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

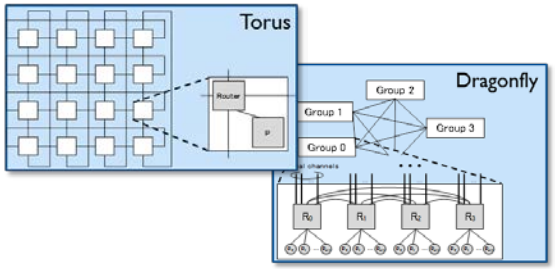
九州大学:
南里豪志、高見利也、深沢圭一郎、本田宏明、薄田竜太郎、森江善之
富士通株式会社:
住元真司、安島雄一郎、三浦健一、岡本高幸、秋元秀行、佐賀一繁、
安達知也、野瀬貴史、今出広明、神林亮
財団法人九州先端科学技術研究所:
柴村英智、曽我武史

2013年10月11日
CREST「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」領域会議

研究のねらい

- エクサスケール計算環境での利用に耐える、
  「スケーラブルな通信ライブラリ」
  の開発

- 想定する計算環境：
  数十万ノードの多次元トーラスもしくは high-radix網（Dragon-Fly等）
  - コア数：数十コア～数百コア / ノード
  - メモリ：　数百GB / ノード
  - プログラミングモデル：MPI + X (スレッド or PGAS)
  - プロセス数：～数十プロセス / ノード



通信ライブラリへの性能要求 ⇒ 増
プロセス当たり通信用ハード資源（メモリ, リンク, etc）⇒ 減

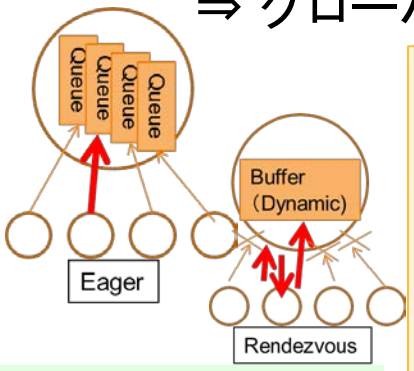
エクサスケール計算環境における通信ライブラリの要件：
省メモリ通信の低遅延実装
アプリケーションやシステムの状況に応じた資源の有効利用

ACE (Advanced Communication for Exa) ライブラリの開発

- 省メモリ・低遅延通信のためのグローバルメモリ管理モデル
  - 必要な時に、必要な場所に、必要なだけメモリを配置し、参照可能なメモリ空間
- プログラムやシステムの状況に応じた通信インタフェースの動的最適化技術
  - アルゴリズム、プロトコル、パラメータ等の適応的選択
- アプリケーションの計算・通信パターンに応じたチューニング技術
  - 定型計算・通信パターンの効率的な実装

グローバルメモリ管理モデル

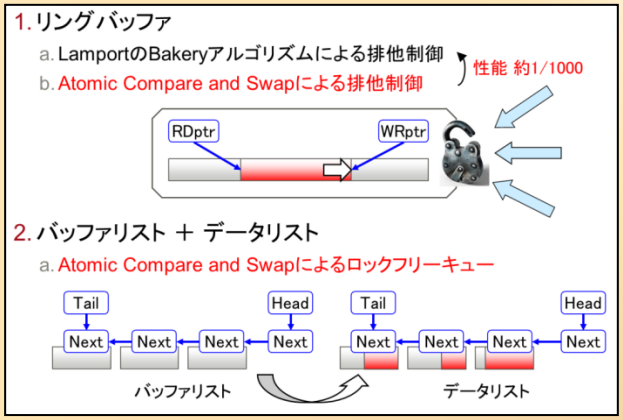
- 省メモリ・低遅延通信実現のためのメモリモデル
  - 各プロセスがローカルに配置した領域をグローバル領域として参照可能
    ⇒ グローバルなデータ構造（キュー、リスト等）の構築が容易



従来のプロトコル

グローバルメモリを用いた省メモリ・低遅延メッセージパッシング通信の例（昨年度の本プロジェクトの成果）

- **共有受信キューにデータ送信（リングバッファ or リスト）**
- **既存の Eagerや Rendezvousと違い、各プロセスでの受信キューの配置や送受信プロセス間同期が不要**



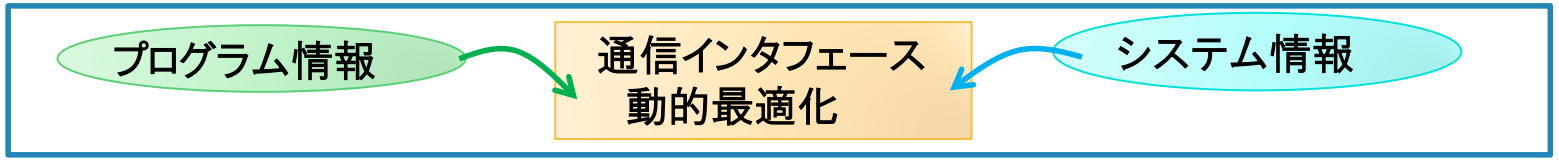
- 従来のモデルとの比較
  - メッセージパッシング
    - 送信側と受信側でメッセージを管理するためグローバルなデータ構造構築に向かない
  - PGAS
    - グローバルなメモリ割り当てやオブジェクト生成は全プロセスで同期実行する
    - プロセスがローカルに配置した領域をグローバルに参照させるのは困難

通信インタフェースの動的最適化技術

- 状況に応じた通信インタフェース実装の動的な調整
  - 参照可能な情報の活用

システム情報）
  - プロセス配置、トポロジ、ネットワーク性能、負荷状況、etc.

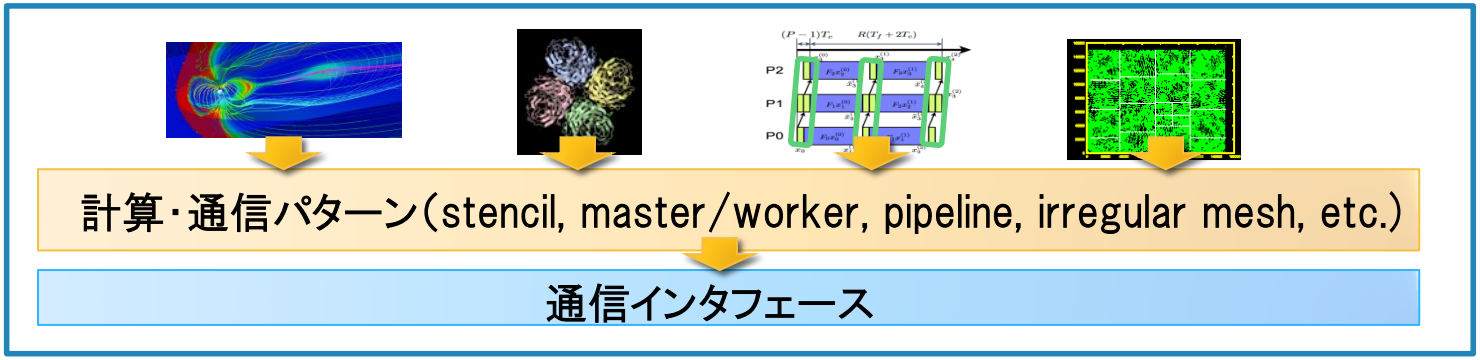
プログラム情報）
  - プログラマによるヒント情報
    - プログラム中で特に高速化が必要な領域の指示、プログラムのメモリ使用量、同時に進行する非ブロッキング通信の数、通信関数呼び出しの繰り返し回数、etc.



- 動的最適化技術の例）
  - 性能的に重要な通信へのバッファの優先的割り当て
  - 繰り返し回数の多い通信に対する動的アルゴリズム選択技術の適用
  - メモリの残量に応じたバッファ割り当て
  - 負荷状況や通信パターンに応じたシステムパラメータの調整

並列計算フレームワーク

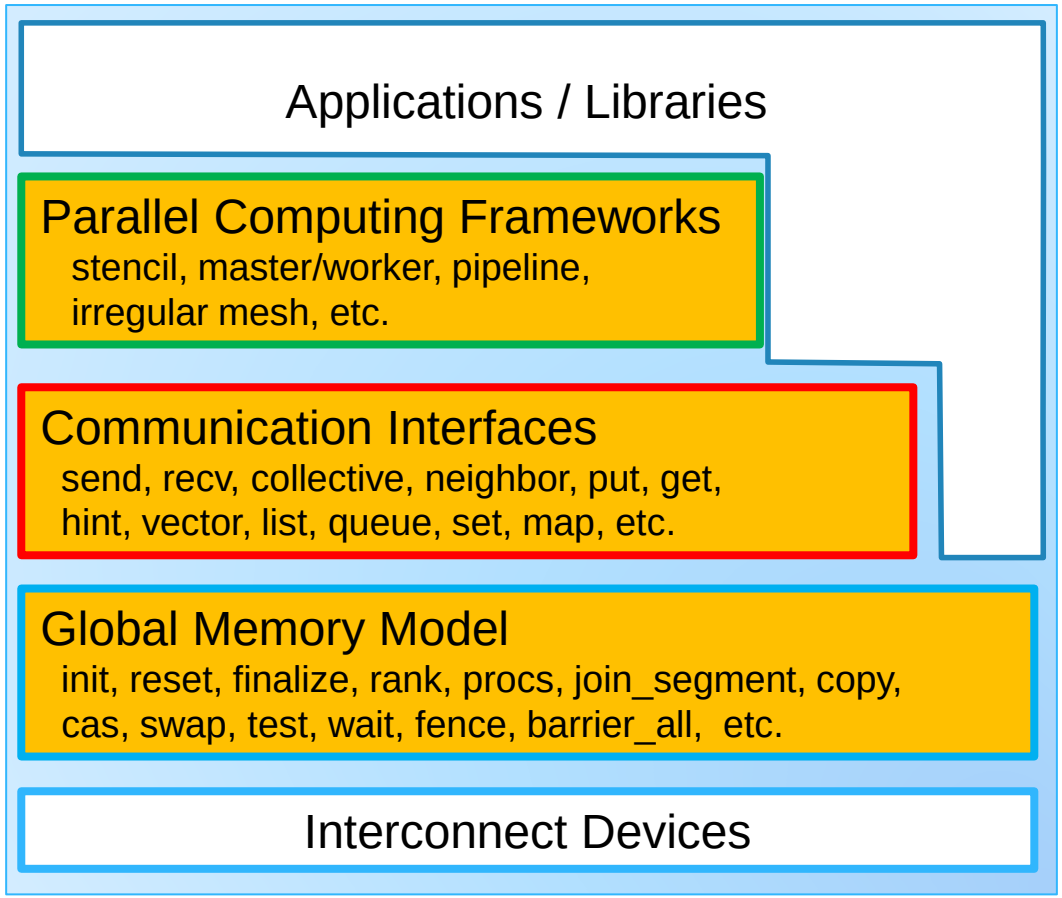
- 定型計算・通信パターンのインタフェース提供
  - 既存のアプリケーションからパターンを抽出
  - stencil計算、master/worker、並列パイプライン、不規則格子、etc.



- 通信ライブラリを効率的に利用した実装
  - 個々の通信命令ではなく、一連の通信と計算の組み合わせに対するチューニング技術
    - チューニングの例)
      - 通信の隠蔽、パターンに応じた適切な通信アルゴリズム選択、局所性や通信衝突を考慮したデータ配置やプロセス配置、適切なプログラムのヒント情報の提供、etc.

ACEライブラリの構成

- 並列計算フレームワーク
  - 定型の計算・通信パターンを高速実装
  - 一般的なアプリケーションプログラマ向け
- 通信インタフェース
  - 双方向通信、片側通信、集団通信、隣接通信、ヒント、グローバルデータ操作
  - エキスパートプログラマ向け
- グローバルメモリ管理モデル
  - グローバルメモリ空間の確保とアクセス
  - 通信ライブラリ/並列言語処理系の開発者向け

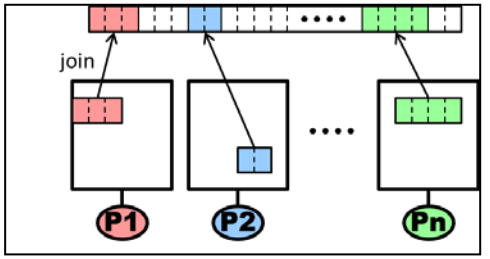


現在までの主な成果

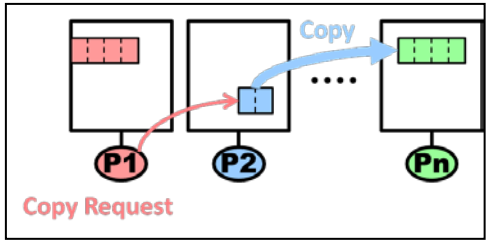
- グローバルメモリ管理モデル
  - インタフェース設計
  - メモリ使用量解析ツール
- 通信の動的最適化技術
  - ヒントインタフェース提案
  - 隣接通信アルゴリズム
  - パケットペーシングの実機検証
- 並列計算フレームワーク
  - 実アプリケーションにおける計算・通信パターン調査
    - 時間並列プログラムの並列パイプライン
    - MHD計算の stencil 計算
    - FMO計算の master/worker モデル
    - 多体問題の不規則メッシュ通信

グローバルメモリ管理モデルの仕様設計

- グローバルアドレス空間
  - 各プロセスがセグメントを確保してグローバルメモリ空間に提供



- グローバルメモリ参照
  - 任意のプロセス間のデータコピー



- インタフェース
  - セグメント操作：join, disjoin
  - データコピー: copy
  - アトミック参照: cas, swap, add, xor, or, and
  - 完了保証: test, wait
  - 順序保証: fence
  - 同期: barrier
  - 各種問い合わせ: rank, procs, query_address

外部発表：

1. Y. Ajima, et al, "Asynchronous Global Heap: Stepping Stone to Global Memory Management", PGAS2013, 2013.10.
2. 住元, et al, "遠隔Atomic通信を用いた省メモリ性実現のための方式検討", 138回HPC研究会, 2013.2.
3. 安島, et al, "非同期グローバルヒープの提案と初期検討", 138回HPC研究会, 2013.2.
4. 安島, et al, "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", SWoPP2013, 2013.7.

ライブラリの動的メモリ使用量測定ツール DMATP-MPI

- 関数のフック機能を用いたメモリ使用量測定
  - メモリ配置関数(malloc, realloc, memalign)、解放関数(free)をフック
  - 各関数の呼び出し元(メインプログラム or ライブラリ)、及びメモリ使用量の増加分(or 減少分) を記録
  - 呼び出し元毎に結果を集計

通信ライブラリでのメモリ使用量を抽出

```
---- Statistics of individual thread memory usage ----------
Thread ID: 3517
mem_size = 773,720, mem_max = 547,195,816
malloc: 970060, realloc: 518, memalign: 786, free: 967877

  ---- Statistics of individual library memory usage -------
  Library: /home/akimoto/OpenMPI/lib/libmpi.so.1
  mem_size = 791,680, mem_max = 43,866,296   MPI使用分
  malloc: 969573, realloc: 518, memalign: 786, free: 967395
  ------------------------------------------------------------
  Library: /lib64/libc.so.6                  アプリ使用分
  mem_size = 384, mem_max = 503,329,424
  malloc: 483, realloc: 0, memalign: 0, free: 477
  ------------------------------------------------------------
```

外部発表:

1. S. Sumimoto, et al., "Dynamic Memory Usage Analysis of MPI Libraries Using DMATP-MPI", EuroMPI2013, 2013.9.
2. 住元, et al, " DMATP-MPIを用いたMPIライブラリの関数別メモリ使用量評価", 138回HPC研究会, 2013.2.
3. 秋元, et al, "DMATP-MPI: MPI向け動的メモリ割当分析ツール",138回HPC研究会, 2013.2.
4. 秋元, et al, "MPI向け動的メモリ割り当て分析ツール", SACSIS2013, 2013.5.

ヒントインタフェース提案

- プログラムの情報を通信ライブラリに提供する Hint関数
  - 例）性能的に重要な kernel regionの範囲を指示

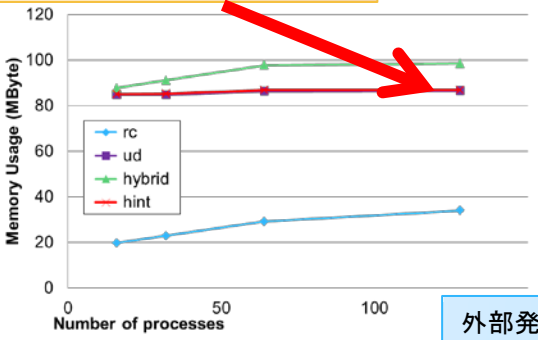
```
main()
{
  MPI_Init();

  for (r = 0; r < 100){
    for (i = 0; i < 100; i++)
      for (p = 1; p < procs; p++){
        MPI_Isend(sbuf, 64, MPI_CHAR, (myrank+p)%procs, 0, MPI_COMM_WORLD, &
        MPI_Recv(rbuf, 64, MPI_CHAR, (myrank-p+procs)%procs, &stat);
        MPI_Wait(&req, &stat);
    }
    Hint("IN_KERNEL", 1);
    for (i = 0; i < 10000; i++){
      MPI_Isend(sbuf, m, MPI_CHAR, (myrank+1)%procs, 0, MPI_COMM_WORLD, &req
      MPI_Recv(rbuf, m, MPI_CHAR, (myrank-1+procs)%procs, &stat);
      MPI_Wait(&req, &stat);
    }
    Hint("IN_KERNEL", 0);
  }
  MPI_Finalize();
}
```

準備フェーズ：
Alltoall or Allreduce
遅くて構わない

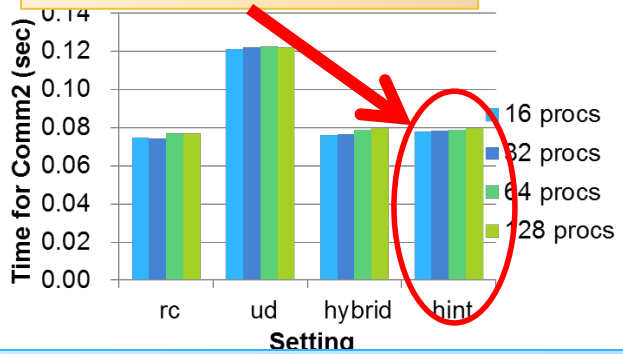
kernel region:
シフト通信
高速化したい

- kernel region内の通信だけに優先的に通信バッファ割り当てた場合

メモリ使用量：
プロセス数によらず一定



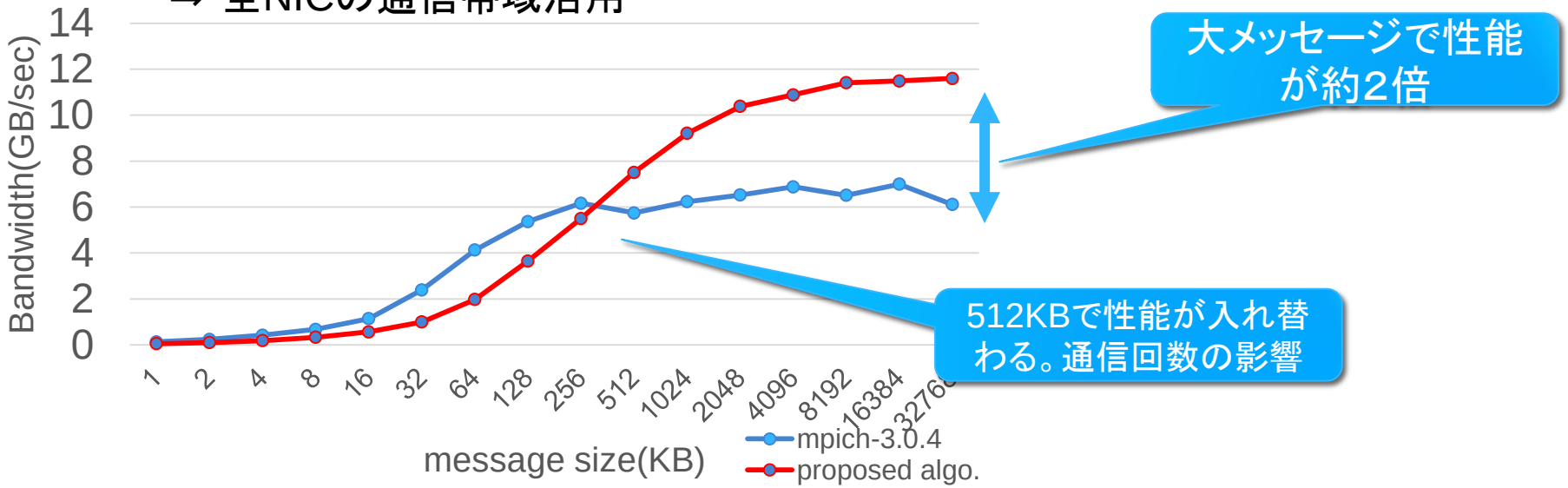
通信時間：最速を維持



外部発表：
1. 南里, et al, "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HPC研究会, 2013.10.
2. T.Nanri, et al, "What Communication Library Can do with a Little Hint from Programmers?", MVAPICH User Group Meeting'13, 2013.8,

隣接通信

- ネットワークアーキテクチャを考慮した隣接通信手法
  - ネットワークトポロジやNIC数を考慮
  - 論理トポロジが物理トポロジと同じ場合
    - NIC数 ＜ 隣接プロセス数 : メッセージを分割して NICに等分
      ⇒ 全NICの通信帯域活用



  - 論理トポロジが物理トポロジと異なる場合
    - NIC数 ＜ 隣接プロセス数 : 通信順序の調整により衝突回避 (予定)
- 隣接通信向けランク配置最適化
  - リンクで要求される最大通信数の最小化

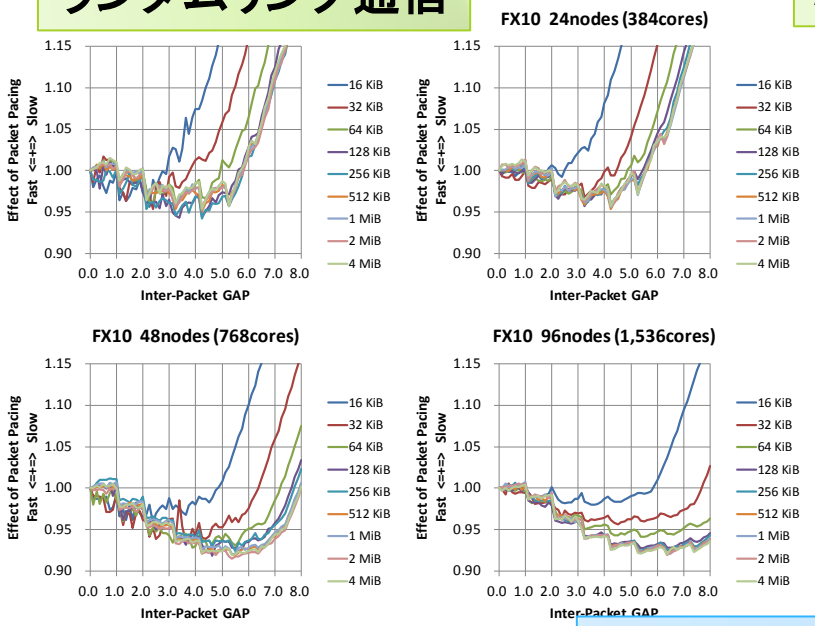
外部発表:

1. Y. Morie, et al, "Implementation of neighbor communication algorithm using multi-NICs effectively by extended RDMA interfaces", SC13 (poster), 2013.11 (to appear).
2. 森江, et al, "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 ACS, 6(3), 2013.9.

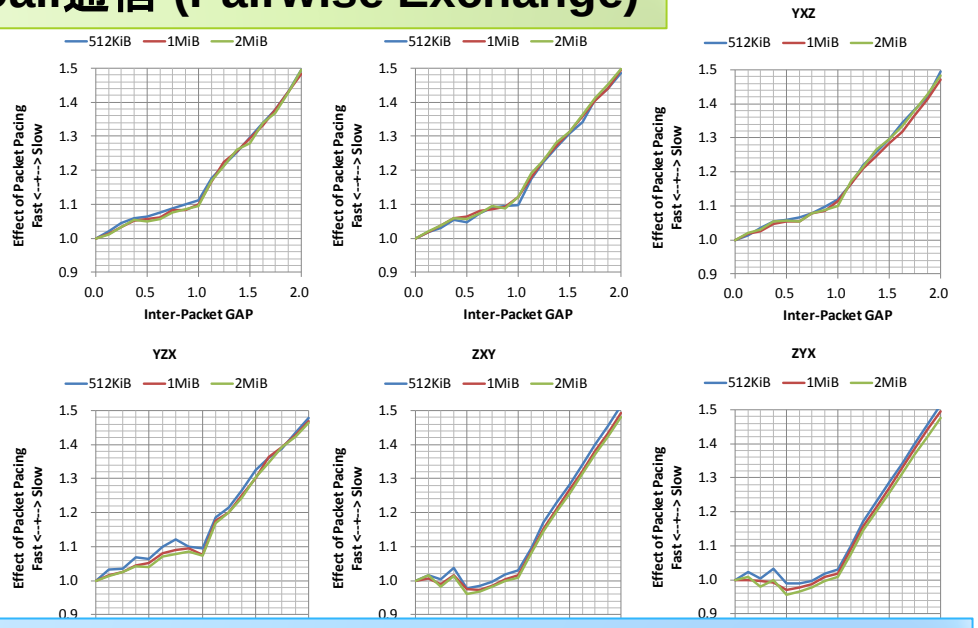
パケットペーシング技術の実機検証

- パケットペーシング技術
  - 特定のリンクにトラフィックが集中する高トラフィック通信において パケットの送信間隔を調整することにより衝突を回避
- 実験結果
  - 実験環境: Fujitsu PRIMEHPC FX10 （Tofuインターコネクト）
  - 実機でのパケットペーシング技術の効果を確認
  - ネットワークシミュレータ NSIM による予測結果ともほぼ一致

ランダムリング通信



Alltoall通信 (Pairwise Exchange)

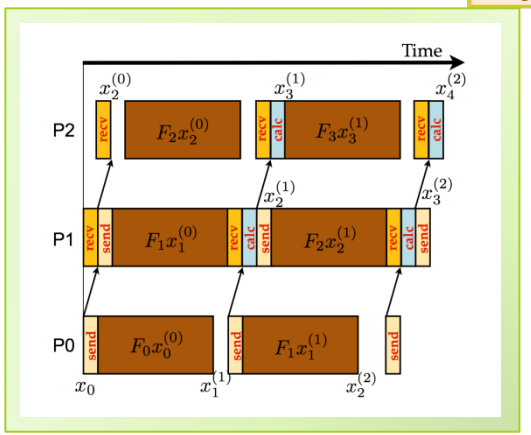


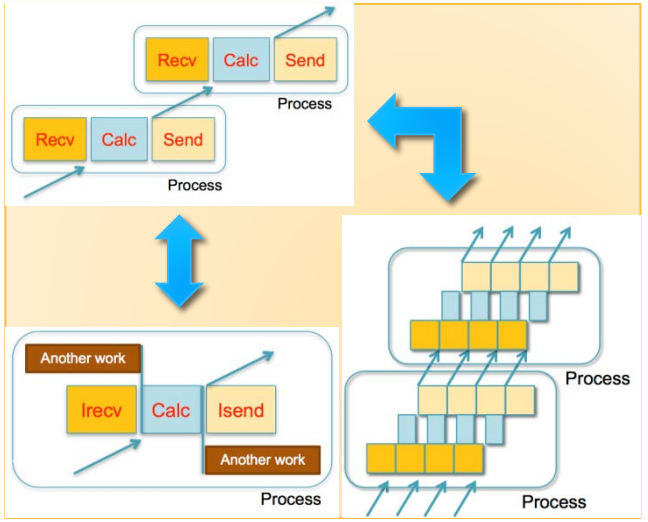
外部発表:

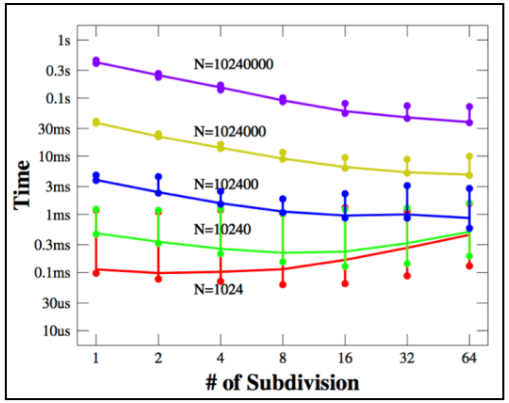
1. 柴村, et al, "FX10におけるパケットペーシングを用いたアプリケーションの通信性能評価", 第141回HPC研究会, 2013.10.

時間並列プログラムの並列パイプライン

- 時間並列プログラム
  - 時間発展計算や依存関係のある繰り返し計算を、時間方向・依存関係のある方向に並列化
    - 常微分・偏微分方程式の初期値問題
    - 線形方程式の反復解法
- 成果：並列パイプラインによる実装の提案と評価
  - 親プロセスからの受信、計算、子プロセスへの送信をパイプライン化
  - メッセージを細分化して通信と計算をオーバラップすることによる効果を確認
- 今後の予定
  - 実アプリケーションへ適用
  - ACE通信インタフェースでの効率的な実装と動的最適化技術の検討



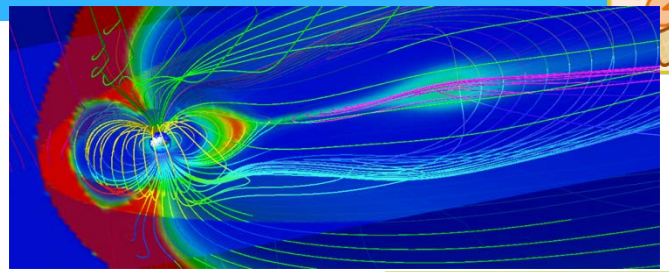


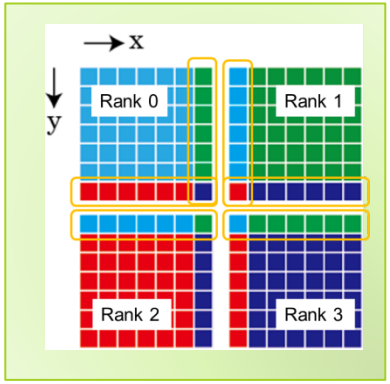


外部発表：

1. T. Takami, et al, " An Efficient Pipelined Implementation of Space-Time Parallel Applications", PARCO2013, 2013.9.
2. T. Takami, et al, "A simple implementation of parareal-in-time on a parallel bucket-brigade interface", PPAM2013, 2013.9.

MHD（電磁流体）計算の stencil 計算





- MHD計算：電磁力を含んだ流体シミュレーション
  - 主にプラズマの振る舞いを調査
  - 陽的差分法、領域分割法で並列化

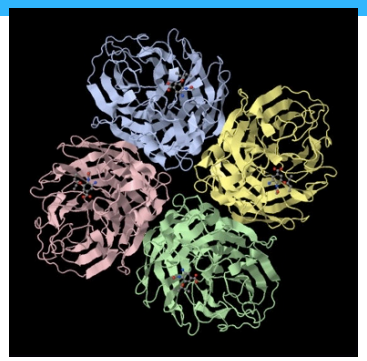
- 成果： 様々なアーキテクチャでの性能評価と性能モデルの構築
  - アーキテクチャ毎に最適な領域分割次元を調査
  - メモリ性能の pack/unpack速度への影響を指摘
  - 高並列化時の性能予測モデル

- 今後の予定
  - stencil計算の ACE通信インタフェースでの効率的な実装と動的最適化技術の検討
    - pack/unpack, 通信順序, ランク配置の効率化
    - 領域分割次元、のりしろ領域の幅への対応

外部発表：

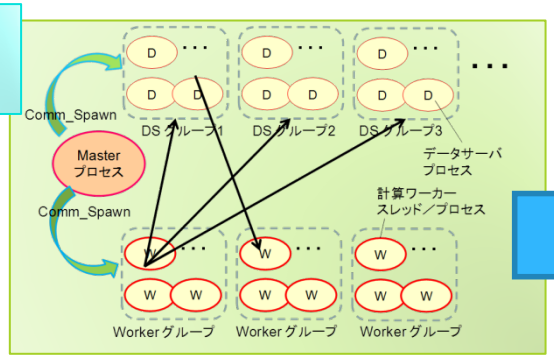
1. K. Fukazawa, et al, " Performance Measurements of MHD Simulation for Planetary Magnetosphere on Peta-Scale Computer FX10 ", PARCO2013, 2013.9.
2. K. Fukazawa, et al, ". Performance of Large Scale MHD Simulation of Global Planetary Magnetosphere with Massively Parallel Scalar Type Supercomputer Including Post Processing", HPCC, 2012.11.

FMO計算のmaster/workerモデル

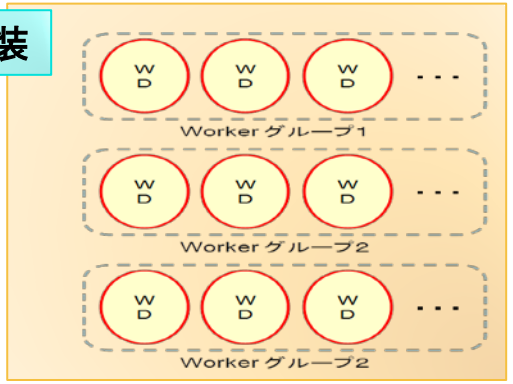


- FMO計算： master / worker モデル
  - 大規模分子の量子科学計算をタスク並列処理
  - 成果：グローバルメモリ管理モデルに向けた実装
    - グローバルカウンタ、グローバル配列を ARMCIで試験実装
      - 従来は MPIで専用プロセスを割り当てて実現

MPIでデータサーバ専用プロセスにより実装

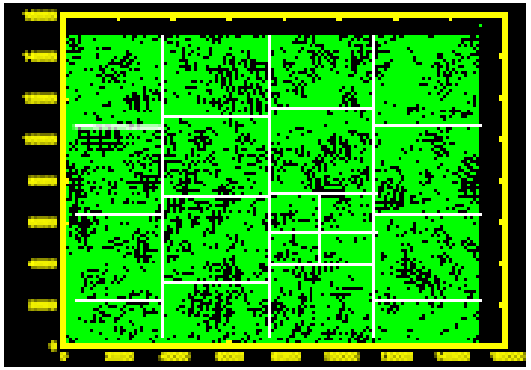


ARMCIにより実装



- 今後：
  - ACE通信インタフェースを用いたグローバルカウンタ、グローバル配列の効率的な実装
  - 負荷分散とデータ配置の最適化技術

多体問題の irregular mesh通信



- 重力N体シミュレーション、荷電粒子系シミュレーション等を Adaptive Mesh法により計算
  - ⇒ 負荷バランスに応じて不規則な空間分割
  - ⇒ 複雑な隣接通信

- 成果: P^3M法の調査
  - 空間分割手段
    - X, Y方向は固定し、Z方向の分割幅で負荷調整
  - 隣接通信手法
    - 分割情報からの隣接プロセス算出、データの pack/unpack、非ブロッキング送受信、wait
  - 性能評価
    - Z方向の調整だけでは特定のセルに粒子が集中するため十分な負荷均等化が困難:
      ⇒ 高密度領域での負荷均等化手法の検討 ⇒ さらに複雑な隣接通信

- 今後:
  - 不規則な隣接通信のためのインタフェース設計
  - ACE通信インタフェースを用いた効率的な実装
  - ランク配置に応じた通信最適化
  - 動的最適化技術を用いた負荷均等化

今後の予定

- グローバルメモリ管理モデル
  - 各デバイス向け実装
    - Ethernet, InfiniBand, Tofu
  - グローバルデータ構造の構築
    - リスト、キュー、グローバル配列、etc.

- 動的最適化技術
  - グローバルメモリ管理モデル上のACE通信インタフェースの設計と実装
  - 各インタフェースの動的最適化技術開発
  - 各最適化技術を統合した動的最適化機構の構築

- 並列計算フレームワーク
  - 各フレームワークのインタフェース設計
  - ACE通信インタフェース上の実装
  - 最適化技術の開発、および性能評価