した。この時、Open MPI とのメモリ使用量の比較を行った。評価結果としては、ACP ライブラリが 5,788KB、Open MPI が 10,488KB となり、メモリ使用量が半分程度となることを示した。

さらに、既存のミドルウェアへの ACP ライブラリ適用によるスケーラビリティ向上を目的として、

九州工業大学の八杉研究室と京都大学の平石研究室で開発されているタスク並列言語 Tascell の言語処理系への ACP 適用の可能性を検討した。Tascell の言語処理系は、現在 MPI の両側通信で実装されているため、同期通信が必要となっているのに対し、ACP の片側通信を利用することにより、例えばタスクが使用するデータの get 通信を、他の処理と並行して行えるタイミングで発行する、等の効率化が行えそうであることを確認した。この研究は、来年度以降も、南里が両研究室と共同で進めることにしている。

3.2 プロトコルグループ(富士通株式会社)

(1)研究実施内容及び成果

初年度から6年度10月まで、進捗に影響を与える問題も発生しておらず、当初計画通りに研究を進めることができています。中間報告で指摘や助言を受け、4年目以降は実際のアプリケーションにおけるACPの有効な利用方法を広げる方向で研究開発を展開した。以下に年度毎の研究実施内容及び成果を述べる。

初年度(平成23年度)は、「基礎的な片側通信ライブラリの確立」を目標に、既存の通信ライブラリの評価、グローバルデータ構造の効率的な操作方法の検討を行った。

既存の通信ライブラリの評価としては、省メモリ化のための研究の調査と、既存のライブラリでの実行評価を行った。調査の結果、数千並列程度までのメモリ消費は問題ないレベルであるが、来るエクサ時代に想定される数千万プロセスの環境ではプロセスごとに数十GBのメモリを消費し、そのままではアプリケーションの動作が不可能なレベルであることを確認した。

グローバルデータ構造の効率的な操作方法の検討としては、片側通信と遠隔 Atomic 操作を用いた操作方法を検討し、実際のハードウェア上で動作させた場合の性能の予測を行った。その結果、遠隔 Atomic 操作を用いることで、Put、Get の単純な片側通信のみの場合に比べて、100万プロセス規模で最大1000倍程度の性能改善が得られることを明らかにした。また、遠隔 Atomic 操作をグローバルデータ構造の操作に用いる場合、ハードウェアによる順序保証サポートが性能に大きく寄与することを明らかにした。

2年度目(平成24年度)には、「メッセージパッシング通信プロトコル低遅延・省メモリ化技術の研究開発」を目標に、通信ライブラリのメモリ使用量を定量的に評価する方法の確立、通信バッファの削減方式や通信データの局所的な集中に対する性能改善の方式検討、複数プロセスのメモリに配置したグローバルデータ構造を主たるデータ格納先として利用する場合のメモリ管理手法の検討を行った。

通信ライブラリのメモリ使用量を定量的に評価する方法の確立としては、アプリケーションやライブラリに変更を加えることなく、動的メモリ使用量をライブラリやその内部関数毎に測定・集計可能な測定ツール DMATP-MPI を提案および実装し、既存の MPI ライブラリ(Open MPI および MVAPICH MPI の一部)の主要な関数について、プロセス数に対するメモリ使用量の変化を調査した。その結果 MPI_Init 関数におけるプロセス数に比例したメモリ使用量の増加、Unexpected Message のメモリ開放処理の問題などを検出し、対策が必要との結論を得た。

通信バッファの削減方式や通信データの局所的な集中に対する性能改善の方式検討としては、「受信バッファ獲得方式」、「送信メッセージ数制限方式」、「遠隔 Atomic 通信を用いた受信バッファ固定方式」、および「遠隔 Atomic 通信を用いた受信バッファ数制限方式」による通信バッファの削減、およびサーバーへのアクセス数を一定以下に抑える機能をインターコネクタの構成に合わせて分散配置する方式による通信データの局所集中に対する性能改善を検討した。

複数プロセスのメモリに配置したグローバルデータ構造を主たるデータ格納先として利用

する場合のメモリ管理手法の検討としては、非同期にローカルノード、リモートノードから動的に変更できる非同期グローバルヒープの提案と、インターコネクットの通信順序保証や CPU/インターコネクット間の Atomic 操作の不可分性の有無による性能評価・比較を行った。

3 年度目(平成 25 年度)は「片側通信によるデータ構造遠隔操作技術の確立」を目標に、ACP ライブラリの基本コンセプトを設定し、全体構成および ACP 基本層の詳細検討、仕様策定、ならびに実装を行った。

ACP ライブラリは、ACP 基本層、チャンネルライブラリ、データライブラリから構成する。ACP 基本層は、インターコネクットデバイスの機能を抽象化したグローバルメモリ管理、グローバルメモリアドレスを基にしたデータ参照と Atomic 操作を行う。チャンネルライブラリは、ACP の設計方針である「メモリ使用を明示的に制御する」という考え方にに基づき ACP 基本層上にストリーム転送の使用メモリを最適化するインタフェースを提供する。データライブラリは、グローバルメモリ上での大域的なデータ配置を最適化し、データ転送のオーバーヘッドが小さいプロトコルとアルゴリズムを提供する。ACP 基本層は、初期化、終了処理や総プロセス数、プロセス番号取得などの共通機能を提供するインフラストラクチャ関数、メモリの登録やグローバルアドレス変換を行うグローバルメモリ管理関数、グローバルアドレスを指定してメモリを参照するグローバルメモリ参照関数で構成される。ACP 基本層は並列データ構造の実装に適した仕様となっており、グローバルメモリ参照機には、読み出し側もしくは書き込み側のプロセッサがデータ転送を制御する必要がある片側通信ではなく、送信元でも宛先でもない第三者プロセスがメモリツーメモリコピーを制御するグローバルメモリ参照(Global Memory Access、GMA)を導入し、グローバルアドレス空間には、グローバルメモリへのポインタに対する Atomic 操作が可能となるように、64 ビットのフラットなアドレス空間を導入した。本仕様は UDP スタック上と「京」などに採用されている Tofu インターコネクット上に実装した。UDP は標準化されたソフトウェアによる汎用通信スタックであり、Tofu は独自仕様の RDMA 機能を持つ高性能通信ハードウェアである。このように、性質、機能が全く異なる通信デバイス上に ACP 基本層を問題なく実装できることを確認した。また上位プログラムからも通信デバイスの性質の違いを意識しないプログラミングが可能であることを確認した。

4 年度目(平成 26 年度)は「PGAS 言語向け分散共有配列ライブラリの低遅延・省メモリ化技術の研究開発」を目標に、データライブラリの詳細検討、仕様策定、ならびに実装を行った。また、スクリプト言語ベースの PGAS 言語ライブラリの仕様の検討と実装を行った。さらに、新しいインターコネクットである Tofu インターコネクット 2 への ACP 基本層の対応を行い、ライブラリの対応プラットフォームを拡充した。

データライブラリの詳細検討、仕様策定、ならびに実装としては、データ構造を操作するアルゴリズム自体を変えずに、配置指定の変更だけでグローバルなデータ配置の最適化を可能にする仕様を目標とした。そのため、データ生成時に配置を明示的に制御することで、データ構造を複数プロセスに分散させる。データの生成、操作、破棄は非同期に、配置するプロセスと同期せずに行う。さらに、データがローカルに配置されている場合、通常のローカルなデータ構造の操作と比較して遜色のない性能を目指す。基本層のインタフェースでグローバルアドレスを取得するために登録できるメモリはローカルプロセスのメモリだけである。しかしリモートプロセスに動的にデータを生成するためにはリモートプロセスのメモリを確保する必要がある。これを行うのがグローバルメモリアロケータである。ACP のグローバルメモリアロケータでは、2 年目に研究開発を行った非同期グローバルヒープより高度なメモリ管理アルゴリズムを持つ実装を導入する。グローバルメモリアロケータではブレイクポイントの操作でメモリ割当、解放を行うのではなく、メモリ割当と解放に別のインタフェース、具体的には `acp_malloc` および `acp_free` 関数を用意する。割り当てたメモリ量などの管理情報は利用者からは隠されて管理されており、割り当てたメモリを解放する際は該当のグローバル

アドレスを指定して `acp_free` 関数を呼び出すだけで良い。データ構造の型は C++ 言語の標準テンプレートライブラリ(Standard Template Library、STL)を参考に、可変長一次元配列 `vector`、双方向リンクリスト `list`、双方向キュー `deque`、連想配列 `map`、集合 `set` をサポートする。また、インタフェースも同様に STL を参考とするが、ACP ライブラリは C 言語のライブラリであるので、演算子は全て関数として実装され、イテレータは元のデータ構造への参照も引数として求められる場合があるなど、やや複雑なインタフェースとなっている。ただし上位層の言語処理系などでラップされることを想定して、データ構造間でインタフェースの直交性が高くなるように考慮される。設計した仕様および性能を評価するため、ACP 基本層上にグローバルメモリアロケータを実装し、UDP スタック上および Tofu インターコネクト上で初期評価を行った。その結果、`acp_malloc` および `acp_free` 関数の平均実行時間は割当先 `local` の場合、Tofu 版で 9 及び 19 μ 秒、UDP 版で 16 および 26 μ 秒、割当先 `remote` の場合、Tofu 版で 34 及び 68 μ 秒、UDP 版で 2.0 および 7.9 ミリ秒となり、低遅延なインターコネクトでは割当先 `remote` の場合でもローカルと遜色ない性能を持つライブラリが確立できた。また、本アロケータをベースに `vector` および `map` を実装し性能を評価した。その結果、リモートの要素の挿入・検索ともに約 1 ミリ秒以下の性能となり、サーバークライアント型のインメモリ分散 KVS と同等であった。これは、ACP のデータライブラリが、HPC ののみならずビッグデータの処理基盤としても有用であることを示している。

スクリプト言語ベースの PGAS 言語ライブラリの仕様の検討と実装としては、Ruby および Python から ACP 基本層を使用して、比較的シンプルな仕様である Co-Array Fortran に準じたモデルを実装し、PGAS 言語向け分散共有配列ライブラリに適用可能な技術を確認した。PGAS 言語は既存の高いパフォーマンスを持つ言語に新たな言語機能を追加することで並列計算の記述の生産性を高める言語、および最初から並列計算を記述することを考慮に入れ全く新規に設計した言語に大別されるが、それぞれ互換性や学習コストに問題を抱えている。一方、Ruby や Python といったスクリプト言語は学習しやすい文法および強力なメタプログラミング機能を持つため、DSL (Domain Specific Language) の記述能力が高く、言語自体の文法に拡張を加えなくても新規キーワード相当の機能や言語機能の挙動の変更を実現できる。このため、互換性を保ったまま新しい言語の概念を導入しやすい。実装した実行モデルはデータへのアクセスがプロセスのランク番号と各ローカル配列内の添字に基づいて行われるローカルビューを持ち、プログラムの実行形態は SPMD (Single Program Multiple Data) であり、通信は片側通信によって行われる。各プロセスには均等な長さのローカル配列が確保され、あるプロセスは他プロセスに存在するローカル配列をリモートから読み書きすることができる。PGAS 言語を実装するにあたっては、通常は分散配列への添字を用いた参照や書き込みがそれぞれ片側通信の `Get` と `Put` に対応付けられるが、このような実装では、ACP で記述することのできるリモート間通信が実現できないという問題がある。我々は、プロキシオブジェクトによる評価順の制御により、リモート間通信を記述できることを示した。

ACP 基本層の Tofu インターコネクト 2 への対応に関しては、対応プラットフォームの拡充の他、ACP ライブラリが想定した機能を一部搭載したハードウェアが登場したため、その機能が実際に有用であることを確認する意味合いで研究開発を行った。Tofu インターコネクト 2 は不可分操作機能を有し、かつそれがインターコネクトとプロセッサの緊密な連携のお陰で相互に排他制御ができるという特徴がある。ACP 基本層のリモート側不可分操作機能を Tofu2 の機能を用いてハードウェア化し、その性能を測定した結果、5 マイクロ秒のレイテンシ改善が達成できた。

5 年度目(平成 27 年度)は「分散データ構造ライブラリの整備」を目標に、データライブラリ性能の改善を目的とした ACP ライブラリの整備を実施した。

初期実装におけるグローバルメモリアロケータは、各プロセスの空きグローバルメモリをアドレスでソートした片方向リスト、フリーリストで保持しており、メモリ割当は典型的には計算量が $O(1)$ で高速である一方、解放は隣接する空き領域をマージするため、計算量が $O(n)$ と空きメモリ領域数に比例していた。ACP では使用しないメモリを解放することによる省メモリ化を推奨しているため、メモリ解放のコストが高いことは望ましくない。そこでメモリ解放時はフリーリストではなく隣接領域を直接調べて空き領域であるかどうか判定するアルゴリズムに改良し、計算量を $O(1)$ に削減した。また、データ構造操作関数を高速化する方式を提案、評価した。UDP 版基本層では宛先と開始元が同じプロセスの場合も UDP でデータ転送していたが、CPU コピーに改良した。またコマンド実行順序は開始元で制御していたが、一部を宛先に制御するように改良した。データ構造操作ではデータ移動回数の多い省メモリアルゴリズムを使用していたが、一時バッファ併用に改良した。

6 年度目(平成 28 年度)は「集団演算等の機能拡張、および性能解析にもとづいた改良」を目標に、データライブラリの機能整備、UDP 版 ACP 基本層の性能改善、生命情報科学アプリケーションへの ACP の組み込みを行った。

データライブラリの整備としては、データセット全体の融合、交換、移動などの操作を追加した。また、生命情報科学アプリケーションでの利用を想定して、連想配列 map、集合 set を細粒度の排他制御を行う並列データ構造に改良するとともに、排他制御の少ない高速なイテレータ参照を可能とするための仕様変更を実施した。

UDP 版 ACP 基本層の性能では、遠隔順序制御の導入、ノード内複数プロセス時のノード間通信性能の改善、および自プロセス内のメモリコピー性能の改善を行った。遠隔順序制御の導入では、宛先プロセス上における連続メモリコピーの順序制御を開始元プロセスではなく宛先プロセスで行う方式により、vector および deque の要素削除、挿入操作のオーバヘッドを $1/5$ 以下に削減した。ノード内複数プロセス時のノード間通信では、1 ノードあたり 1 つのゲートウェイプロセスのみノード間通信を行う新方式を提案、実装し、輻輳を回避した。自プロセス内のメモリコピー性能では、通信スレッドではなくメインスレッドが memcpy を行う方式を導入したが、これでは通信性能が改善しなかった。さらに通信スレッドの通信コマンド実行を制御する状態変数をメインスレッドが書換えることでメモリコピー性能を改善した。

生命情報科学アプリケーションへの ACP の組み込みとしては、九州大学医学部大川研究室の協力により、生命情報科学研究で広く使用されており、使用メモリ容量が課題となっている既存アプリケーションである Trinity を選定した。Trinity の中で特にメモリ容量を必要とするのはゲノムシーケンサーの出力から辞書を作成し、ゲノム配列を復元する Inchworm 処理である。ここで Inchworm に ACP を組み込み、データライブラリを使用して複数ノードのメモリ上に辞書を配置した。また、ACP の分散データ構造と Python の豊富な科学計算ライブラリの両方を利用可能な例として、Inchworm を Python に移植した。この際、Inchworm の実装を分析し、本アプリケーションを高速化するのに役立つデータ構造として、multiset を ACPd1 に追加した。また、大川研究室の指導の元、実際のバイオインフォマティクスの研究に用いられる巨大なサイズのファイルを用いて移植プログラムの試験を行い、大規模な問題サイズでしか検出できない障害を ACP に複数発見・修正した。この取り組みの結果、ACP の品質と機能を向上させることができた。

3.3 通信路制御グループ(公益財団法人九州先端科学技術研究所)

(1) 研究実施内容及び成果

通信路制御グループでは、プログラム実行時のネットワークの状況に応じて、パケットの送信間隔を調整し、通信レイテンシの削減と通信スループットの向上を実現する、通信最適化技術の研究を行った。また、エクサスケール級システムにおける本最適化技術の有効性及び当該課題で採用する RMA 通信性能を評価するために、アプリケーションの性能推定環境の整備を行った。

近年のマルチ/メニーコア CPU をはじめとする諸ハードウェアの高密度実装技術の進歩にともない、今後、計算ノード内のプロセス数は加速的に増加することは明らかである。その結果、エクサスケール時代では、現在よりも 1 ノードあたりのノード間通信数が増え、通信リンク 1 本が担う通信数も増えるため、通信混雑や輻輳が発生する頻度が今以上に増すことが考えられる。これをハードウェアで回避するには、リンク数を増やすかリンク帯域を向上させるしか方法が無い。さもなければ、なんらかの方法で通信混雑や輻輳の発生を抑制しなければならない。

一方、計算機の性能や規模の向上にともない、これからもユーザーが要求・実行するアプリケーションの並列度(利用コア数、利用ノード数)は確実に増加する。本プロジェクトで想定する多次元の直接網では、ノード数が大きくなるにつれて通信ホップ数が増えるため、大規模システムではホップ遅延をはじめとする通信遅延を極力小さくすることが重要である。ホップ数の増加により、通信混雑や輻輳が発生する頻度もさらに増すことが予想できるが、いくら通信リンクが高速になったとしても、ひとたび輻輳が発生すると代償として通信遅延は一気に大きくなる。

そこで、本研究では、通信混雑や輻輳の発生、および通信遅延の増加を抑制するために、ネットワークの状況に応じて適切なパケット送信間隔で通信を行うパケットペーシングに着目した。そして、時々刻々と変化するネットワークの通信混雑情報から最適なパケットペーシングを行う、「パケット送信間隔動的最適化技術」の開発を目指した。さらに、このような動的最適化技術の評価、ならびに当該課題で開発する ACP ライブラリを用いたアプリケーションの評価を行うための「エクサスケール環境のアプリケーション性能予測技術」の開発を行った。

初年度(平成 23 年度)は、既存の集団通信アルゴリズムを対象に通信衝突を緩和するパケットペーシングの効果についてシミュレーションによる基礎的な評価を実施した。様々な実行環境・状況を対象に、パケットの送出間隔(以下、パケット間ギャップ)を制御するパケットペーシングを適用した集団通信の基本的な性能について調査した。具体的には、2 次元トラス網、ならびに 3 次元トラス網において、集団通信アルゴリズム、ノード数、メッセージサイズ、パケット間ギャップ値をそれぞれ変化させ、集団通信に要する時間をはじめとする各種統計値を、インターコネクトシミュレータ NSIM を用いて調査した。その結果、一般的な集団通信にパケットペーシングを適用した場合の有効性を確認するとともに、アルゴリズム、トポロジー、ノード数、メッセージサイズに応じて、集団通信を最適に実行するパケットの送出間隔が異なることが分かった。また、通信ホップ数に応じたパケット間ギャップを設定することで、アルゴリズムによっては大幅な実行時間の短縮が達成されることを確認した。次に、次年度に実施するパケットペーシングのモデル化への指針を得るために、一部の集団通信アルゴリズムについて実行時間を最小化させるパケット間ギャップ値の導出手法について検討した。同時期に通信リンクを流れようとするメッセージの数、すなわちリンクあたりのメッセージ重複数に着目し、通信ステップ毎に最適なパケット間ギャップのモデル式を設計するとともに、NSIM によるシミュレーション評価を行った。その結果、モデル式に基づくパケットペーシングを行うことで、8K ノードの 2 次元、3 次元トラス網において、Pairwise exchange アルゴリズムでは約 2 倍速くなることがわかった。また、ノード数が大きくなるにつれて、パケットペーシングによる速度向上率も増加することを明らかにした。

2 年度(平成 24 年度)は、隣接・集団通信におけるパケットペーシングの評価とモデル化を実施した。隣接通信は近隣ノード群との 1 対多あるいは多対多通信であり、科学技術計算

アプリケーションに多く利用されているものの、多くのHPC向け通信ライブラリでは規定されていない。そこで、実践的なアプリケーションから利用頻度が高く、効果的な隣接通信パターンの調査を行うとともに、本研究で開発する通信ライブラリへの実装方針ならびに汎用性のあるユーザーAPIについて検討した。次に、パケットペーシングを用いた集団通信に対するインバランスの影響を調査した。負荷不均衡(ロードインバランス)や通信不均衡(コミュニケーションインバランス)によって、パケットペーシングの効果が抑制される場合があるためである。その結果、集団通信のアルゴリズムによってインバランスの感受性が異なることがわかった。また、集団通信に対するペーシングの有効性を確認するとともに、メッセージサイズやノード数の増加に応じて実行時間の高速化率も向上することが確認された。さらに、ペーシングを適用した集団通信にインバランスが及ぼす影響を評価した結果、通信アルゴリズムによっては、わずかなインバランスが加わることで実行時間が大幅に増加し、ペーシングの効果を損なう場合があることが明らかになった。このように、通信衝突を緩和するパケットペーシングの効果について、ロードインバランスやネットワークインバランスが与える影響をシミュレーションによって明らかにした。

3年度(平成25年度)は、実機によるパケットペーシングの有効性評価、ネットワーク混雑情報収集技術の研究、エクサスケール級アプリケーションの性能推定環境の整備を行った。実機におけるパケットペーシングの有効性を実証することを目的とし、既存のHPCシステムによる検証実験を行った。具体的には、パケットの送出間隔を制御できる富士通社製「PRIMEHPC FX10」(以下、FX10)を利用して、ランダムリング通信と全対全通信にパケットペーシングを適用した場合の通信性能を調査した。ランダムリング通信では、実機におけるパケットペーシングの効果を実証するとともに、メッセージ長やノード数が増加した場合におけるペーシング効果の向上について確認した。また、全対全通信では、FX10のインターコネクトであるTofuを駆動する専用ライブラリを利用し、FX10での実機評価、ならびにインターコネクトシミュレータNSIMによる評価結果との比較を行った。その結果、これまでのシミュレーション評価で認められてきたパケットペーシングの有効性をはじめ、メッセージ長やノード数に応じたペーシング効果の向上が実際に実システム上で確認された。また、ネットワーク混雑情報収集技術の研究については、前年度に引き続き、全体の通信スループットを低下させない、効率の良い通信混雑状況の表現手法、ならびに多数のノードで協調しながら通信混雑情報を分散管理する手法について検討した。時々刻々と変化するネットワークの状況を隣接するルータ間で伝達しあい、適切な領域内に混雑情報を拡散させる機構を、従来の通信データに加え、フリットあるいはパケットに搬送させて送受信させるアイデアを検討した。

エクサスケール級システムにおけるアプリケーションの実行性能を詳細に評価するためには、ノード演算性能の推定に加え、通信衝突によって発生する通信レイテンシを含めた通信時間の推定が重要となる。通信衝突の様相は、システムのネットワークアーキテクチャやアプリケーションの通信パターンに応じて異なり、通信衝突によって通信タイミングも変化するため、複雑なアプリケーションについては通信状況の把握や机上での通信時間の予測が困難な場合が多い。そこで、衝突も含めたエクサスケール級の通信を模擬し、システムの仕様や通信パターンに則した実行時間を算出するインターコネクトシミュレータNSIMを核としたエクサスケール級アプリケーションの性能推定環境の整備を行った。

4年度(平成26年度)は、混雑状況に応じた最適なパケット送信間隔導出技術、ならびにエクサスケール環境のアプリケーション性能予測のためのシミュレーション技術の研究開発を行った。昨年度に引き続き、実機におけるパケットペーシングの有効性を実証することを目的とし、既存のHPCシステムにおける評価実験を継続して行った。ランダムリング通信にパケットペーシングを適用した場合の通信性能について、大規模なノード数による評価実験を行い、実機におけるパケットペーシングの効果を実証するとともに、ノード数が増加した場合におけるペーシング効果の向上について確認した。その結果、ノード数が増加する

とともに、これまでに確認できたパケットペーシングの有効性をはじめ、メッセージ長やノード数に応じたペーシング効果がさらに顕著なることを確認した。

当該課題で開発を進めている ACP 通信ライブラリは、エクサスケール時代に向けた低遅延、省メモリな低レベル通信ライブラリを目指し、その基本層は RMA 通信を直接操作する機能をサポートしている。従来の大規模インターコネクトシミュレータは、インターコネクトや通信ライブラリの実装を行うシミュレータユーザに使いやすい形で RDMA 通信を直接シミュレーションする機能を持っているものは皆無である。そこで、従来のメッセージパッシング通信向けに開発されたインターコネクトシミュレータ NSIM を拡張し、ユーザーに使いやすい形で RMA 通信をサポートした NSIM-ACE を実装した。また、基本的な RMA 通信について NSIM-ACE によるシミュレーションと実機との比較評価を行い、今後の ACP アプリケーションの性能推定に向けて良好な精度を達成していることを確認した。

5 年度(平成 27 年度)は、実機によるパケットペーシングの有効性検証について、昨年度よりもさらに大規模なノード数での評価や実践的なアプリケーションを対象とした評価を行った。また、本プロジェクトで開発を進めている ACP 通信ライブラリを用いたアプリケーションの性能推定を行うために、NSIM-ACE を用いて実アプリケーションを対象とした RMA 通信の性能推定を行った。

実機におけるパケットペーシングの有効性を実証することを目的とし、昨年度に引き続き既存 HPC システムにおける評価実験を行った。富士通社製 PRIMEHPC FX10 を利用して、昨年度よりもさらに大規模なノード数について、ランダムリング通信にパケットペーシングを適用した場合の通信性能を調査した。その結果、実機におけるパケットペーシングの効果ならびにノード数が増加した場合のペーシング効果のさらなる向上について確認した。一方、実機においてパケットペーシングの効果が表れないケースも観測されたため原因を調査したところ、アプリケーションが通信していない場合にも、システムの管理・保守のための通信が多数発生していることが確認された。このような通信はパケットペーシングの効果を大きく低下させるため、通信時間を最適化するパケット間ギャップを導出するには、熟慮されたペーシング手法が必要であることが認識された。

また、実用的なアプリケーションにパケットペーシングを適用した場合の通信高速化の可能性を探るために、全球雲解像モデル NICAM を対象にシミュレーションによる通信性能の推定を行った。具体的には、NICAM の複数の主要通信部分についてパケットペーシングを適用し、シミュレーションによって各通信における最適なパケット間ギャップを調査し全通信時間を推定した。その結果、通信衝突によって増加した通信時間が大幅に削減され、理想通信時間に迫るペーシングの効果を確認した。

エクサスケール環境におけるアプリケーションの性能予測に向けて、NSIM-ACE の整備を継続した。本研究課題で開発している ACP 通信ライブラリの性能評価ならびに、当該ライブラリを用いたアプリケーションの性能推定やスケラビリティを詳細に評価するために、ACP 通信ライブラリを実装したシステムと NSIM-ACE に入力するシミュレーションパラメータとの較正(キャリブレーション)を実施し、実機と NSIM-ACE との精度誤差を削減した。また、ACP 通信関数を用いて開発された実践的なアプリケーションについて NSIM-ACE によるシミュレーション評価を行った。具体的には、重力多体問題シミュレーションの領域分割時に発生する粒子データ通信について、NSIM-ACE を用いたシミュレーションと実機との精度を調査し、およそ 10% 以下の誤差に留まることを確認した。

最終年度(平成 28 年度)は、ACP 通信ライブラリを用いた大規模アプリケーションの性能推定精度をさらに向上させるために、実在する大規模並列システムと同等のルーティングポリシーを設定できるよう NSIM-ACE を拡張した。

NSIM-ACE は、基本的なフルバイセクションバンド幅の FatTree をシミュレーション可能であるが、実機システムでは運用コストや全体性能を鑑みて上位層のスイッチ数を減らす、あるいは、スイッチ間の接続リンク数(バンド幅)を変更していることが多い。したがって、

NSIM-ACE によるアプリケーションの性能推定シミュレーション結果と実機による性能測定結果は、トポロジー、リンクバンド幅、ルーティングポリシー(アルゴリズム)によって異なる場合があった。そこで、NSIM-ACE に対して、実機に即したトポロジー、リンクバンド幅でもシミュレーションできるようにし、ルーティングについても実機と全く同じポリシーでパケットが転送されるように拡張した。これにより、基本的な FatTree のみならず、現場で実際に利用されているシステム構成を対象とした性能推定シミュレーションが可能となった。

本拡張は、実機システムの構成変更や、運用状況に応じたルーティングポリシー変更時の予備性能推定にも効果的である。したがって、シミュレーションによるアプリケーションの性能推定精度の向上のみならず、本課題終了後も研究機関や運用現場での有用性が期待できる。

3.4 アプリケーショングループ(九州大学)

(1)研究実施内容及び成果

アプリケーショングループは、本プロジェクトで研究開発する省メモリ低オーバーヘッド通信技術を活用した、スケーラブルなアプリケーション作成技術の開発を目的として、研究期間の前半の2年半程度を各アプリケーションの性能解析にあて、後半の3年程度で通信効率化と評価、および他グループへのフィードバックを行った。なお、前述の通り、アプリケーショングループは、5年度(平成27年)の8月以降、研究内容と構成員をインタフェースグループに統合している。

・分子軌道計算プログラム OpenFMO:

京コンピュータをはじめとしたクラスタ型超並列計算機において効率的に超並列 FMO の計算が可能となることを目指して、FMO の1つの実装である OpenFMO プログラムの高性能化に取り組んだ。方法としては、これまでのプログラムにおけるモノマーやダイマーの小規模電子状態計算部分を対象とし、主に MPI/OpenMP のハイブリッド並列化や新たに実装したグローバルカウンタを用いた動的負荷分散手法の適用を行った。その結果、フラグメントの電子状態計算において負荷の大きな分子積分計算部分の並列化効率が、256 並列時に 94%にまで向上し、プログラム全体でおおよそ 2 万並列までのスケーラビリティとすることに成功した。これにより、近い将来、京を超える大規模計算機において弱スケール並列性を利用した場合に、1000 フラグメント程度の分子に対する FMO 計算を手軽に行えるようになることが期待される。強スケール並列性については、細粒度並列処理により各モノマーやダイマーの計算遅延を小さくする必要があるが、これ以上の並列化については、我々の超並列化を実施するまであまり性能に影響を与えなかった対角化等の逐次処理の遅延が顕在化したため、今後は抜本的なアルゴリズム変更が必要となる。ただ、世界的にこの対角化の遅延の問題が顕在化するまでの並列化に成功した例はなく、画期的な結果が得られたと考えられる。

OpenFMO の通信パターンは、以下の4つから構成される。1. マスタプロセスから、複数のプロセスグループ(ワーカグループ)にジョブの分配や同期を行うマスタプロセス⇄ワーカグループ間通信(MW 通信)、2. 各ワーカグループ内にてグループのルートプロセスが他のプロセスに対し動的負荷分散によるタスク分配を行うためのグローバルカウンタ値の取得変更を行うための通信(GC 通信)、3. 各ワーカグループ内にて1つのモノマーやダイマーを計算する際に必要な Bcast や Reduce からなる集団通信(CO 通信)、4. ワーカグループ間のデータ共有に必要な通信(DS 通信)である。これらを我々が開発している ACP ライブラリを利用して実装した場合に、各プロセスが保持するグローバルアドレス(GA)や他のメモリ量についての見積もりを実施した。その結果、MW 通信では、最大でマスタプロセスにおいて 10MB 程度、GC 通信については、1つのグローバルカウンタ保持プロセスへの Atomic-fetch-and-add 型の通信(ACP では `acp_add 8` 操作)でよく、各プロセスに 10Byte 程度となった。また、各ワーカにおける CO 通信については、2分木アルゴリズムを利用する際には Bcast、Reduce 型をあわせて 40Byte 程度ですむ。最後の DS 通信については、

各ワーカグループのルートプロセスが 11 種類のデータにアクセスする必要があるため、グループルートプロセス毎に 110Byte 程度のメモリ量となる。その結果、GA 保持、つまり通信のための管理データのメモリ領域に関しては無視できるメモリ消費量となることが分かった。また、必要となる通信バッファについては、MW 通信と GC 通信、DS 通信についてはゼロコピー型の片側通信で全て実施することが可能である。CO 通信については、集団通信の途中のバッファが必要になるが最小限の量でよく、最大で 100MB 程度で十分である。その結果、ACP を OpenFMO に適用した場合には、ACP 部分が利用するメモリ量は 100MB 以下となり、十分に省メモリとなると見積もることが出来た。

・電磁流体(MHD)コード

電磁流体(MHD)コードは電磁場を考慮した流体計算で有り、その数値計算はステンシル計算である。このステンシル計算は一般的に格子を分割する領域分割法によって並列化され、その分割によって生じる袖(Halo)通信が並列計算時の主な通信となる。この Halo 通信を最適化し、エクサスケールにおいても高効率に計算できるようにすることが目標である。

まず初めに、MHD コードの特徴を把握するために、ベクトル機(SX シリーズ)、スカラ機(X86 系、SPARC 系、POWER 系)の計算機システムにおいて、電磁流体(MHD)コードの性能評価を行い、コードの実行性能、並列計算効率がどのようになっているか調査した。スカラ機においては、京コンピュータなど 5,000 プロセス並列以上のシステムを利用することができたため、超並列計算時に MHD コードの性能がどのように変化するか調べ、その結果、通信における同期が大きな計算負荷になっていると分かった。

次に、様々な計算機システムでの MHD コードの性能向上のため、コードにチューニングを実施した。計算機アーキテクチャ、通信トポロジーなどハードウェアの違いから、どのような領域分割や配列モデルが最適かということが分かり、計算機システムの違いによる最適手法を明らかにした。現在エクサフロップス級の計算機システムはないため、それらの情報を利用し、隣接ノードとの袖領域通信時間と計算時間の実測値から全体の性能予測を可能にするモデルを構築した。このモデルに関して、現在実測が可能なノード数の範囲内で検証した結果、かなり信頼性が高い性能予測が可能であることが分かっている。この性能予測の結果、同期による通信負荷からエクサフロップス級の計算では並列化効率が 20%以上劣化することが示されている。

この並列化効率の劣化問題を解決するために、通信と通信に関連する計算を行う Halo スレッド、通信に関連しない他の部分を計算する計算スレッドとスレッドの役割を分けることで、スレッド間で同期が必要の無く、プロセス間の同期も隠蔽できる計算モデルを開発した。この Halo スレッドを実際に MHD コードに導入し、性能評価したところ、計算スレッドの計算量が特別に多い場合を除き、weak スケーリングにおいて、導入しない場合より高い並列化効率を実現でき、strong スケーリングにおいては、導入しない場合には並列化効率が 50%を切るような場合でも、高い並列化効率(90%)を達せることができた。

更に、Halo スレッド内の通信と計算をオーバーラップさせるために Halo 関数を開発した。Halo 通信は通信量や通信相手が固定の場合が主で有り、この情報を利用し、Halo 関数では、データのパック・アンパック、通信自体の最適化を行っており、通信と計算をオーバーラップさせやすい実装となっている。この Halo 関数を導入するだけで、Xeon 搭載の PC クラスタで最大 2 倍近くの Halo 通信時間(パック・アンパック含む)の短縮が確認できた。Halo 関数を利用し、通信と計算をオーバーラップさせた場合には、シミュレーションコードの実装次第で性能が変わる場合があり、良い時では 1 割程度の性能向上が見られるが、最悪値では 2 割程度性能劣化することも分かった。この問題には事前のコードの性能調査か動的最適化が必要と考えられる。

最後に Halo スレッドと Halo 関数を他のステンシル計算から容易に利用できるように、ステンシル計算に対する Halo 通信インタフェースを作成し、公開をする予定である。

このうち Halo 関数は、アプリケーションから通信ライブラリを隠蔽することとなり、MPI 以外の通信でも実装可能である。そこで、現在、この関数の ACP ライブラリによる実装に取り組

んでいる。現在の Halo 関数の実装は MPI_Isend/MPI_Irecv を用いており、前述の通り、想定していたよりもオーバーラップの効果が得られにくいことが分かっている。一方 ACP ライブラリでは、非ブロッキングの片側通信を活用することにより、プロセス間の同期を最小限に出来る上、ゼロコピー通信によってプロセス内のメモリコピーを削減できるため、通信以外のオーバーヘッドを低減でき、より高いオーバーラップ効果が期待できる。そこで、予備実験として、2次元のステンシル計算において、Halo 通信を ACP で実装し、MPI_Isend/MPI_Irecv による実装、および MPI_Put による実装と比較して、ACP による実装が最も通信時間の隠蔽効果が得られることを示した。現在、インタフェースグループで、この知見に基づいてマルチチャンネルインタフェースを実装中であり、その後、そのインタフェースを用いた Halo 関数の実装を予定している。

・重力 N 体シミュレーション

重力 N 体シミュレーションや分子動力学などの粒子系シミュレーションでよく用いられる並列化法は領域分割である。領域分割法ではシミュレーション領域をプロセス数と同じ数の部分領域に分割し、各部分領域に存在する粒子の計算とデータを1つのプロセスに割り当てる。各時間ステップ後に隣接する部分領域に移動した粒子のデータはその担当プロセスへ移動させる。このような粒子データの通信では元の担当プロセスは新しい粒子位置を保持しているために新しい担当プロセスを決定できるが、新しい担当プロセスはまだ新しい粒子位置を保持していないため、どのプロセスから粒子データが送信されてくるか予測できない。従って、各プロセスはいくつのプロセスから粒子データが送信されてくるか予測できない。プロセス並列化で通常用いられる MPI の 1 対 1 通信では受信プロセスは送信してくるプロセスの数と同じ回数、受信関数を呼ぶ必要があるため、このような通信パターンを直接記述することができない。シミュレーション途中で領域分割をやり直して、部分領域を移動させる場合は、領域分割後に類似した粒子データの通信が現れる。

このような通信パターンは他のアプリケーション分野にも現れる。このような通信のためのアルゴリズムの性能評価はこれまでも行われているが、これらの性能評価は実際のアプリケーションのデータではなく、通信パターンを簡略化したベンチマークと性能モデルを用いているものか、専用の MPI ライブラリを用いているものである。我々は通信性能を測定し、一般的に利用されている MPI ライブラリまたは ACP 通信ライブラリを用いて粒子データ通信を実装し、実際のアプリケーションにおける粒子データを用いて性能評価を行った。

粒子データ通信を記述する方法の 1 つはまず各プロセスに送信してくるプロセス数を別の通信で集計し、その後その数と同じ回数、受信関数を呼ぶことである。集計は例えば MPI_Reduce_scatter 通信関数を用いて実行できる。他の方法の 1 つは片側通信を用いることである。片側通信では送信プロセスのみデータを送信する put 関数を呼び、受信プロセスで対応する関数を呼ぶ必要がない。従って、元の担当プロセスが put 関数を呼ぶことで粒子データ通信を直接記述できる。我々は MPI_Reduce_scatter 関数を用いた実装と片側通信による実装を含めて MPI による実装を 5 種類、ACP の片側通信による実装、計 6 種類の実装を作成した。これらの実装を用いて性能評価用のコードを作成した。重力 N 体シミュレーションコード GADGET-2 でスナップショットを生成し、そのスナップショットを性能評価用コードに読み込んで領域分割を行い、時間ステップの前後で担当プロセスが変わった粒子のデータの通信時間を測定した。また領域分割後の粒子データ通信に要した時間も測定した。

評価結果の一部を下図に示す。ACP による実装または MPI_Reduce_scatter 関数を用いた実装(reduce-scatter + send-recv)が最も高いまたはその次に高い性能を示している。また ACP による実装は MPI の片側通信(one-sided)に比べて 2 倍程度高速である。ACP の方が高速である理由の 1 つとして、ACP では通信関数を呼び出した直後に通信が開始されるのに対し、MPI の片側通信では実装上の都合により通信開始が遅れる場合があるためと考えられる。

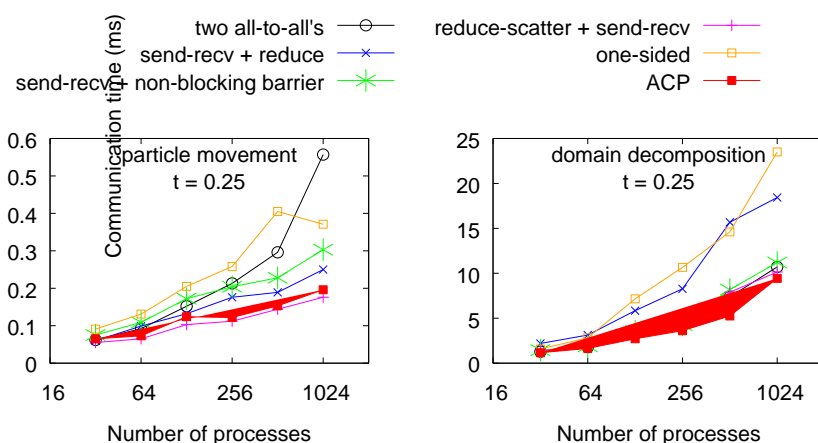


図: ACPによる粒子データ通信とMPIとの性能比較。左は粒子移動後、右は領域分割後の通信。富士通PRIMERGERY CX400、Intel MPI 5.1.3を使用。

・時間並列プログラム

通常的空間領域での並列化によるスピードアップが飽和した状況では、チューニング等による通信レイテンシなどの短縮にも限界があるため、超並列計算機で大幅な高速化を期待することはできない。このような場合の方法として、時間方向に領域分割を実施するアルゴリズムを導入することにより、さらにスピードアップが可能であることが知られている。この場合のプログラム構造は、時間方向については一次元の隣接通信によりパイプライン型の並列実行により高速化が可能であることが知られている。本研究では、この隣接通信をバケツリレー通信関数として実装し、非同期通信を利用してセグメント化を行うことで、通信レイテンシを大幅に短縮することに成功した。この方式により、分子動力学計算、量子動力学、線形計算などの幾つかのプログラムで高速化の効果を確かめた。これらの応用計算の多くでは、複数回実行されるバケツリレー通信のデータサイズ、および、通信相手は固定であることが多いため、固定バッファに対して片側通信による read/write を実施する形式で実装することで、通信関数中での不要なコピーをなくし、さらにレイテンシの短縮を図ることが可能である。

§4 成果発表等

(1)原著論文発表 (国内(和文)誌 11件、国際(欧文)誌 19件)

<国内>

1. 深沢圭一郎, 梅田隆行, 南里 豪志, “超並列惑星磁気圏電磁流体シミュレーションに向けた隣接通信の効率化”, 2012 ハイパフォーマンスコンピューティングと計算科学シンポジウム論文集, 101-106, 2012.
2. 松本幸, 安達知也, 住元真司, 曾我武史, 南里豪志, 宇野篤也, 黒川原佳, 庄司文由, 横川三津夫, “MPI Allreduce の「京」上での実装と評価”, 先進的計算基盤システムシンポジウム (SACIS2012), May. 2012
3. 松本幸, 安達知也, 住元真司, 南里豪志, 曾我武史, 宇野篤也, 黒川原佳, 庄司文由, 横川三津夫, “MPI Allreduce の「京」上での実装と評価”, 情報処理学会論文誌コンピューティングシステム(ACS), Vol. 5, No. 5, pp.152-162, 2012.
4. 稲富雄一, 眞木淳, 本田宏明, 高見利也, 小林泰三, 青柳睦, 南一生, “京コンピュータでの効率的な動作を目指した並列 FMO プログラム OpenFMO の高性能化”, Journal of Computer Chemistry, Japan, 2012.

5. 森江善之, 南里豪志, “多次元メッシュ/トーラスにおける通信衝突を考慮したタスク配置最適化技術”, 2013 ハイパフォーマンスコンピューティングと計算科学シンポジウム論文集, No. 2013, pp. 95-103, Jan. 2013.
6. 森江 善之, 南里 豪志, “多次元メッシュ/トーラスにおける通信衝突を考慮したタスク配置最適化技術”, 情報処理学会論文誌コンピューティングシステム, Vol. 6, No. 3, pp. 12-21, Sep. 2013.
7. 本田宏明, 稲富雄一, 眞木淳, “SIMD 演算の効率的利用に向けたアレイジョブ型 Hartree-Fock 法計算”, 情報処理学会論文誌 コンピューティングシステム Vol. 7, No. 4, pp. 15-24, 2014.
8. 本田 宏明, 稲富 雄一, 森江 善之, 南里 豪志, 高見 利也, “分子軌道法に向けた RDMA に基づく通信ミドルウェアの開発”, Journal of Computer Chemistry, Japan, Vol. 13, No. 6, pp. 335-336, 2014, (DOI: 10.2477/jccj.2014-0064)
9. 森江 善之, 南里 豪志, “直接網において複数の通信デバイスを有効に使用する隣接通信アルゴリズムの提案”, 情報処理学会論文誌コンピューティングシステム, Vol.8, No.4, pp.26-35, 2015.
10. 森江 善之, 南里 豪志, “直接網において複数の通信デバイスを有効に使用する隣接通信アルゴリズムの提案”, 2015 ハイパフォーマンスコンピューティングと計算科学シンポジウム論文集, 2015.
11. 深沢 圭一郎, 森江 善之, 曾我 武史, 高見 利也, 南里 豪志, “ Halo スレッドと Halo 関数を用いた MHD シミュレーションの高効率並列化”, 2017 ハイパフォーマンスコンピューティングと計算科学シンポジウム論文集, 2017(採択決定)。

<国際>

1. Fukazawa, K., T. Ogino, and R. J. Walker, “A magnetohydrodynamic simulation study of Kronian field-aligned currents and auroras”, J. Geophys. Res., 117, A02214, 2012 (doi: 10.1029/2011JA016945).
2. Fukazawa, K., T. Umeda, “Performance measurement of magnetohydrodynamic code for space plasma on the typical scalar type supercomputer systems with the large number of cores”, International Journal of High Performance Computing, 2012 (doi: 10.1177/1094342011434813).
3. T. Umeda and K. Fukazawa, “Performance measurement of parallel Vlasov code for space plasma on various scalar-type supercomputer systems”, In: Algorithms and Architectures for Parallel Processing, Lecture Notes in Computer Science, Vol.7439, edited by Y. Xiang, I. Stojmenovic, B.O. Apduhan, G. Wang, K. Nakano, and A. Zomaya, pp.233-240, 2012. (ISBN: 978-3-642-33077-3) (DOI:10.1007/978-3-642-33078-0_17)
4. T. Umeda, K. Fukazawa, Y. Nariyuki, and T. Ogino, “A Scalable Full Electro-Magnetic Vlasov Solver for Cross-scale Coupling in Space Plasma”, IEEE Transactions on Plasma Science, Vol. 40, No. 5, pp.1421-1429, 2012. (DOI:10.1109/TPS.2012.2188141)
5. T. Takami and A. Nishida, “Parareal Acceleration of Matrix Multiplication”, Advances in Parallel Computing, vol. 22, pp.437-444, 2012. (DOI:10.3233/978-1-61499-041-3-437)
6. Yoshiyuki Morie, and Nanri Takeshi, “Task Allocation Optimization for Neighboring Communication on Fat Tree”, in Proceedings of 14th IEEE International Conference on High Performance Computing and Communication , pp.1219-1225, Liverpool, United Kingdom, Jun. 2012. (DOI:10.1109/HPCC.2012.179)
7. K. Fukazawa, T. Nanri, “Performance of Large Scale MHD Simulation of Global Planetary Magnetosphere with Massively Parallel Scalar Type Supercomputer

- Including Post Processing”, in Proceedings of 14th IEEE International Conference on High Performance Computing and Communication, pp. 976–982, Liverpool, United Kingdom, Jun. 2012. (DOI:10.1109/HPCC.2012.142)
8. S. Fujino, T. Nanri, K. Kusaba, “Balancing Communication and Execution Technique for Parallelized Sparse Matrix–Vector Multiplication”, 4th International Conference on Future Computational Technologies and Applications, Jul. 2012.
 9. K. Fukazawa and T. Nanri, “Effective Performance of Large–Scale MHD Simulation for Planetary Magnetosphere with Massively Parallel Computer”, Proc. JSST, pp.243–247, 2012.
 10. K. Fukazawa, T. Nanri, and T. Umeda, “Performance evaluation of magnetohydrodynamics simulation for magnetosphere on K computer”, In: AsiaSim 2013, Communications in Computer and Information Science, Vol.402, edited by G. Tan, G. K. Yeo, S. J. Turner, and Y. M. Teo, pp.570–576, Springer–Verlag Berlin Heidelberg, 2013. (ISBN: 978–3–642–45036–5) (DOI: 10.1007/978–3–642–45037–2_61)
 11. K. Fukazawa, T. Nanri and T. Umeda, “Performance Measurements of MHD Simulation for Planetary Magnetosphere on Peta–Scale Computer FX10”, Parallel Computing: Accelerating Computational Science and Engineering (CSE), Advances in Parallel Computing 25, pp.387–394, IOS Press, 2014. (DOI: 10.3233/978–1–61499–381–0–387)
 12. T. Takami and D. Fukudome, “An efficient pipelined implementation of space–time parallel applications”, Parallel Computing: Accelerating Computational Science and Engineering (CSE), Advances in Parallel Computing 25, pp.273–281, IOS Press, 2014. (DOI: 10.3233/978–1–61499–381–0–273)
 13. Yuichiro Ajima, Hideyuki Akimoto, Tomoya Adachi, Takayuki Okamoto, Kazushige Saga, Kenichi Miura, and Shinji Sumimoto, Asynchronous Global Heap: Stepping Stone to Global Memory Management, In Proceedings of the 7th International Conference on Partitioned Global Address Space Programming Model, pp. 212–216, 2013
 14. T. Takami and D. Fukudome, “An identity parareal method for temporal parallel computations”, Lecture Notes in Computer Science Vol. 8384, edited by R. Wyrzykowski, J. Dongarra, K. Karczewski, and J. Wasniewski, pp. 67–75 (2014), (DOI: 10.1007/978–3–642–55224–3_7)
 15. Yuichiro Ajima, Takafumi Nose, Kazushige Saga, Naoyuki Shida and Shinji Sumimoto, “ACPd: Data–Structure and Global Memory Allocator Library over a Thin PGAS–Layer”, Proceedings of the First International Workshop on Extreme Scale Programming Models and Middleware, pp. 11–18, 2015
 16. Kin’ya Takahashi, Sho Iwagami, Taizo Kobayashi, Toshiya Takami, “Theoretical Estimation of the Acoustic Energy Generation and Absorption Caused by Jet Oscillation”, J. Phys. Soc. Jpn., Vol.85, No.4, Article ID: 044402
 17. Shinji Sumimoto, Yuichiro Ajima, Kazushige Saga, Takafumi Nose, Naoyuki Shida and Takeshi Nanri, The Design of Advanced Communication to Reduce Memory Usage for Exa–scale Systems, in Proceedings of 12th International Meeting on High Performance Computing for Computational Science, Porto, Portugal, June 28th to 30th, 2016, to be published as Springer’s Lecture Notes in Computer Science (LNCS).
 18. R. Susukita, Y. Morie, T. Nanri, H. Shibamura, NSIM–ACE: An Interconnection Network Simulator for Evaluating Remote Direct Memory Access, Proceedings of 6th International Conference on Simulation and Modeling Methodologies, Technologies

and Applications (SIMULTECH 2016), 2016

19. R. Susukita, Y. Morie, T. Nanri, Efficient communications of particle data in particle-based simulations, Proceedings of 35th JSST Annual Conference International Conference on Simulation Technology (JSST 2016), 2016

(2)その他の著作物(総説,書籍など)

1. 世木裕人,高見利也「リアルタイムシミュレーションのためのエッジトーンの一次元モデル化」九州大学大学院 システム情報科学紀要 第20巻 第2号 pp. 45-51 (2015)

(3)国際学会発表及び主要な国内学会発表

① 招待講演 (国内会議5件,国際会議10件)

<国内>

1. 深沢圭一郎(九州大学),“スーパーコンピュータを利用した大規模惑星磁気圏シミュレーションと可視化”,第17回ビジュアライゼーションカンファレンス,東京,2011年11月4日
2. 南里豪志(九州大学),“通信ライブラリにおける実行時自動チューニング技術”,第3回自動チューニング技術の現状と応用に関するシンポジウム,東京,2011年12月5日
3. 深沢圭一郎(九州大学),“磁気圏グローバルシミュレーションの動向とこれからの課題”,STEシミュレーション研究会,広島,2012年3月1-3日
4. 小林泰三(九州大学),複雑システムにおける創造的破壊現象の原理に迫る,京都大学基礎物理学研究所研究会,2015
5. 南里豪志(九州大学),エクサスケールに向けた省メモリ通信ライブラリ実装技術,STEシミュレーション研究会 - エクサスケールに向けて -, 2015

<国際>

1. K. Fukazawa(九州大学),“Trends and Future of Magnetospheric Global Simulations”, AOGS2012, Singapore, 13-17 Aug. 2012
2. K. Fukazawa(九州大学), T. Ogino, R. J. Walker, “Comparative Global Dynamics of Planetary Magnetospheres”, AOGS2012, Singapore, 13-17 Aug. 2012
3. H. Honda(九州大学), F. Mehdipour, H. Kataoka, K. Inoue, N. Yoshikawa, A. Fujimaki, H. Akaike, N. Takagi and K. J. Murakami, “Enhancing Performance and Power Efficiencies of Scientific Computing through an SFQ Reconfigurable Data-Path Processor”, Superconducting SFQ VLSI Workshop 2012 (SSV2012), Nagoya, 6-7 Dec. 2012.
4. Y. Inadomi(九州大学), “Performance improvement of FMO program for effective massively parallel execution on K-computer”, International Workshop on Massively Parallel Programming Now in Molecular Science, Tokyo, Japan, 28, Jan. 2013.
5. T. Nanri(九州大学), H. Sugiyama, K. Fukazawa, “A Cost-Efficient Approach for Automatic Algorithm Selection of Collective Communications”, SIAM Conference on Computational Science and Engineering, 25 Feb. - 1 Mar. 2013.
6. H. Honda(九州大学), Y. Inadomi, J. Maki, “Multiple-input parallel RHF calculation for improving SIMD operation efficiency”, 5th JCS International Symposium on Theoretical Chemistry, Nara, 2-6 Dec. 2013.
7. K. Fukazawa(京都大学), and R. J. Walker, An MHD Simulation of the Dynamics of the Kronian Magnetosphere Driven by Solar Wind Observations, AOGS 12th Annual Meeting,
8. Taizo Kobayashi(九州大学), Uncertainty and Dynamical Process on Computation,

International Workshop on Advanced Future Studies, 2015.

9. Yuichiro Ajima(富士通株式会社), Topology-Awareness in the Tofu Interconnect Series Slides, ExaComm 2016, Frankfurt, Germany, June 23rd, 2016
10. R. Susukita(九州大学), “Unpredictable Communication Patterns and Performance Analysis in Parallel Processing of Simulations”, 2017 International Conference for Leading and Young Computer Scientists (IC-LYCS 2017), Okinawa, 25-28 Feb. 2017

② 口頭発表 (国内会議 80 件, 国際会議 41 件)

<国内>

1. 高見利也(九州大学), 西田晃, “時間方向並列化の線形計算への適用可能性”, ハイパフォーマンスコンピューティング研究会, 京都, 2011 年 10 月 6 日
2. 深沢圭一郎(九州大学), 荻野瀧樹, “Study of Kronian Field-Aligned Currents and Aurora”, 地球電磁気・地球惑星圏学会 第130回総会及び講演会, 神戸, 2011 年 11 月 3-5 日
3. 高見利也(九州大学), “時間方向並列計算とマルチスケール動力学”, 研究集会「非線形現象の解析・モデル化・制御」, 鹿児島, 2012 年 3 月 4-6 日
4. 吉田匡兵(九州大学), 柴村英智, 井上弘士, 村上和彰, 全対全通信向けパケットペーシングにおける送信間隔の導出手法, 情報処理学会第 74 回全国大会, 1K-6, 2012 年 3 月 6-8 日.
5. 南里豪志(九州大学), 黒川原佳, “ランク配置に応じた集団通信アルゴリズム動的選択技術の提案”, 第 133 回ハイパフォーマンスコンピューティング研究会, 神戸, 2012 年 3 月 26-27 日.
6. 安島雄一郎(富士通株式会社), 秋元秀行, 岡本高幸, 三浦健一, 住元真司, “片側通信による, グローバルデータ構造の効率的な操作方法の検討”, 第 133 回ハイパフォーマンスコンピューティング研究会, 神戸, 2012 年 3 月 26-27 日.
7. 秋元秀行(富士通株式会社), 三浦健一, 岡本高幸, 安島雄一郎, 住元真司, “InfiniBand Atomic Operation の性能評価”, 第 133 回ハイパフォーマンスコンピューティング研究会, 神戸, 2012 年 3 月 26-27 日.
8. 三浦健一(富士通株式会社), 秋元秀行, 安島雄一郎, 岡本高幸, 住元真司. ” エクサスケールコンピューティングに向けた省メモリ通信ライブラリの検討”, 第 133 回ハイパフォーマンスコンピューティング研究会, 神戸, 2012 年 3 月 26-27 日.
9. 稲富雄一(九州大学), 眞木 淳, 高見利也, 本田宏明, 小林泰三, 南里豪志, 青柳睦, 南一生, “並列 FMO プログラム OpenFMO の性能最適化”, 第 133 回ハイパフォーマンスコンピューティング研究会, 神戸, 2012 年 3 月 26-27 日
10. K. Fukazawa(九州大学), R. J. Walker, and S. Eriksson, 「高時間分解能太陽風データを用いた土星磁気圏の MHD シミュレーション」, 日本地球惑星連合 2012 年大会, 千葉, 2012 年 5 月 20-25 日
11. 久保田康文, 村田健史, 山本和憲, 深沢圭一郎, 坪内健, 「NICT サイエンスクラウドによる, 地球磁気圏グローバル MHD シミュレーション大規模データの可視化技術」, 日本地球惑星連合 2012 年大会, 千葉, 2012 年 5 月 20-25 日
12. 村田健史, 久保田康文, 深沢圭一郎, 亘慎一, 渡邊英伸, 山本和憲, 田中高史, 「Global MHD シミュレーションの拡散領域の評価」, 日本地球惑星連合 2012 年大会, 千葉, 2012 年 5 月 20-25 日
13. 久保田康文, 村田健史, 山本和憲, 深沢圭一郎, 坪内健, 「グローバル MHD シミュレーション磁力線追尾システムによる磁束拡散領域の評価」, 日本地球惑星連合 2012 年大会, 千葉, 2012 年 5 月 20-25 日
14. 柴村英智(九州先端科学技術研究所), 三輪英樹, 三吉郁夫, 井上弘士, “パケットペーシングを用いた集団通信にするロード/ネットワークインバランスの影響”,

- 第 136 回ハイパフォーマンスコンピューティング研究発表会, 鳥取市, 2012 年 8 月 1 日
15. 南里 豪志(九州大学), Tofu ネットワークにおけるプロセス配置形状による集団通信アルゴリズムの性能解析, 第 136 回ハイパフォーマンスコンピューティング研究発表会, 那覇市, 2012 年 10 月 3-4 日.
 16. 深沢圭一郎(九州大学), 南里豪志, 高見利也, 「異なるスカラアーキテクチャ(x86, SPARC64)の電磁流体コードによる性能評価」, 第 136 回 HPC 研究会, 沖縄, 2012 年 10 月 3-4 日
 17. 梅田隆行, 伊藤陽介, 深沢圭一郎, 「超並列ブラソフコードによる宇宙プラズマの第一原理シミュレーション」, 第 136 回 HPC 研究会, 沖縄, 2012 年 10 月 3-4 日
 18. 深沢圭一郎(九州大学), 梅田隆行, 「Performance Measurement of Magnetohydrodynamic Simulation Code for Planetary Magnetosphere on FX10」, 地球電磁気・地球惑星圏学会 第 132 回総会及び講演会, 札幌, 2012 年 10 月 20-23 日
 19. 梅田隆行, 深沢圭一郎, 「超並列ブラソフシミュレーション」, 地球電磁気・地球惑星圏学会 第 132 回総会及び講演会, 札幌, 2012 年 10 月 20-23 日
 20. 村田健史, 久保田康文, 山本和憲, 深沢圭一郎, 坪内健, 「流体要素追跡による, 太陽風プラズマの磁気圏への侵入過程の研究」, 地球電磁気・地球惑星圏学会 第 132 回総会及び講演会, 札幌, 2012 年 10 月 20-23 日
 21. 森江 善之(九州大学), 南里 豪志, 「通信衝突を削減するタスク配置最適化における通信タイミングの予測方式の影響」, 第 194 回計算機アーキテクチャ・第 137 回ハイパフォーマンスコンピューティング合同研究発表会 (HOKKE-20), 北海道, 2012 年 12 月 13-14 日
 22. 南里 豪志(九州大学), 「スケーラブルな通信ライブラリ実装技術」, 第 8 回戦略的高性能計算システム開発に関するワークショップ, 2013 年 2 月 2 日.
 23. 南里 豪志(九州大学), 杉山 裕宣, 森江 善之, 「多次元メッシュ/トーラスにおけるプロセス配置に応じた集団通信アルゴリズム選択技術の提案」, 第 138 回ハイパフォーマンスコンピューティング研究会, あわら市, 2013 年 2 月 21-22 日.
 24. 安島雄一郎(富士通株式会社), 秋元秀行, 岡本高幸, 三浦健一, 住元真司, 「非同期グローバルヒープの提案と初期検討」, 第 138 回ハイパフォーマンスコンピューティング研究会, あわら市, 2013 年 2 月 21-22 日.
 25. 住元真司(富士通株式会社), 安島雄一郎, 安達知也, 岡本高幸, 秋元秀行, 三浦健一, 「遠隔 Atomic 通信を用いた省メモリ性実現のための方式検討」, 第 138 回ハイパフォーマンスコンピューティング研究会, あわら市, 2013 年 2 月 21-22 日.
 26. 秋元秀行(富士通株式会社), 安島雄一郎, 安達知也, 岡本高幸, 三浦健一, 住元真司, 「DMATP-MPI: MPI 向け動的メモリ割当分析ツール」, 第 138 回ハイパフォーマンスコンピューティング研究会, あわら市, 2013 年 2 月 21-22 日.
 27. 高見利也(九州大学), 福留大貴, 「時間領域並列化法の通信パターンと効率実装」, 第 138 回 HPC 研究会, 福井, 2013 年 2 月 21 日-22 日
 28. 村上和彰(九州大学), 本田宏明, 「ポストエクサスケールスーパーコンピュータの実現の観点から見た超伝導デジタル技術への期待」, 2013 年電子情報通信学会総合大会 エレクトロニックソサイアティ CT-1-6, 岐阜, 2013 年 3 月 19 日-22 日
 29. 安島 雄一郎(富士通株式会社), 秋元 秀行, 安達 知也, 岡本 高幸, 佐賀 一繁, 住元 真司, 三浦 健一, 「グローバルデータ構造のためのメモリ管理モデルの検討」, SWoPP 2013, 北九州市, 8 月 2 日
 30. 南里豪志(九州大学), 「通信ライブラリの自動チューニングを支援する Hint API の提案」, 第 141 回ハイパフォーマンスコンピューティング研究発表会, 那覇市, 2013 年 10 月 1 日

31. 柴村英智(九州先端科学技術研究所,FX10におけるパケットペーシングを用いたアプリケーションの通信性能評価,情報処理学会研究報告,Vol.2013-HPC-141,No.14,pp.1-7,沖縄県那覇市,10月1日.
32. 本田 宏明(九州大学),稲富 雄一,眞木 淳,“SIMD 演算有効利用のための入力データ並列 2 電子フォック行列計算”,第 141 回 HPC 研究会,沖縄,2013 年 9 月 30 日-10 月 1 日
33. 深沢圭一郎(九州大学),“Performance Measurement of MHD Simulation Code for Planetary Magnetosphere on Xeon Phi”,地球電磁気・地球惑星圏学会 第 134 回総会及び講演会,高知,2013 年 11 月 2-5 日
34. 児玉大器(九州大学),南里豪志,“性能予測と実測を併用した集団通信アルゴリズム選択”,今後の HPC(基盤技術と応用)に関するワークショップ,長崎市,2013 年 12 月 8 日
35. 南里豪志(九州大学),“MPI における最適化情報提供のためのインタフェースに関する評価”,今後の HPC(基盤技術と応用)に関するワークショップ,長崎市,2013 年 12 月 9 日
36. 杉山裕宣(九州大学),南里豪志,“プログラムのヒント情報を用いた通信ライブラリ動的最適化技術について”,今後の HPC(基盤技術と応用)に関するワークショップ,長崎市,2013 年 12 月 9 日
37. 本田宏明(九州大学),稲富雄一,眞木 淳,“SIMD 演算の活用に向けたアレイジョブ型の分子軌道法計算”,今後の HPC(基盤技術と応用)に関するワークショップ,長崎,2013 年 12 月 8-9 日
38. 薄田竜太郎(九州大学),“分布が一樣でない粒子シミュレーションにおける負荷分散”,今後の HPC(基盤技術と応用)に関するワークショップ,長崎,2013 年 12 月 8-9 日
39. 深沢 圭一郎(九州大学),片桐 孝洋,大宮 学,江川 隆輔,大島 聡史,青木 尊之,下川辺 隆史,荻野 正雄,岩下 武史,東田 学,“実アプリを用いた様々なアーキテクチャからなる計算機システムの性能評価”,第 199 回 ARC・第 142 回 HPC 合同研究発表会,札幌,2013 年 12 月 16-17 日
40. 梅田 隆行,深沢 圭一郎,“京,FX10 及び CX400 におけるブラソフコードの性能チューニング”,STE シミュレーション研究会&太陽地球惑星系科学(STP)シミュレーション・モデリング技法勉強会 合同研究集会 - 宇宙プラズマ・大気・天体 -,福岡,2013 年 12 月 24-27 日
41. 深沢 圭一郎(九州大学),梅田 隆行,“エクサ・ゼタスケールに向けた宇宙プラズマ分野における計算法の発展”,環瀬戸内応用数理研究部会第 17 回シンポジウム,松山,2014 年 1 月 11-12 日
42. 高見利也(九州大学),“Parareal 法の応用とマルチスケール解析”,環瀬戸内応用数理研究部会第 17 回シンポジウム,松山,2014 年 1 月 11-12 日
43. 住元 真司(富士通株式会社),安島 雄一郎,佐賀 一繁,野瀬 貴史,三浦 健一,南里 豪志,“エクサスケール通信向け ACP スタックの設計思想”,第 143 回 HPC 研究会,七尾市,3 月 3 日
44. 安島 雄一郎(富士通株式会社),佐賀 一繁,野瀬 貴史,三浦 健一,住元 真司,“ACP 基本層のインタフェース”,第 143 回 HPC 研究会,七尾市,3 月 3 日
45. 佐賀 一繁(富士通株式会社),安島 雄一郎,野瀬 貴史,三浦 健一,住元 真司,“ACP 基本層の実装と初期評価”,第 143 回 HPC 研究会,七尾市,3 月 3 日
46. 高見利也(九州大学),福留大貴,“恒等変換による時間並列化法 Identity Parareal の性能とバケツリレー通信”,第 143 回 HPC 研究会,石川,2014 年 3 月 3-4 日
47. 本田宏明(九州大学),“通信の省メモリ化に向けた Remote Memory Access に基づく分子軌道法プログラムの実装”,ワークショップ「工学・理学をベースにした線

- 形・非線形問題へのアプローチ」,石垣島,2014年5月10-11日.
48. 本田宏明(九州大学),”量子化学計算向け ACP 通信ライブラリの開発”,第2回 CUTE シンポジウム コンピュータ化学,三重,2014年11月1日
 49. 安島雄一郎(富士通株式会社),”ACP のグローバルデータ構造インタフェース”,第146回 HPC 研究会,沖縄,2014年10月2日~3日
 50. 野瀬貴史(富士通株式会社),”メタプログラミングに適した PGAS 通信クラスライブラリの検討”,第146回 HPC 研究会,沖縄,2014年10月2日~3日
 51. 高見利也(九州大学),”小規模疎行列演算の並列化手法と効率” 環瀬戸内応用数理研究部会第18回シンポジウム,倉敷,2014年12月5-6日
 52. 薄田竜太郎(九州大学),”RDMA 評価のための大規模インターコネクトシミュレータ「NSIM-ACE」”情報処理学会 第205回計算機アーキテクチャ・第147回ハイパフォーマンスコンピューティング合同研究発表会(HOKKE-22),小樽,2014年12月9-10日
 53. 南里豪志(九州大学),”チャンネル:省メモリ通信インタフェースの開発” 大学 ICT 推進協議会 2014 年度年次大会,仙台,2014年12月10-12日
 54. 安島雄一郎(富士通株式会社),”ACP の分散連想配列インタフェース”,第148回 HPC 研究会,別府,2015年3月2日~3日
 55. 野瀬貴史(富士通株式会社),”Tofu インターコネクト 2 上での ACP 基本層の実装と性能評価”,第148回 HPC 研究会,別府,2015年3月2日~3日
 56. 森江善之(九州大学),”InfiniBand による ACP 基本層の実装と評価”,第148回 HPC 研究会,別府,2015年3月2日~3日
 57. 本田 宏明(九州大学),”ACP ライブラリの集団通信インタフェース”,第148回 HPC 研究会,別府,2015年3月2日~3日
 58. 本田 宏明(九州大学),”ACP 通信ライブラリによる集団通信の実装と RHF 計算”,第12回稲盛フロンティア研究講演会,福岡,2015年1月
 59. 小林泰三(九州大学),”曲管を有する管楽器を対象とした大規模並列流体音シミュレーション”,JHPCN:学際大規模情報基盤共同利用・共同研究拠点 第7回シンポジウム,14-NA17,2015.7.
 60. 薄田 竜太郎(九州大学),森江善之,南里 豪志,柴村 英智,RDMA における同期通信のインターコネクトシミュレーション,2015年並列/分散/協調処理に関する『別府』サマー・ワークショップ(SWoPP2015),2015.8.
 61. 野瀬 貴史(富士通株式会社),ACPライブラリの性能最適化に関する検討,第150回ハイパフォーマンスコンピューティング研究発表会,2015.8.
 62. 杉山裕宣(九州大学),南里 豪志,RDMAモデル向け集団通信インタフェースの実装技術に関する研究開発,第150回ハイパフォーマンスコンピューティング研究発表会,2015.8.
 63. 深沢圭一郎(京都大学),森江善之,曾我武史,高見利也,南里豪志,Halo スレッドの MHD シミュレーションへの導入と性能評価,STE シミュレーション研究会 - エクサスケールに向けて -,2015.9.
 64. 深沢圭一郎(京都大学),森江善之,曾我武史,高見利也,南里豪志,エクサスケールコンピューティングに向けた Halo スレッドの電磁流体シミュレーションに対する効果,第151回ハイパフォーマンスコンピューティング研究発表会,2015.9
 65. 安島雄一郎(富士通株式会社),ACP グローバルメモリ管理アルゴリズムの改善,第151回ハイパフォーマンスコンピューティング研究会,那覇市,2015.9
 66. 薄田 竜太郎(九州大学),南里 豪志,ACP 通信ライブラリを用いた粒子系シミュレーションにおける粒子データ通信の実装,環瀬戸内応用数理研究部会第19回シンポジウム,2016.1.
 67. 安島雄一郎(富士通株式会社),ACPdl: Data-Structure and Global Memory Allocator Library over a Thin PGAS-Layer,ACSI2016,2016.1.

68. Takeshi Nanri, Evaluation of On-Demand Message-Passing Module over RDMA Network, ACSI2016, 2016.1.
69. 深沢圭一郎 (京都大学), 森江善之, 曾我武史, 高見利也, 南里豪志, ステンシル計算における効率的な Halo 通信・計算モデルの開発, 第 153 回ハイパフォーマンスコンピューティング研究発表会, 2016.3.
70. 森江 善之 (九州大学), 南里 豪志, ACP ライブラリの通信性能およびメモリ使用量の評価, 第 153 回ハイパフォーマンス研究発表会, 2016.3.
71. 本田宏明 (九州大学), ACP ライブラリによる MPI_Comm_spawn の置き換えと OpenFMO への適用, 情報処理学会研究報告 2015-HPC-153, 2016.3.
72. 杉山裕宣 (九州大学), 南里 豪志, 非同期通信を使用した動的な集団通信アルゴリズム選択技術の実装と評価, 第 153 回ハイパフォーマンスコンピューティング研究発表会, 2016.3.
73. 今村 憲亮 (九州大学), 本田 宏明, 原始基底 RHF 分子軌道法計算プログラムの SIMD 最適化, 第 153 回ハイパフォーマンスコンピューティング研究発表会, 2016.3.
74. 安島雄一郎 (富士通株式会社), 基本層データ転送方式の改良による ACP データ構造操作関数の高速化, 第 153 回ハイパフォーマンスコンピューティング研究会, 別府市, 2016 年 3 月 2 日
75. 本田宏明 (九州大学), 森江善之, 南里豪志, “ACP 通信ライブラリによる複数 MPI プログラムの接続の提案”, オーガナイズドセッション: PGAS プログラミング環境, HPCS2016, 仙台, 2016 年 6 月 6-7 日.
76. 深沢圭一郎 (京都大学), 森江善之, 曾我武史, 高見利也, 南里豪志, 並列ステンシル計算における Halo 領域通信の高効率実装, H28 年度 STE シミュレーション研究会, 仙台, 2016 年 8 月 31 日-9 月 2 日
77. 安島雄一郎 (富士通株式会社), ACP 基本層 UDP 版におけるノード内複数プロセス時のノード間通信性能の改善, 第 156 回ハイパフォーマンスコンピューティング研究会, 小樽市, 2016 年 9 月 15 日
78. 森江善之 (九州大学), 本田宏明, 南里 豪志, “ACP ライブラリの通信性能およびメモリ使用量の評価”, 情報処理学会研究会報告. HPC, ハイパフォーマンスコンピューティング, Vol. 2016-HPC-157(24), pp. 1-6, (2016).
79. 薄田竜太郎 (九州大学), “NSIM-ACE による RDMA 通信のシミュレーション”, 「ポストペタスケール高性能計算に資するシステムソフトウェア技術の創出」研究領域 平成28年度公開ワークショップ, 東京, 2016 年 12 月 14-15 日
80. 安島 雄一郎 (富士通株式会社), ACP 基本層 UDP 版におけるノード内通信性能の改善, 第 157 回ハイパフォーマンスコンピューティング研究会, 那覇市, 2016 年 12 月 22 日

<国際>

1. Y. Morie (九州大学), T. Nanri, “Task Allocation Optimization for Neighboring Communication on Fat Tree”, 14th IEEE International Conference on High Performance Computing and Communication 9th IEEE International Conference on Embedded Software and Systems, HPCC-ICCESS 2012, pp.1219-1225, Liverpool, UK, 25-27 Jun. 2012.
2. K. Fukazawa (九州大学) and T. Nanri, “Performance of large scale MHD simulation of global planetary magnetosphere with massively parallel scalar type supercomputer including post processing”, The 14th IEEE International Conference on High Performance Computing and Communications, Liverpool, UK, 25-27 Jun, 2012.
3. T. Nanri (九州大学), M. Kurokawa, “Efficient Runtime Algorithm Selection of

- Collective Communication with Topology-Based Performance Models”, International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)’12, Las Vegas, USA, 16–19 Jul. 2012.
4. T. Takami (九州大学), “Dependent Iterations in Scientific Calculations Parallelized by the Parareal-in-Time Algorithm”, CSC’12, Las Vegas, USA, 16–19 Jul. 2012
 5. Y. Kubota, K. T. Murata, K. Yamamoto, and K. Fukazawa, “Estimation of Magnetic Flux Diffusion Region Using a System of Magnetic Field Tracing in Global MHD Simulations”, AOGS2012, Singapore, 13–17 Aug. 2012
 6. T. Umeda and K. Fukazawa, “Performance Measurement of Parallel Vlasov Code for Space Plasma on Various Scalar-Type Supercomputer Systems”, The 12th International Conference on Algorithms and Architectures for Parallel Processing, Fukuoka, Japan, 4–7 Sept., 2012
 7. K. Fukazawa (九州大学) and T. Nanri, “Effective Performance of Large-Scale MHD simulation for Planetary Magnetosphere with Massively Parallel Computer”, JSST2012 International Conference on Simulation Technology, Kobe, Japan, 27–28 Sept., 2012
 8. K. Fukazawa (九州大学), R. J. Walker, S. Eriksson, and R. Wilson, “Global MHD simulation of Kronian magnetosphere with solar wind data from Cassini”, AGU Fall Meeting 2012, San Francisco, 3–7 Dec. 2012
 9. T. Ogino, R. J. Walker, and K. Fukazawa, “Boundary layer dynamics and energy transport in the earth’s magnetosphere”, AGU Fall Meeting 2012, San Francisco, 3–7 Dec. 2012
 10. T. Nanri (九州大学), “Introduction of ACE(Advanced Communication library for Exa) Project”, International workshop on HPC, Krylov Subspace method and its application, Beppu, Japan, 13–14, Jan. 2013.
 11. Y. Morie (九州大学), T. Nanri, “TASK ALLOCATION METHOD FOR AVOIDING CONTENTIONS BY THE INFORMATION OF CONCURRENT COMMUNICATION”, International workshop on HPC, Krylov Subspace method and its application, Beppu, Japan, 13–14, Jan. 2013.
 12. T. Okuma (九州大学), T. Nanri, “Evaluation of Implementation Methods for Non-Blocking Collective Communications in Overlapping Communication and Computation”, International workshop on HPC, Krylov Subspace method and its application, Beppu, Japan, 13–14, Jan. 2013.
 13. H. Sugiyama (九州大学), T. Nanri, “Performance Prediction Technology for Collective Communication Algorithm on Multi-Dimensional Mesh/Torus”, International workshop on HPC, Krylov Subspace method and its application, Beppu, Japan, 13–14, Jan. 2013.
 14. Takeshi Nanri (九州大学), “What Communication Library Can do with a Little Hint from Programmers?”, MVAICH User Group Meeting, Columbus, OH, Aug. 2013.
 15. Yoshiyuki Morie (九州大学), and Takeshi Nanri, “A neighbor communication algorithm with making an effective use of NICs on multidimensional-mesh/torus”, International Conference on Simulation Technology (JSST2013), in Tokyo, Sep. 2013.
 16. T. Takami (九州大学) and D. Fukudome, “A simple implementation of parareal-in-time on a parallel bucket-brigade interface”, 10th International Conference on Parallel Processing and Applied Mathematics, Warsaw, Poland, 8–11 Sep. 2013.
 17. Tsuyoshi Okuma (九州大学) and Takeshi Nanri, “Performance Study of

- Non-blocking Collective Communication Implementations Toward Adaptive Selection,” Bulletin of Networking, Computing, Systems and Software, Matsuyama, Japan, Dec. 2013.
18. Hironobu Sugiyama (九州大学) and Takeshi Nanri, “Topology Aware Performance Prediction of Collective Communication Algorithms on Multi-Dimensional Mesh/Torus,” Bulletin of Networking, Computing, Systems and Software, Matsuyama, Japan, Dec. 2013.
 19. Takeshi Nanri (九州大学), “Proposal of HINT Interface for Runtime Tuning of Communication Links,” 22nd Euromicro International Conference on Parallel, Distributed and network-based Processing, Turin, Italy, Feb. 2014.
 20. T. Takami (九州大学), “Identity Parareal Method and its Performance,” 3rd Workshop on Parallel-in-Time integration, Jülich, Germany, May 26-28, 2014
 21. T. Nanri (九州大学), “Design and Implementation of Channel Interface as a Memory Efficient Communication Model”, Annual Meeting on Advanced Computing System and Infrastructure (ACSI) 2015, Tsukuba, Japan, Jan 2015.
 22. T. Nanri (九州大学), “Channel Interface: a Primitive Model for Memory Efficient Communication”, 23rd Euromicro International Conference on Parallel, Distributed and network-based Processing, Turku, Finland, Feb. 2015.
 23. Y. Morie (九州大学), “Implement and Evaluation of ACP Basic Layer of InfiniBand”, International Workshop on Information Technology, Applied Mathematics and Science (IMS2015), Kyoto, Japan, Mar. 2015.
 24. Hiroaki Honda (九州大学), “Explicit Hamiltonian Matrix Expressions based on Vector-Recoupling Formula for Molecular Orbital Configuration Interaction Calculation”, International Workshop on Information Technology, Applied Mathematics and Science (IMS2015), pp.99-105, Kyoto, Japan, Mar.2015.
 25. Toshiya Takami (九州大学), “Stability and performance estimation of space-time multi-grid parallelization”, International Workshop on Information Technology, Applied Mathematics and Science (IMS2015), pp.71-74, Kyoto, Japan, Mar.2015.
 26. Keiichiro Fukazawa (京都大学), “Performance Evaluation of MHD Simulation Code with Many Core Systems”, International Workshop on Information Technology, Applied Mathematics and Science (IMS2015), pp.85-89, Kyoto, Japan, Mar.2015.
 27. Hiroaki Honda (九州大学), Performance Evaluation of Hartree-Fock Program developed by Ruby Scripting Language, 1st Pan-American Congress on Computational Mechanics (PANACM 2015), 2015.4.
 28. Takeshi Nanri (九州大学), Performance and Memory Usage Evaluations for Channel Interface of Advanced Communication Primitives Library, 1st Pan-American Congress on Computational Mechanics (PANACM 2015), 2015.4.
 29. Taizo Kobayashi (九州大学), A New Bottleneck in Large-Scale Numerical Simulations of Transient Phenomena, and Cooperation Between Simulations and the Post-Processes, PANACM 2015, 1st. Pan-American Congress on Computational Mechanics, 2015.4.
 30. S. Iwagami and Taizo Kobayashi, Numerical Analysis on the Lighthill Sound Sources of Oscillating Jet, PANACM 2015, 1st. Pan-American Congress on Computational Mechanics, 2015.4.
 31. Hiroaki Honda (九州大学), A Prototyping Environment for Electronic Structure Calculations by Scripting Language, Workshop on Information Technology, Applied Mathematics and Education (IME2015), 2015.8.
 32. Hidetomo Shibamura (九州先端科学技術研究所), Active Packet Pacing as a

- Congestion Avoidance Technique in Interconnection Network, International Conference on Parallel Computing 2015 (ParCo 2015), 2015.9.
33. Takafumi Nose (富士通株式会社), Basic Layer and Data Library of ACP (Advanced Communication Primitives) Library, LENS2015, 2015.10
 34. Takeshi Nanri (九州大学), Introduction of ACE (Advanced Communication library for Exa) Project, International Workshop on Language, Network and System Software 2015 (LENS2015), 2015.10.
 35. Hiroaki Honda (九州大学), Development of Applications on ACP Library, Language, Network and System Software 2015 (LENS2015), 2015.10.
 36. Hidetomo Shibamura (九州先端科学技術研究所), Simulation of RDMA Communication with NSIM-ACE, International Workshop on Language, Network and System Software 2015 (LENS2015), 2015.10.
 37. Ryutaro Susukita (九州大学), Yoshiyuki Morie, Takeshi Nanri, Hidetomo Shibamura, Performance Evaluation of RDMA Communication Patterns by Means of Simulations, 2015 Joint International Mechanical, Electronic and Information Technology Conference, 2015.12.
 38. Yuichiro Ajima (富士通株式会社), Reducing manipulation overhead of remote datastructure by controlling remote memory access order, ExaComm 2016 Workshop, Frankfurt, Germany, June 23rd, 2016 (DOI:10.1007/978-3-319-46079-6_7)
 39. H. Honda (九州大学), Y. Morie, T. Nanri, "Development of A Memory Efficient Communication Method for Connecting MPI Programs by using ACP Library", The 35th JSST Annual Conference International Conference on Simulation Technology (JSST 2016), Kyoto, Japan, Oct. 27-29, 2016.
 40. T. Nanri (九州大学), K. Fukazawa, "Effect of Overlapping Halo Exchange with One-Sided Communication," the 35th JSST Annual Conference International Conference on Simulation Technology, Oct. 2016.
 41. Shinji Sumimoto, Yuichiro Ajima, Takafumi Nose, Kazushige Saga, Naoyuki Shida, Morie Yoshiyuki, Takeshi Nanri, Parallel Application Experiences Using Advanced Communication Primitives, 25th Euromicro International Conference on Parallel, Distributed and network-based Processing (PDP 2017), March 6-8, 2017

③ ポスター発表 (国内会議 21 件, 国際会議 26 件)

<国内>

1. 森江 善之 (九州大学), 南里 豪志, "隣接通信に対する通信衝突を考慮した通信性能向上のためのタスク配置最適化の評価", 2012年ハイパフォーマンクスコンピューティングと計算科学シンポジウム, 愛知県名古屋市, 2012年1月24-26日
2. 稲富 雄一 (九州大学), 眞木 淳, 高見 利也, 小林 泰三, 青柳 睦, 南 一生, "並列 FMO プログラム OpenFMO の性能最適化", ハイパフォーマンクスコンピューティングと計算科学シンポジウム (HPCS2012), 名古屋, 2012年1月24-26日
3. 大隈 強志 (九州大学), 高見 利也, 松野 哲也, 青柳 睦, "分子動力学計算の時間方向並列化と通信パターン", ハイパフォーマンクスコンピューティングと計算科学シンポジウム (HPCS2012), 名古屋, 2012年1月24-26日
4. 稲富 雄一 (九州大学), 眞木 淳, 高見 利也, 小林 泰三, 青柳 睦, 南 一生, "超並列 FMO プログラム OpenFMO", 第2回次世代ナノ統合シミュレーションソフトウェア説明会, 東京, 2012年1月24-26日
5. 深沢 圭一郎 (九州大学), "超大規模計算時代における並列処理とポスト処理の問題点と解決手法", 宇宙科学情報解析シンポジウム, 神奈川, 2012年2月15日

6. 稲富雄一(九州大学), 眞木淳, 高見利也, 小林泰三, 青柳睦, “超並列 FMO プログラム OpenFMO の高性能化”, 次世代ナノ統合シミュレーションソフトウェアの研究開発 ナノ・ライフ公開シンポジウム, 神戸市, 2012 年 3 月 5-6 日
7. 本田 宏明(九州大学), 稲富 雄一, 眞木 淳, 「RT 並列法によるメニーコアプロセッサ向け分子軌道法計算」, 分子科学討論会 2012, 東京, 2012 年 9 月 18 日-21 日
8. 稲富雄一(九州大学), 眞木淳, 本田宏明, 高見利也, 小林泰三, 青柳睦, 南一生, 「並列 FMO プログラム OpenFMO の高性能化」, 分子科学討論会 2012, 東京大学, 2012. 9. 18-21
9. 本田 宏明(九州大学), 稲富 雄一, 眞木 淳, 「メニーコアプロセッサ SIMD 演算向けのデータ並列 2 電子フォック行列計算」, 日本コンピュータ化学会 2012 秋季年会, 山形, 2012 年 10 月 13 日-14 日
10. 本田 宏明(九州大学), 稲富 雄一, 眞木 淳, “メニーコアプロセッサ SIMD 演算向けのデータ並列 2 電子フォック行列計算 II”, 日本コンピュータ化学会 2013 春季年会, 東京, 2013 年 5 月 30-31 日
11. 秋元秀行(九州大学), MPI 向け動的メモリ割当分析ツール (DMATP-MPI), SACSIS 2013, 仙台市, 2013 年 5 月 23 日
12. 筒井 亮太, Liu Kai, 折本 裕一, 本田 宏明, 青木百合子, “ELG 法による含金属 BN/C ヘテロナノチューブの電子状態と機能解析”, 第 7 回分子科学討論会 2013, 京都, 2013 年 9 月 24-27 日
13. 本田 宏明(九州大学), 稲富 雄一, 眞木 淳, “スクリプト言語を用いた分子軌道法プログラムの開発”, 第 7 回分子科学討論会 2013, 京都, 2013 年 9 月 24-27 日
14. 本田 宏明(九州大学), 稲富 雄一, 眞木 淳, “メニーコアプロセッサ SIMD 演算向けのデータ並列 2 電子フォック行列計算 III”, 日本コンピュータ化学会 2013 秋季年会, 福岡, 2013 年 10 月 18-19 日
15. 高見 利也(九州大学), 福留 大貴, “新しい時間並列化法 iParareal とマルチスケール解析”, HPCS2014, 東京, 2014 年 1 月 7-8 日
16. 本田宏明(九州大学), “PGAS モデルに基づく通信ライブラリを用いた OpenFMO プログラムの実装”, 日本コンピュータ化学会 2014 春季年会, 東京, 2014. 5
17. 本田 宏明(九州大学), “スクリプト言語を利用したハミルトニアン行列利用環境の開発”, 第 8 回 分子科学討論会, 東広島, 2014 年 9 月
18. 本田 宏明(九州大学), “分子軌道法に向けた RDMA に基づく通信ミドルウェアの開発”, 日本コンピュータ化学会 2014 秋季年会, 郡山, 2014 年 10 月
19. 本田宏明(九州大学), 分子軌道法プログラムでの利用に向けた ACP 通信ライブラリによる集団通信の実装, 日本コンピュータ化学会 2015 春季年会, 2015.5
20. 本田宏明(九州大学), スクリプト言語を利用した電子構造計算プロトタイプ環境の開発, 第 9 回 分子科学討論会, 2015.9
21. 本田宏明(九州大学), ACP 通信ライブラリを用いた OpenFMO プログラムの実装 日本コンピュータ化学会 2015 秋季年会, 2015.10

<国際>

1. Umeda, T., Y. Ito, Y. Kiani, and K. Fukazawa, “Study of the Moon-Solar Wind Interaction with Parallel Kinetic Plasma Simulations”, SC11, Seattle, USA, 12-18 Nov. 2011
2. Fukazawa, K. (九州大学), T. Umeda, and T. Ogino, “Massively Parallel Computing of Magneto-Hydro-Dynamic Simulation Toward Peta and Exa Scale Computers”, SC11, Seattle, USA, 12-18 Nov. 2011
3. Fukazawa, K. (九州大学), T. Ogino, and R. J. Walker, “A High Resolution

- Magnetohydrodynamic Simulation Study of Kronian Field-Aligned Currents and Aurora”, AGU Fall Meeting 2011, San Francisco, 5-9 Dec. 2011
4. Y. Inadomi(九州大学), J. Maki, H. Honda, T. Takami, T. Nanri, M. Aoyagi, and K. Minami, “Performance Tuning of FMO Code (OpenFMO) for Effective Massively Parallel Execution”, Theory and Applications of Computational Chemistry (TACC-2012), Pavia, Italy, 2-7, Sep. 2012.
 5. K. Fukazawa(九州大学), T. Nanri, and T. Takami, “Massively Parallel Computing of Magneto- Hydro-Dynamic Simulation Toward Peta and Exa Scale Computers”, SC12, Salt Lake City, USA, 10-16 Nov. 2012
 6. Hidetomo Shibamura (九州先端科学技術研究所), Congestion Avoidance Technique Using Aggressive Packet Pacing toward Exascale Interconnect, International Supercomputing Conference 2013 (ISC’13), Poster at HPC in Asia Poster Session (17a), Leipzig, Germany, June 17 2013.
 7. Shinji Sumimoto(富士通株式会社), Hideyuki Akimoto, Yuichiro Ajima, Takayuki Okamoto, Tomoya Adachi, and Kenichi Miura, “Dynamic Memory Usage Analysis of MPI Libraries Using DMATP-MPI”, EuroMPI’13, Madrid, Spain, 9月16日
 8. D. Fukudome(九州大学) and T. Takami, “Parallel bucket-brigade communication interface for scientific applications”, EuroMPI’13, Madrid, 15-18 Sep. 2013.
 9. Yoshiyuki Morie (九州大学) and Takeshi Nanri, “Implementation of Neighbor Communication Algorithm Using Multi-NICs Effectively by Extended RDMA Interfaces”, SC13, in Denver, Nov. 2013.
 10. H. Honda(九州大学), Y. Inadomi, J. Maki, “Molecular orbital calculation for efficient SIMD operations by means of multiple geometry inputs”, The 4th AICS International Symposium, Kobe, 1-2 Dec. 2013.
 11. T. Takami(九州大学), “Identity Parareal for Time-evolution Problems”, The 4th AICS International Symposium, Kobe, 1-2 Dec. 2013.
 12. K. Fukazawa(九州大学), T. Ogino, and R. J. Walker, “Key parameter of planetary magnetospheric configuration and dynamics”, AGU Fall Meeting 2013, San Francisco, 9-13 Dec. 2013
 13. Shinji Sumimoto (富士通株式会社), “ACP: Advanced Communication Primitives for Exa-scale Systems”, VECPAR2014, Eugene, 2014.7
 14. Hidetomo Shibamura(九州先端科学技術研究所), Active packet pacing as a congestion avoidance technique toward extreme scale interconnect, International Supercomputing Conference 2014 (ISC’14), Leipzig, Germany, June 2014
 15. H. Honda(九州大学), “Performance Evaluation of Multiple-inputs Parallel Hartree-Fock Calculation,” VECPAR 2014, Eugene, Oregon, June 30-July 3, 2014.
 16. T. Takami(九州大学), M. Shimokawa, and T. Kobayashi, “Temporal Parallel Approach to Nonlinear Problems with Multiple Time-scales” Dynamics Days Europe, Bayreuth, Germany, 8-12 Sep. 2014
 17. R. Susukita(九州大学), “Pure Communication Time in Parallel N-body Simulations,” The 6th East-Asian Numerical Astrophysics Meeting, Suwon, South Korea, September 15-19, 2014.
 18. Yuichiro Ajima(富士通株式会社), “Distributed Dynamic Data Structure API over A Novel Low-Level Communication Layer”, PGAS2014, Eugene, 2014年10月6日~10日
 19. T. Takami(九州大学), “ACP (Advanced Communication Primitives) Middle Layer”, ISP2S2, Kobe, 2-4 Dec. 2014.
 20. T. Takami(九州大学), “Temporal Parallel Approach to Configure Multiscale

Simulations for Complex Phenomena”, The 5th AICS International Symposium, Kobe, 8-9 Dec. 2014.

21. Hidetomo Shibamura (九州先端科学技術研究所), NSIM-ACE: Network Simulator for Global Memory Access, JST/CREST International Symposium on Post Petascale System Software (ISP2S2), Poster session, Dec.2014
22. Hiroaki Honda (九州大学), Development of ACP Middle Layer Communication Library for Molecular Orbital Calculation International Congress of Quantum Chemistry 2015 Satellite Symposium, 2015.6
23. Toshiya Takami (九州大学), Parallelization of Irregular Sparse Computations with One-sided Communications, HD³-2015, 2015.12
24. K. Fukazawa, R. J. Walker, A Simulation Study of the Dependence of the Vorticity in Saturn’s Magnetosphere on the IMF, AGU Fall Meeting 2015, 2015.12
25. Ryutaro Susukita (九州大学), Yoshiyuki Morie, Takeshi Nanri, Hidetomo Shibamura, NSIM-ACE: A Simulator for Evaluating RDMA on Interconnection Networks, 2nd Annual Meeting on Advanced Computing System and Infrastructure, 2016.1
26. Fukazawa, K. (京都大学), T. Takami, T. Soga, Y. Morie, and T Nanri, “Effective Calculation with Halo communication using Halo Functions”, Proceedings of the 23rd European MPI Users’ Group Meeting. ACM, New York, NY, USA, 2016

(4)知財出願

① 国内出願 (1 件)

1. 資源使用量集計プログラム、資源使用量集計方法及び資源使用量集計装置、秋元秀行、安島雄一郎、三浦健一、岡本高幸、安達知也、住元真司、富士通、JST 企、2013 年 1 月 31 日、2013-016970

② 海外出願 (0 件)

③ その他の知的財産権

(5)受賞・報道等

① 受賞

情報処理学会計算機アーキテクチャ研究会若手奨励賞、深沢圭一郎、2013 年
情報処理学会山下記念賞、南里豪志、2014 年

* VECPAR 2014 Best Poster Award、Shinji Sumimoto、Jul 1、2014。

日本シミュレーション学会 Outstanding Presentation Award、薄田竜太郎、2017 年

② マスコミ(新聞・TV等)報道

③ その他

(6)成果展開事例

・デモンストレーション

Yoshiyuki Morie, Hiroaki Honda, Takeshi Nanri, Taizo Kobayashi, Hidetomo Shibamura, Ryutaro Susukita, Yuichiro Ajima, “Memory Efficient One-Sided Communication Library “ACP” in Globary Memory on Raspberry Pi”, The 36th IEEE International Conference on

①実用化に向けての展開

<公開可能なもの>

- ・開発した通信ライブラリ「ACP」について、プロジェクト HP(URL: <http://ace-project.kyushu-u.ac.jp>)にて公開中
- ・ SC14, SC15, SC16 にて、開発した通信ライブラリ ACP のチュートリアルを開催。

②社会還元的な展開活動

- ・ 本プロジェクトのチャネルインタフェースに関する成果をもとに、MPI の仕様を策定する MPI Forum に対して、プロセス間接続の確立と解放を明記することによる省メモリ化の効果について提言を行った。
- ・ 世界的に利用されている通信ライブラリ Open MPI の開発グループに対して、本プロジェクトにおけるメモリ消費量に関する解析結果を示し、当該ライブラリの実装の省メモリ化に寄与した。

§ 5 研究期間中の活動

5.1 主なワークショップ、シンポジウム、アウトリーチ等の活動

年月日	名称	場所	参加人数	概要
2015/8/18	ワークショップ IME2015 津和野	津和野	20 人	計算科学、計算機科学および教育に関するワークショップ
2015/10/29 、 30	LENS2015 International Workshop	東京	60 人	HPC におけるプログラミング言語、ネットワークおよびシステムソフトウェアに関する国際ワークショップ
2016/1/8	オーガナイズドセッション「About Parallel-in-Time Methods ～時間並列計 算法～」(環瀬戸内応用 数理研究部会第 19 回シ ンポジウム)	日田	20 人	時間並列計算法に関するオーガナイズドセッション
2016/6/7	オーガナイズドセッション「PGAS プログラミ ング環境」	仙台	40 人	PGAS プログラミング環境に関するオーガナイズドセッション

§ 6 最後に

• ACP ライブデモの様子 (SC15 九州大学ブースにおいて)

