超ロバスト並列計算

小柳義夫 須田礼仁 西田晃
情報理工学系研究科コンピュータ科学専攻

概要

サブプロジェクト「超ロバスト並列計算」では，グリッドのようにネットワークや計算機の構成・性能が動的に変動する並列計算環境において，ネットワークや計算機の故障や追加・負荷の変動などの外乱に対しロバストに対応し計算能力を効率よく引き出す手法の開発をめざす．すなわち，数値アルゴリズムと並列化手法の両面から，性能（計算機資源の利用効率）のロバストネスを導くことが研究の目的である．本報告では平成 16 年度の成果の概要と今後の計画について略述する．

1 研究の背景・経過と全体計画

計算機をつなぐネットワークの高性能化と急速な広がりは目覚しい．研究・開発レベルのみならず，一般家庭にも高速・大容量のネットワークが普及しており，今後のさらなる展開が期待される．その展開の形として最も期待されているのが複数の組織やネットワークを結合する技術であるグリッドであり，その基盤技術・システムの研究・開発・標準化と並行してアプリケーションの開発が活発化している．

全世界規模の計算機ネットワークは，極めて大規模な並列計算機であり，総合的な計算性能と記憶容量とを有効に利用することができれば，科学と技術のさらなる発展に寄与することは疑う余地がない．しかしこのようなネットワーク計算環境は従来の単体スーパーコンピュータとはまったく異なるアーキテクチャである．従来の並列処理では均一な仕様の計算機を想定していたのに対し，グリッドなどでは不均一な仕様の計算機が結合されている仮定のほうが自然である．また従来の並列処理では計算機・ネットワークとも占有していたのに対し，グリッドなどでは多くのユーザと共有する計算機を汎用・共有のネットワークで接続した形態も多く，実効的な性能は動的に変化するものと考えられる．さらに，システム的にまた地理的に離れた計算機を非常に多数結合しているため，動的な構成の変化や故障の可能性についても考慮する必要がある．

このような新しい並列計算環境の台頭と普及を受け，そのポテンシャルを最大限に活かし利用するためには，従来とは異なる新しい並列化の手法が必要である．我々のサブプロジェクト「超ロバスト並列計算」では，並列プログラミングの概念と手法を根本的・全面的に刷新し，21 世紀の科学技術計算の発展を支える高性能並列計算の実現の礎となる並列化手法の総合的な開発を目指している．

実験環境の整備・既存研究の調査を中心に，本 COE の「大規模ディペンダブル情報基盤プロジェクト」の田浦研究室との情報交換などを行った平成 14 年度に続いて，平成 15 年度には不均一な計算環境に適応できるロバストな集団通信とデータ再分散の手法を開発し，簡易かつロバストに並列性能を引き出す手法である ERXPP の提案を行った．そして今年度の主な成果としては不均一な計算機・ネットワーク環境をも考慮し並列化オーバーヘッドを抑えることができる新しい並列計算機へのデータマッピング（分割・割付手法）の開発，ならびに予測しない計算機・ネットワークの停止に際しても停止することなく計算を継続できる並列計算ソフトウェアシステムの開発がある．今後はこれまでに開発してきた手法の集成とアプリケーションでの実証のほか，動的に変化する実行性能の検出と効率的な適応のための技術の開発，予測しない遅延に対して性能劣化の少ない並列処理スケジューリングの研究などを行う計画である．

2 本年度の成果

本節では平成 16 年度の主な研究成果とそれに関する今後の研究計画などについて報告する．

まず，多様な並列計算環境で効率的な並列計算

を実現できるよう改良されたデータマッピングの手法と，その分子動力学アプリケーションへの適用評価について報告する．

次に，計算機やネットワークの予期せぬ停止に至った際に，できるだけ停止することなく失われた情報を回復し残っている計算資源で計算を継続することが比較的容易に実現できるような並列計算のモデルとシステムについて報告する．

2.1 ロバストデータマッピング

多くの科学技術計算において並列化は欠かせない．一方並列計算機は多様化しており，ネットワークの性能・トポロジーが不均一である場合も多い．このような環境では均一性を仮定した既存手法の多くは負荷の不均衡を起こす．

本研究では計算に必要なデータの依存関係がわかっていて，計算と通信がグラフで表現できる場合を想定する．すなわち，グラフの各頂点に計算負荷，各枝に通信量を表す重みを与え，目的のプロセッサ数にグラフの頂点を分割することにより，データマッピングが行えるものと仮定する．このとき，各プロセッサに割り当てられた頂点の重みを均等にし，プロセッサをまたがる枝の重み（枝カット）が最小になるようにすることにより，計算負荷が均等になり，データの局所性が抽出され通信が削減され，高い並列性能が期待できる．

しかし，従来手法には，(1) 枝カットは総通信量に正しく対応していない，(2) 個々のプロセスの通信量を低く抑えることが重要なのに総通信量で評価していた，(3) パーティションのプロセッサへの割り当てまで考慮されなかった，という 3 つの欠点があった．そこで我々はこれらの問題点を解決する手法を提案する．

2.1.1 提案手法

まず，通信量の正確な見積もりのために，枝に通信するべきデータが関連付けられている計算グラフを定義した．通信時間は遅延とバンド幅でモデル化されることが多いが，通信遅延の一部はほかの通信で隠蔽でき，また通信が多い場合に重要となるのがバンド幅であるため，各資源のバンド幅で通信のモデル化を行った．

並列計算機は通信資源をもとにして階層的に表現し，それに対応して計算グラフを再帰的な二分割を繰り返し，データ分割を決定する．これにより，ネットワーク構造に適したデータ分割が可能になる．

通信コストは，個々の資源によって伝送されるデータ量を抑える点を考慮し，Kernighan-Lin Fiduccia-Mattheyses アルゴリズムを拡張した手法で最小化する．これにより，個々のプロセスの通信量を均衡化させるだけでなく，通信が集中するような資源を用いる通信を抑制することができる．

2.1.2 分子動力学法での実証実験

提案手法を生体分子の分子動力学法プログラム MolTreC2-DM に実装した．MolTreC2-DM では，空間を小セルに分け，個々のプロセスが直接データの通信を行う．

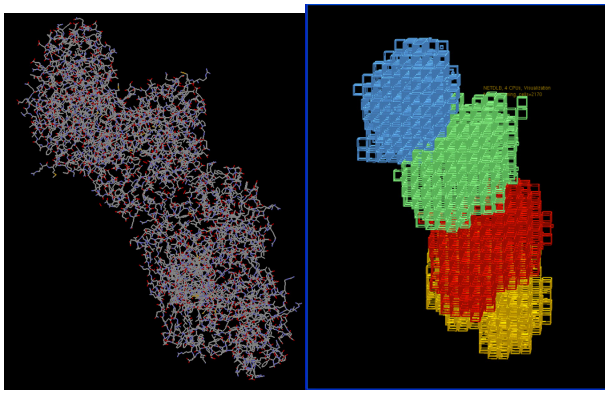

図 1　PDB-1XS2 とその 4 分割の例

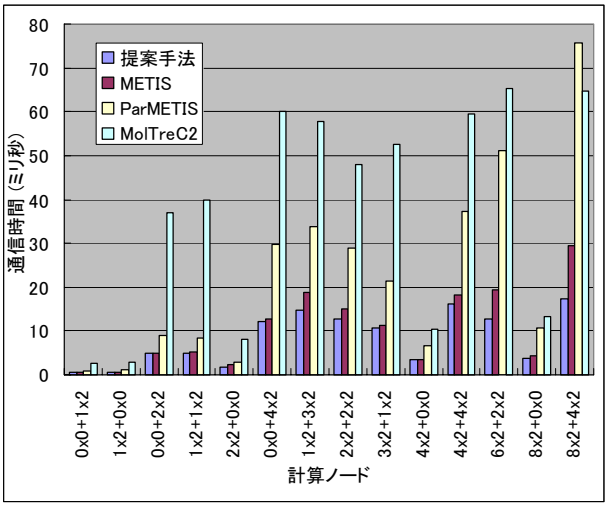

図 2 ヘテロなクラスタでの通信時間

図 1 は提案手法による分割の例である．図 2 に

分子動力学法を2つのクラスタをつないだシステムで実行した場合の1反復における通信時間を示す．2 つのクラスタをつなぐ回線に通信が集中するが，提案手法では通信時間が短くなっている．

提案手法により，計算負荷を均等に保ちながら通信時間を大きく削減できることが示された．この手法によりロバストな並列性能が期待でき，また多くの科学技術計算が同様の計算パターンを持つため，多くのアプリケーションで同様の効果が期待できる．

2.2 故障からの回復処理の柔軟性の実現

汎用の製品を組み合わせたクラスタやグリッドなどの並列計算環境では，故障が発生する確率が従来のスーパーコンピュータに比べて格段に高い．特に長時間の計算を必要とする大規模な科学技術シミュレーションや最適化などでは，計算の途中で発生した故障によりそれまでの計算が無駄になることを防ぐために，一定の間隔で計算途中のデータすべてを保存しておくチェックポインティングが用いられることも多い．このとき通常は，故障部分を修理し，それが完了してから最新のチェックポイントから計算を再開することになる．

しかし，多数の計算機のうち 1 台のみが故障して，残りの計算機が使える状況であったとしても，修理してからでなければ計算が再開できないというのはあまりに原始的である．故障に備えて予備の計算機を準備しておく方法もあるが，複数資源の故障に対応するには，多くの予備資源を，発生するかどうかわからない故障のために準備しなければならないという欠点がある．

むしろ一部が故障しても使える計算資源だけで計算を再開したい．しかも，最小限のコストで人間の操作なしに失われた情報を回復させて，あたかも（回復のための計算・通信・記憶コスト以外は）何事もなかったかのように計算を継続させられるのが望ましい．

このような並列プログラムを書くことは MPI などのミドルウェアが適当に対応していれば原理的には可能であり，FT-MPI と呼ばれるシステムはそのようなプログラムを可能に（かつ強制）するMPIの拡張である. しかしそのためにはMPIのライブラリ呼び出しの一つ一つに対してあらゆる故障への対応をプログラムとして作りこまなければならず，実際にアプリケーションプログラムを上記のような理想的な形に書き改めるのは限りなく不可能に近い．そもそもいつ発生するかわからない故障への対応と回復のプログラムを完全に書くことは極めて難しく，一般のプログラマに要求するのは無理がある．

そこで我々は，故障の検出とそこからの回復はシステムが自動的に行うが，その方法はアプリケーションプログラムから柔軟に制御できるような方式を提案する．

2.2.1 提案方式：ドメイン境界ログ

我々の提案方式「ドメイン境界ログ」はメッセージログと呼ばれる従来手法の拡張となっている．提案手法では，計算全体を「ドメイン」と呼ばれるいくつかの部分に分割し，ドメイン間の情報の流れをログとして保存する．故障が発生した場合，故障に関係するドメインの計算は保存されたログから回復でき，故障に関係しないドメインの計算は（回復中の待ち時間が発生することを除いて）停止することなく継続される．故障の検出と回復の処理はアプリケーションプログラムが感知することなく，システムが自動的に行うことができる．

従来手法のメッセージログ方式は，各プロセッサの担当する計算をひとつのドメインと定義したドメイン境界ログとみなすことができる．故障からの回復時にドメインを再定義することはできないため，メッセージログ方式では回復時に故障前と同じ数のプロセッサが必要であり，また同じデータ分散を行わなければならなかった．これに対し我々が提案するドメイン境界ログでは，ドメインの計算内容が同じであれば，具体的な実装方法は故障前と故障後で異なっていてよい．すなわち，異なるプロセッサで，異なるデータ分散で，あるいは異なるアルゴリズムで，故障からの回復を行うことが可能である．

2.2.2 ドメイン境界ログの並列プログラミング

我々は提案するドメイン境界ログの正当性の確認とシステム・アプリケーションの構築の例証のためにプロトタイプシステムを開発した．ここではアプリケーションプログラムの並列処理モデルに関して簡単に紹介する．

プロトタイプシステムは Java で書かれており，アプリケーションも Java で構築する．計算の主要部分はドメインとして定義するが，これは DomainBody のサブクラスとして実現する．このクラスの doExecute メソッドにドメインの計算内容を実装する．このメソッド内では send と

recv というメソッドを用いてドメイン内のプロセスとメッセージ通信ができるほか，他のドメインと BoundaryCommunicationObject (BCO) を用いてメッセージの交換を行うことができる．BCO は自動的にログに保存され，故障からの回復処理時にはシステムにより自動的に再実行される．

故障はシステムにより自動的に検出されるが，その際各プロセス上にリカバリスレッドが生成される．リカバリスレッドは回復が必要なドメインの情報をシステムから得て，それらのドメインのインスタンスを再構築する．このときに故障前のインスタンスと異なる実装を採用することが可能であり，リカバリスレッドを適当にプログラムすることによりユーザが故障からの回復処理を制御することが可能となる．

構築したプロトタイプシステム上で偏微分方程式を並列に解くプログラムを作成し，プロセッサが計算中に故障しても残ったプロセッサだけで計算を継続することができることを実証した．

3 まとめ

本報告では超ロバスト並列処理プロジェクトの平成 16 年度の研究開発の成果について報告した．2 節で報告したもの以外に，ヘテロプロセッサ上での密行列計算ライブラリの開発，MMDL (Multi-Master Divisible Load) モデルによるデータ再分散手法の開発なども行っている．

これらを含めたこれまで 3 年間の研究により，不均一な計算機・ネットワークで構築された並列計算環境における並列処理手法についてかなりまとまった知見が得られたものと考えている．また MMDL によるデータ再分散手法は計算環境の動的な変化に効率よく対応する手法の基礎を与えるものと期待している．さらに，故障により計算結果が失われてしまう場合にも，ドメイン境界ログの手法により効率的にデータを回復することができるようになった．今後はこれらの手法の研究をさらに推進して幅広いアプリケーションに適用可能な技術として確立することを目指す．

並列処理の性能のロバスト性を目的とする本プロジェクトにおいてまだ十分検討がなされていない課題もある．データ再分散や故障からの回復は，変化の頻度が比較的少ない「遅い」システムの変動にはある程度適切に対応することができる．しかし，変動に対処しようとしている間にシステムの状態が変動してしまうようでは意味がないわけであるから，高い頻度で変動したり，短時間だけ大幅に性能が低下したり停止したりするような状況では適切ではない．このような状況に対応するためには，予想外の遅延に対する所要時間の変動ができるだけ少なくなるように，あらかじめプログラムを工夫しておくことが必要であると考えられる．プログラムやアルゴリズムのこのような性質を耐遅延性 (latency tolerance) と呼ぶのであるが，耐遅延性に関する既存の手法・研究は十分な説得力があるものとはいいがたい．耐遅延性は従来の並列処理の基本技術であった遅延隠蔽 (latency hiding) に代わるものであり非常に重要であるので，これまでに取り上げてきたテーマについて掘り下げることと並行して，この課題に関して検討を進めたいと考えている．

主な外部発表

[1] A. Fujii, A. Nishida, and Y. Oyanagi, “The Evaluation of The Aggregate Creation Orders：Smoothed Aggregation Algebraic MultiGrid Method,” Proc. HPCSE-04 (CDROM), August 2004, Toulouse.

[2] A. Fujii, A. Nishida, and Y. Oyanagi, “Vectorized Algebraic Multigrid Algorithm for Unstructured Finite Element Problems,” VECPAR2004 (poster), June 2004, Valencia.

[3] 西田晃,「非対称固有値問題への並列 AMG 前処理付共役残差法の適用と評価」，情報処理学会研究報告，2004(81)，pp. 85-90.

[4] 須田礼仁，「マルチクラスタ環境での MMDL 漸近最適スケジューリング」，情報処理学会研究報，2004(81)，pp. 103-108.

[5] 須田礼仁，「高速球面調和関数法：アルゴリズム，応用，展開」2004 年日本応用数理学会年会予稿集，pp. 26-27.

[6] 西田晃,「非線形最適化問題としての固有値解法: 最適化手法の適用と評価」，2004 年日本応用数理学会年会予稿集，pp. 220-221.

[7] R. Suda, "Stability analysis of the fast Legendre transform algorithm based on the fast multipole method", Proc. Estonian Acad. Sci. Phys. Math., 53(2), 2004, pp. 107-115.

[8] R. Suda, "Fast spherical harmonic transform routine FLTSS applied to the shallow water test set", Mon. Wea. Rev. (to appear).