3M-9

近似アルゴリズムによるGrid計算機におけるスケジューリングの実装と評価

三ッ石勇作† 鈴木康人‡

沼津工業高等専門学校専攻科 制御・情報システム工学専攻† 沼津工業高等専門学校 制御・情報工学科‡

1. 序論

グリッドコンピュータでは性能が不均一なコンピュータを接続して運用することが前提とされる. 単純に考えれば, コンピュータの性能に応じてスケジューリングができれば, グリッドシステムを高速化できると考えられる. しかし, 割り当てを決定する問題はNP困難な問題に分類されるため, 一般的にはFirstFitアルゴリズムがスケジューリングのアルゴリズムとして採用されている.

本研究ではグリッドコンピュータにおけるスケジューリングアルゴリズムを, ビンパッキング問題を解くための近似アルゴリズム [1] を用いて作成した.

アルゴリズムを作成して, 実験を行った結果, FirstFitアルゴリズムよりも高速に問題を解くことができた. 以下に詳細について報告する.

1.1 研究背景

「グリッドコンピュータ」とは, 利用者の需要に対して必要な種類や量の端末をネットワーク内から集め, それを高速に連携させて目的を果たし, その後は利用者の視界から消滅するソリューションまたはインフラであると定義されている [2]. 本研究では, グリッドコンピュータは, ネットワークで結ばれたコンピュータの各端末へ, 小規模に分割された大規模問題の一部を割り当てて処理するシステムを指すことにする.

グリッドコンピュータにおいて, 分割された問題を接続された端末に割り当てる仕組を「スケジューリング」と呼ぶ. スケジューリングは, グリッドコンピュータの高速化において, 重要な機能である. グリッドコンピュータに接続される端末は性能が不均一であることが多く, 正確なスケジューリングを考えると, 計算時間が多くなる. そこで近似解であるが, 高速に解を得ることができる「近似アルゴリズム」を用いることにした.

1.2 研究内容

本研究の目的は, 近似アルゴリズムを用いて, グリッドコンピュータシステムの計算時間を解の保証をしつつ高速化することである.

「ビンパッキング」とは, サイズの異なる複数の品物と一定の大きさのビンが複数個用意されているとき, 品物全てをビンへ詰め込むことであり [1], ビンの数を最小にする, すなわち最も効率のよくビンへ品物を詰め込む問題を「ビンパッキング問題」という. 本研究ではスケジューリング問題は, ビンパッキング問題における品物をグリッドコンピュータで解かせたい問題とみなし, ビンをグリッドコンピュータの端末とみなすことで, スケジューリングがビンパッキング問題を解くことに相当するとした. 本研究では「ビンパッキングを解くためのアルゴリズム」[1] を用いてアルゴリズムを作成した.

2. 本研究におけるスケジューリングアルゴリズム

本研究では,「ビンパッキングを解くためのアルゴリズム (BP)」を用いてスケジューリングアルゴリズムを作成した. 以下にBPの流れを示す.

1. 品物をサイズの小さい順に並び替える.
2. 規格化された品物のサイズに対する閾値 $\epsilon(0 \leq \epsilon \leq 0.5)$ を設定する.
3. サイズが ϵ 未満の品物を一旦除く
4. 品物を $\lfloor n\epsilon^2 \rfloor$ 個の連続するグループになるように分割する
5. 各グループに分割されたすべての品物のサイズをグループ内で最も大きい品物のサイズに変更する
6. サイズを変更した品物によって最適なビンパッキングの解を得る
7. 6で得られた解に ϵ 未満の品物を割り当てていく

グリッドにおけるスケジューリングでは, 品物をグリッドコンピュータに与える問題, ビンをコンピュータと捉える. またコンピュータは, そのコンピュータのCPUClockSpeedに応じた本数のビンを持つ.

3 実験環境

グリッドコンピュータシステムは, GlobusToolkitを用いて構築した. また各コンピュータの資源状態を取得するためGangliaを用いた.

本研究では,5つのコンピュータを同一ネットワーク上に接続する. 5つのコンピュータはそれぞれ,Jan-

Implementation and evaluation of scheduling for a grid computerwith approximation algorithm

† Yusaku MITSUISHI,Numazu National College of Technology Advanced Engineering Course

‡ Yasuhito SUZUKI,Numazu National College of Technology Department of Control and Computer Engineering

Copyright ©2012 Information Processing Society of Japan.
All Rights Reserved.

uary,February,March,April,May というホストネームを持つ. 各コンピュータの主な性能は Table 1 で示す.

Table 1 コンピュータの性能

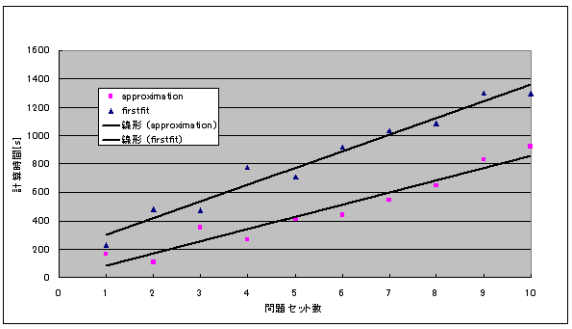
Computer	RAM[MB]	CPU Clock[MHz]
January	985	1800
February	985	1600
March	240	900
April	621	900
May	240	900

スケジューリングは February が担当し,February が, 自分自身を含めた 5 つの端末に実行命令を出す.

2 分割問題とは, 有限個の整数を 2 つのグループに分割したとき, グループ内の整数の総和が,2 つのグループで, ともに一致する分割の組み合わせが存在するか探す問題である. この問題は分割することが可能であるため, グリッドコンピュータで実行することができる. よって本研究では, 最適な組み合わせを探索する「2 分割問題」を解くこととした. 本実験では,1 からある整数 n までの連続する整数を入力とした「2 分割問題」を扱う. 2 分割問題における「問題のサイズ」は, 整数 n を指すこととする. サイズ n の 2 分割組合せの総数は, 組合せの公式より, $\sum_{i=1}^{\lceil \frac{n}{2}-1 \rceil} {}_n\mathrm{C}_i + \frac{{}_n\mathrm{C}_{\lceil \frac{n}{2} \rceil}}{1.5+(-0.5)\times(-1)^{n+1}}$ となる. よってサイズが増加すると, 組み合わせ数は指数的に増加することになる.

4. 実験結果

近似アルゴリズムによるスケジューリングアルゴリズムを用いた, グリッドコンピュータのシステム全体の時間を計測する. 比較のため, スケジューリングをせずに処理が終わったコンピュータへ順に問題を割り当てる「FirstFit アルゴリズム」も作成した.

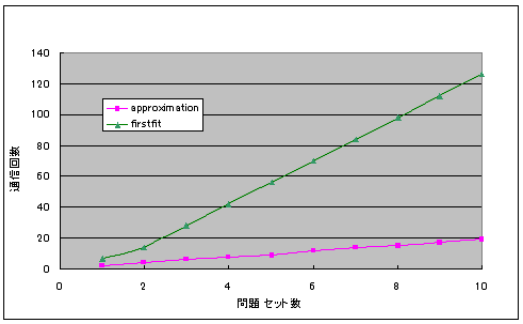


Graph 1 近似アルゴリズムと FirstFit の計算時間

Graph 1 より,FirstFit アルゴリズムより, 近似アルゴリズムのほうが, どの問題セット数においても計算時間が少なくなっていることがわかる. また, その差は本研究で提案する近似アルゴリズムを基準として, 平均で 2 倍の差がある.

5. 考察

Graph 2 は問題セット数あたりの, 問題を送受信する通信回数の変化を示したものである.FirstFit アルゴリズムは, グラフより問題セット数の増加とともに比例して, 通信回数が増えていることがわかる.



Graph 2 近似アルゴリズムと FirstFit の通信回数

グリッドコンピュータは, 問題を送信し解を受信する毎にネットワークを通した通信, 認証などのセキュリティ管理を行う. あらかじめスケジューリングが決まっているビンパッキングでは, ビン毎に問題をまとめて送信するため, FirstFit より通信にかかる遅延が少なくすんでいると考えられる.

6. 研究成果

本研究では CPU Clock をコンピュータ性能として見立て, この性能差に応じて, 多く問題を解くよう近似アルゴリズムを用いてスケジューリングを行った. 結果, 本グリッドシステムは FirstFit を基にしたスケジューリングより, 早く問題を解くことができた. この結果の要因はグリッドコンピュータが問題を送受信する際に必要な時間を抑えることができた点であると考えられる.

参考文献

[1] V.V. ヴァジラーニ. 浅野孝夫 (訳),『近似アルゴリズム』, シュプリンガー・ジャパン,2002.

[2] 日本アイ・ビー・エムシステムズ・エンジニア株式会社,『グリッド・コンピュータとは何か』, ソフトバンクパブリッシング株式会社,2004.

Copyright ©2012 Information Processing Society of Japan.
All Rights Reserved.