マイグレーションを用いたPCグリッドシステムのジョブ分割法
New job scheduling for PC grid system with migration function

森川浩明†, 榎原博之‡, 大西克実†, 中野秀男†, 山川栄樹*, 荒川雅裕*, 檀寛成*
Hiroaki Morikawa†, Hiroyuki Ebara‡, Katsumi Onishi†, Hideo Nakano†, Hideki Yamakawa*, Masahiro Arakawa* and Hiroshige Dan*

概要：ここ数年でスーパコンピュータと家庭用計算機の能力差は縮まりつつあり、家庭用計算機複数台でスーパコンピュータ1台分程度の能力になってきている。これら家庭用計算機は、ユーザが求める計算能力よりもオーバスペックであるため計算資源の有効活用が求められている。また、仮想化技術がOSの機能として搭載されたことにより、計算資源を仮想的に分割・統合し、サーバなどの計算資源を有効活用する仮想計算機技術が利用されている。本研究では、仮想計算機がハードウエアに依存しない特徴から並列計算においてユーザが計算機を利用する場合に別の計算機に仮想計算機ごと計算内容を移行するマイグレーション機能を実装したグリッドシステムにおいて、マイグレーションにかかる時間を考慮したジョブ分割法の提案をおこなう。

キーワード：並列計算、PCグリッド、マイグレーション、ジョブ分割
Key Words:parallelization, PC grid system, migration, job scheduling

1 はじめに

家庭用計算機が高性能であるにもかかわらずその性能をフルに発揮していない点を考慮し、これらの計算資源の有効活用を目的として、大規模なマシンパワーを発揮するシステムを構築するPCグリッドが注目を浴びている。一般的なPCグリッドでは、計算機があまり利用されていない遊休時間に計算ジョブを実行するシステムであり、ジョブ間の通信が必要ない特異な問題にしか適用できないため計算できる問題が限定されている。また、キャンパスグリッドという大学などの計算機室の計算機を夜間に利用するシステムも存在するが、夜間に限定されるため長時間の実行に向かない。

一方、サーバなどの計算機では繁忙期であってもCPU使用率やメモリ利用率が100%に達することは無いことから、計算機のCPUやメモリなど利用する計算資源を指定し、仮想的に指定した範囲の計算資源を利用する計算機を複数構築できる仮想計算機技術が必要になってきている。仮想計算機には、1台の計算機に複数の仮想計算機を設置できる特徴のほかに計算機のハードウエアに依存しない特徴がある。これを利用し、負荷上昇時などに仮想計算機を別計算機に移動させるマイグレーション機能を実装できる。本研究では、キャンパスグリッドなどのおおまかな使用状況がわかりやすい環境を想定し、マイグレーションの頻度に基づくジョブ分割の有効性を示す。通常マイグレーションにかかる時間は並列計算にかかる時間よりも小さいものであるが、効率的なジョブ投入のためには、計算機室内の利用状況からPCグリッドの投入元のスケジューラが各計算機への投入ジョブの演算時間を推定し、決定する。そのため、本研究ではマイグレーションを考慮して投入されるジョブのサイズを変更することで各計算機への投入ジョブの演算時間を変更し、各計算機の予想されるログオフ期間（休止時間）内に収め、効率的なジョブ投入を行うように考慮したメモリ分割法を提案する。

以下、2章では本研究と関連する研究を紹介し、3章では対象となる仮想計算機技術、4章では一般的なPCグリッドシステムについて述べたあと、利用したPCグリッドシステムについて説明を加え、5章ではマイグレーション機能のパフォーマンスの評価をおこない、6章では提案手法の実験・評価結果から得た

† 大阪市立大学創造都市研究科
‡ 関西大学システム理工学部
* 関西大学環境都市工学部

考察について議論する。

2 関連研究

仮想計算機のマイグレーション機能をグリッドシステムに応用した研究として、立薗らの研究 [16] がある。立薗らの研究では、仮想計算機のライブマイグレーション機能を利用し、投入ジョブ実行中の計算用 PC のジョブキュー内に複数のジョブが存在するとき、他の遊休状態にある計算機に投入ジョブの一部をマイグレーションさせ負荷分散をおこなう。この研究では仮想計算機のサスペンド機能を用いたマイグレーション機能やキャンパスグリッドを想定しており、われわれの研究と共通する点が多い。しかし、グリッドシステムの開発に重きをおいているため、効率的な運用や計算機のユーザによる利用をあまり想定していない。本研究ではユーザの利用が多い昼間でも効率の低下を最小限にした運用ができるようジョブ投入を改良することを目的としている。また、本研究では各計算機のキューはジョブを 1 個しか受け入れないため、ジョブ投入の段階で投入ジョブの負荷分散をおこなう。

全社的に遊休計算機の有効活用を図った研究として、中国電力の曽山らの研究 [14] がある。曽山らがおこなった研究では、グリッドシステムの効率運用のため、週間の電源投下状況をジョブ投入スケジュールとし、実際の環境へジョブ投入をおこなった。この研究では、あらかじめ起動している計算機にジョブを投入しているため利用時間の予測が容易である。また、ヘテロ環境を想定しているため、投入ジョブを大きくとることで終了時間が延びることから最適なジョブ分割に関する考察がなされている。この方式のグリッドシステムでは、夜間のジョブ投入やジョブ分割が難しいジョブがあるなど負荷増大で通常業務が圧迫されるなどの問題や互いに独立なジョブしか扱えないため、通信が必要なジョブは計算できない問題がある。本研究では、電源が切れている状態の計算機に WakeOnLAN パケットを投げることでジョブ投入可能状態にすることに加え、消費するメモリ量でジョブ分割をおこない、投入ジョブの細分化をおこなっている。

他に、複数大学間でのグリッドシステム構築にかかわるプロジェクトとして [3] があり、マイグレーションにかかわる研究として [4]、[10] がある。仮想計算機を用いたグリッドシステム構築にかかわる研究として [7] がある。グリッドシステムのセキュリティにかかわる研究として [5]、[6] がある。

3 仮想計算機

3.1 仮想計算機

仮想計算機は、計算機上にメモリや CPU、通信回線などを仮想的に構築し、単一の計算機 (ホスト OS) 上で仮想的に複数の計算機 (ゲスト OS) が動作しているかのように見せかけることのできる技術である。代表的な仮想化ソフトとしては、VMWare[17] や Xen[18]、Jail[8] などがある。

仮想計算機は、仮想計算機イメージ (VM イメージ) と仮想化層 (仮想化ソフト)、ハードウエアから構成されており (図 1)、一般的に仮想計算機は仮想化層を通して間接的にハードウエアを操作している。

仮想計算機では、仮想化層によってハードウエアから切り離されているために以下のような特徴を備えている。

- 複数の仮想計算機を起動できる
  仮想化層によるハードウエアの排他的な使用によって 1 台の計算機が複数台の計算機であるかのように見せかけられる。サーバなどでは、資源の有効活用のため導入されている。
- ハードウエアに依存しない
  ハードウエアの操作は仮想化層でおこなわれるため、仮想計算機にはハードウエアの影響が少ない。このため、異なるハードウエア環境に VM イメージを転送しても仮想化ソフトが同じであるならば動作可能である。
- ホスト OS からの独立性
  仮想計算機はホスト OS に関係なく動作可能であり、一部の仮想計算ソフトではホスト OS が存在しない環境であってもゲスト OS が起

動できる。

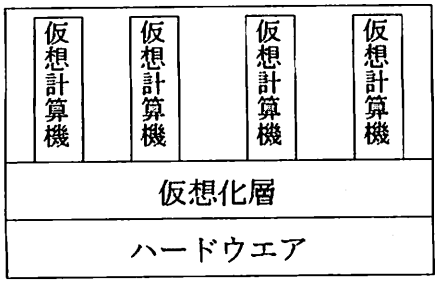

図 1　仮想計算機の構造

3.2　マイグレーション機能

代表的な仮想化ソフトに備わっているマイグレーション機能は、現在ある計算機上で実行中の計算内容や計算環境を別の計算機に移行させる機能である。マイグレーション機能は移行するデータなどによって以下のように分類できる。

1. チェックポイントマイグレーション
   一定期間、もしくは特定のアクションごとに現在実行中の状態（メモリ内容、レジスタ内容など）を保存し、障害発生時などに別計算機で保存した状態を展開する手法
2. プロセスマイグレーション
   メモリ内容などのデータを移行させるのではなくプロセス自体を別の計算機に移行させる手法
3. ライブマイグレーション
   計算機を停止させずメモリ内容などを少しずつ別計算機に移行させることで、外見上はある計算機が計算を停止したと同時にその計算機で実行していた内容を別計算機が引き継いで実行しているように見せかけることができる手法

現在、マイグレーション機能は主に仮想計算機によって実現されており、本研究でもマイグレーション機能の実装に VMware Server を利用している。また、本システムではユーザへの配慮と障害対策のためにチェックポイントマイグレーションを実装する。これは、チェックポイントマイグレーションではライブマイグレーションにおける計算内容の移行のようにマイグレーション情報の把握からメモリ内容を送信する際にライブマイグレーション特有の処理やメモリ内容を分割して送信するなどの余分な処理が発生しないため、障害対策として優れた機能を示すと判断したためである。

4　PC グリッドシステム

4.1　PC グリッドシステム

グリッド協議会 [9] の定義では、「グリッドは、広域ネットワーク上の計算、データ、実験装置、センサー、人間などの資源を仮想化・統合し、必要に応じて仮想計算機（Virtual Computer）や仮想組織（Virtual Organization）を動的に形成するためのインフラ」とされている。グリッドシステムでは、ネットワークにつなげばシステムに参加するすべての計算機がその計算能力の恩恵に授かることができることを目的としている。このため、大学内でのグリッドシステム（キャンパスグリッド）や家庭内でのグリッドシステム（PC グリッド）も広義の意味でのグリッドシステムと定義できる。グリッドシステムでは、主に LAN 内で一定の計算能力を発揮するクラスタシステムと異なり、ネットワーク上に存在する計算機資源を確保する。

本研究では、キャンパス内の遊休 PC を利用したコンピューティンググリッドをおこなっている。特に、演習室などの計算機をオープン利用時に利用することを想定している。この環境では、提供される各計算機にユーザが存在するので、遊休状態にある計算機 (遊休 PC) を探索し、発見した遊休 PC にジョブを投入する。

4.2　PC グリッドが備えるべき機能

PC グリッドが備えるべき機能としてスケジューリング、ユーザビリティ・障害対策、セキュリティな

どがある。

本研究ではこれらの機能のうちスケジューリングとユーザビリティ・障害対策について述べる。

・スケジューリング機能

PC グリッドシステムの多くがジョブを順次実行する。そのため、計算資源を提供する計算機が計算途中でジョブを終了しても問題が発生しないよう同じジョブを複数の計算機に投入している。これは、大規模なシステムではすべての計算ノードの使用状況把握は困難なためである。しかし、実際にはドメインごとに大まかな使用状況を把握し、ジョブ投入スケジュールを作成することで効率的なジョブ投入をおこなう必要がある。このことから、グリッドシステムのスケジューリング機能は一定時間ごとに計算機の状態を把握し、ジョブ投入時予測される各計算機の遊休時間に収まるようにジョブ分割しなければならない。

また、計算機環境がヘテロ環境であるかどうかも考慮しなければならない。ヘテロ環境では低スペックの計算機に終了時間が影響を受ける。低スペックの計算機が、終了時間に影響を及ぼさないためにはジョブ分割数を多くすることが必要である。しかし、ジョブ分割数が多いと通信オーバーヘッドが高くなり、通信オーバーヘッドとジョブ分割数のトレードオフになる。

本研究では、ハードウエアの違いによるヘテロ環境を仮想計算機によりホモジーニアス環境にしている。また、ユーザ利用時間に合わせジョブ分割をおこなうことで、マイグレーションによる実行時間の増加を減少し、ジョブの総実行時間を抑えている。

・ユーザビリティ・障害対策

PC グリッドの各計算機にユーザが存在する環境では、ユーザが資源解放を要求したとき即座に状態保存と資源解放を実行しなければならない。また、障害発生時にも高速な状態保存と資源解放が必要になる。PC グリッドにおいて発生する障害は、計算機のハードウエアの破損、ユーザによる計算機のシャットダウンやトラヒックの急激な増加などの通信による障害がある。これらの障害のうち本論文では、100 秒程度の比較的時間に余裕のある障害やトラヒックの増加などによる通信障害を考慮している。この対策としてマイグレーション機能の実装がある。現在、マイグレーション機能の実装は仮想計算機を活用したものが主流であり、本研究でもユーザが計算機の使用を開始したというアクションや障害を検知し、仮想計算機のサスペンド機能を用いてチェックポイントマイグレーションを実装している。ただし、本研究で想定していない通信障害などに関しては、ジョブが実行されなかったと判断してジョブの再投入をおこなっている。

4.3 Systemwalker Cyber GRIP の概略

本研究では、グリッドミドルウエアである Systemwalker Cyber GRIP を利用している。Systemwalker Cyber GRIP はグリッド計算用の高性能並列演算環境である。このミドルウエアは、富士通研究所によって開発されたものであり、バックグラウンドで動作するデーモンとして起動している。Systemwalker Cyber GRIP は perl や shell ライクな専用スクリプトによって 1 つの処理を複数の処理に分割し、ネットワークにつながった複数の計算機に投入できる。Systemwalker Cyber GRIP では内部に以下の機能を持つ 2 種類のキューが存在する。

1. 仮想キュー
   計算用 PC を仮想的に 1 つの計算機に統合した際、基準となる計算機（マスタサーバ）に存在するキューで、投入ジョブの計算用 PC への分散を処理する。
2. 実行キュー
   各計算用 PC に存在するキューで、投入ジョブの実行をおこなう。

Systemwalker Cyber GRIP では並列に実行したいジョブを記述する場合、専用スクリプト言語であるため通信や暗号化などで制限を受ける。より効率的な並列計算には、実行に必要な入力ファイルや実行ファイルを用意しておき、Systemwalker Cyber GRIP のファイル転送機能によって計算用 PC に投入する必要がある。

4.4 システム概略

利用した PC グリッドシステムは関西大学と富士通研究所が共同開発したシステムでマスタサーバ 1 台と計算用 PC7 台を用いて、サーバ・クライアントからなるスター型のネットワーク構造を成している (図 2)。また、システムで利用している計算用 PC にはそれぞれ使用者が存在し、研究や業務に使用している。本システムではユーザが存在する環境において効率的なジョブ投入をおこなうため、VM 管理テーブルを利用したジョブ管理機能と仮想計算機によるマイグレーション機能を実装している。

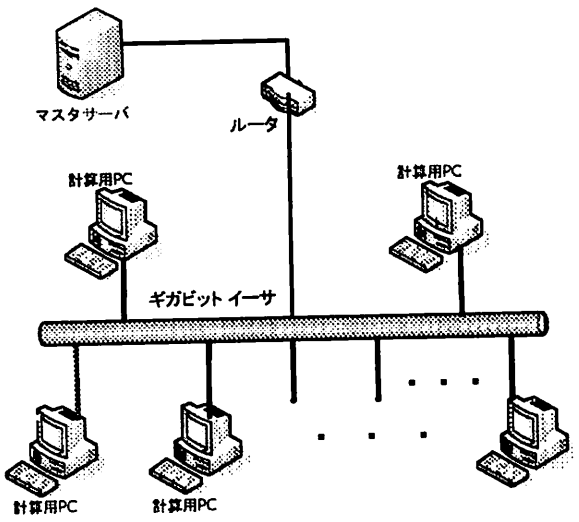

図 2　利用システムの全体図

グリッドシステムでは高速にマイグレーション機能を利用するため、マスタサーバが保持している VM イメージを各計算機がネットワークドライブとして直接操作する。

利用した PC グリッドシステムは図 3 の構成になっており、以下の機能を持つ。

- マスタサーバ
  - 計算用 PC 管理機能
    計算用 PC の利用状況管理機能から利用状況変更通知を受け、計算機の利用状況を更新する。必要に応じてチェックポイントとリスタート命令を計算用 PC のチェックポイントリスタート管理機能に送る。
  - VM イメージ共有機能
    VM イメージ共有では計算用 PC で利用する VM イメージを保持する。VM 管理機能では、VM イメージを利用している計算用 PC のホスト名と投入ジョブ ID を関連付け計算資源管理をおこなう。
- 計算用 PC
  - 子ジョブ生成管理機能
    マスタサーバからジョブ実行要求を受け、仮想計算機の起動とチェックポイントをおこなう。
  - 利用状況管理機能
    ユーザのログオン・ログオフを監視し、その結果をマスタサーバの計算用 PC 管理機能に通知する。
  - チェックポイントリスタート管理機能
    マスタサーバの計算用 PC 管理機能からの要求に応じてジョブの起動・停止をおこなう。
- ゲスト OS
  - 子ジョブ実行機能
    計算用 PC の子ジョブ生成管理機能から子ジョブ実行要求を受け取ると子ジョブを実行する。実行完了後、子ジョブ生成管理機能に結果を通知する。

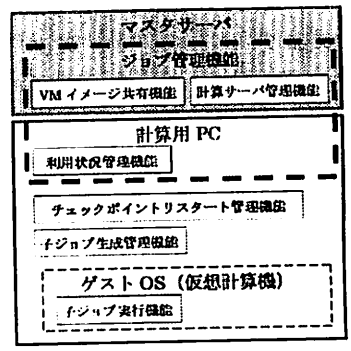

図 3　構築システムの構成

4.5 ジョブ実行

本システムのジョブ実行はマスタサーバによって管理され、計算用 PC は仮想計算機上でのジョブの実行状態を管理するための VM テーブルに排他的アクセスをおこなうことでマスタサーバにジョブの実行状態を通知する。

本システムへのジョブ実行は以下のプロセスを取る。

1. マスタサーバ上でシステム利用者がジョブを投入する。
2. 利用する計算用 PC に wakeonlan パケットを投げ、計算用 PC を起動する。
3. 利用する計算機に必要なファイル（入力ファイル、実行ファイル）とパラメータを送信する。
4. ジョブ ID と VM イメージを結びつけた情報を VM テーブルに書き込むことで、各計算用 PC で利用する VM イメージをロックし、仮想計算機を起動する。
5. ゲスト OS でジョブを実行し、ジョブの出力ファイルなどをマスタサーバに返す。

このプロセス中では、VM イメージ共有機能によって現在 VM イメージを利用している計算用 PC と実行中のジョブ ID を結び付けているため、マスタサーバでジョブの状態を逐次知ることができる。ただし、初回の仮想計算機起動時とマイグレーション後に VM イメージのロックおよび仮想計算機の起動時間である 90 秒程度が必要になる。

4.6 マイグレーション機能

本システムのマイグレーション機能は、仮想計算機のサスペンド機能を用いて行われ、以下の条件のときに発生する。

- ログオンする
  ログオン中はユーザの計算用 PC 利用状況が監視され、ユーザがマウス・キーボードを使用しない状態が 30 分経過すると計算用 PC を再び利用可能にする。ただし、この時間は設定により変更可能である。
- 障害などによるシャットダウン
  仮想計算機が正常にシャットダウンできる障害であれば、仮想計算ソフトの設定によって仮想計算機をサスペンドさせる。ただし、仮想計算機のサスペンドには 100 秒程度の時間が必要になる。また、ネットワークの寸断や急なハードウエアの破損には対応できないため、このような状況では計算を最初からやり直す必要がある。

上記の条件が発生すると、集中的に計算機を管理するマスタサーバにジョブを実行するゲスト OS のホスト OS が使用状況の変化をマスタサーバへ送信すると同時にバックグラウンドでゲスト OS をサスペンドさせる。マイグレーションしたジョブは SystemwalkerCyberGRIP のジョブキューの最後に登録されるため、VM 管理テーブルが満たされるかマイグレーションしていないジョブが全て終了した後ジョブの再投入がおこなわれる。

マイグレーション機能の処理の流れを図 4 に示す。

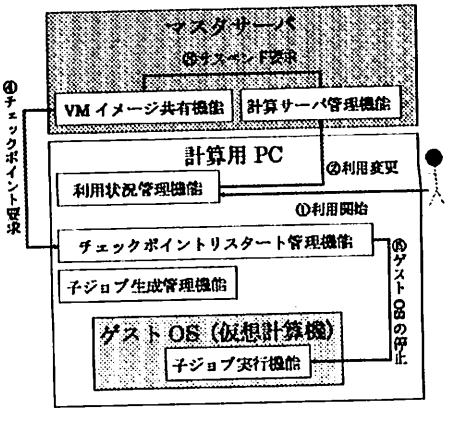

図 4　マイグレーション機能の処理の流れ

5 マイグレーション機能の性能評価

5.1 実験システム

本研究では以下のスペックの計算用 PC(ホスト OS)7 台と各計算用 PC に構築するゲスト OS を用いてシステムの評価と実験をおこなう。

表 1 マスタサーバ

OS	RedHatEnterPriseLinux4
CPU	Intel Xeon 1.86GHz
MEMORY	2GB
分散環境	SystemwalkerCyber GRIP

表 2 ホスト OS

OS	Windows XP Professional SP2
CPU	Intel Core2 Duo 2.4GHz
Memory	2GB
仮想計算機	VMware Server1.0.6
分散環境	SystemwalkerCyber GRIP

表 3 ゲスト OS

OS	CentOS 4.4
CPU	1Unit
Memory	1GB
ネットワーク設定	NAT

本システムのゲスト OS では、メモリと CPU をホスト OS の半分しか使用していないが、これはユーザがログオンしてきた際に、バックグラウンドで実行するマイグレーション処理によってユーザのプログラム実行に影響を与えないためである。

5.2 マイグレーション機能の性能評価

本システムにおけるマイグレーションにかかる時間は、マイグレーションの発生した計算用 PC でのゲスト OS のサスペンド時間と、マスタサーバでの他の遊休 PC 探索時間、加えて投入先の計算用 PC でのゲスト OS の起動時間と消費メモリ領域の展開時間の 4 つからなる処理に分類できる。ただし、マイグレーションが発生した計算用 PC でのサスペンド時間とマスタサーバでの遊休 PC の探索時間、投入先計算用 PC でのゲスト OS での起動時間は一定であり、150 秒程度である。

投入先計算用 PC でのメモリ領域の展開時間は利用しているメモリ領域によって変化するため、指定したメモリ領域を確保する malloc プログラムを用いて、100MB から 700MB までのプロセスが投入先の計算用 PC でマイグレーションに要した時間を実験により求め、実際の投入ジョブにおける消費するメモリ領域からのマイグレーション時間を得る (表 4)。

表 4 メモリサイズごとのマイグレーションにかかる時間

メモリ量	100MB	300MB	500MB	700MB
時間 (sec)	212	216	265	264

表 4 より、消費するメモリ領域が 300MB から 500MB に変化するときに急峻することがわかる。これはマイグレーションで別計算用 PC に移行する際、512MB 単位でメモリ領域の情報を移行しているためである。このため、ジョブ分割を 512MB 単位でおこなえば効率的なジョブ投入となることがわかる。

6 メモリ分割法

6.1 メモリ分割法

マイグレーション機能にかかる時間は、フマイグレーションの発生した計算用 PC でのゲスト OS のサスペンド時間と、マスタサーバでの他の遊休 PC 探索時間、加えて投入先の計算用 PC でのゲスト OS の

起動時間と消費メモリ領域の展開時間の4つによって決定される。サスペンド時間と再開時間については、利用するスワップ領域で決定されるため、マイグレーション時間は表4で示されるとおり利用するメモリ領域が多いほど増加する。

ただし、1つのジョブが利用するメモリ領域を削減する目的でジョブ分割数を増やした場合、ジョブの投入回数が多いほどジョブ投入や最終処理などに遅延が要求されるため、実行時間が延びる場合がある。

マイグレーションの発生頻度の低い環境では、メモリ領域によるマイグレーション時間の増加を考慮する必要はないが、ジョブの分割数による影響が大きい。マイグレーションの発生頻度が高い環境では、マイグレーション時間の増加によってジョブ実行に悪影響がある。そのため、投入ジョブで利用するスワップ領域を含むメモリ領域のサイズをマイグレーション発生頻度から自動変更し、ジョブ実行時間を最適化する。

本研究では、ジョブが利用するメモリ領域を変化させることで、マイグレーション頻度に対するジョブ実行時間の最適化でき、効率的なジョブ投入が可能であると考え、投入プログラムの作成、評価をおこなう。

6.2 メモリ分割法のための準備実験の手法

本実験では2つの8192次正方行列の乗算を4台の計算用PCを用いて並列計算する。行列演算は$O(n^3)$の問題であり、対象となる行列をx分割すると各計算にかかるオーダは$O(x^3) \times O(\left(\frac{n}{x}\right)^3)+O(n^2)$となり、加えて行列演算の最終処理に$O(n^2)$の演算がかかる。並列化では、対象となる8192次正方行列をn次正方行列の4096、2048次正方行列と1024次正方行列の3種類に分割し、メモリ分割法の指標となる各マイグレーション発生頻度0%から40%での実行時間を比較する。

これらのn次正方行列で消費するメモリ領域は表5のようになる。

本実験では、各分割数による評価を目的とし、実行時間の最小のものをメモリ分割法に用いる。

表5 n次正方行列での消費メモリ量

行列サイズ	1024	2048	4096	8192
消費メモリ量	50MB	180MB	680MB	2670MB

マイグレーションの発生には、一定時間ごとに各計算用PCにログインするかの判定をランダムにおこない、仮想計算機のサスペンドが終了するとログオフするプログラムを利用する。

6.3 実験結果

PCグリッドシステムを利用し、メモリ分割とマイグレーションを考慮しないジョブ分割でマイグレーション頻度とユーザのPC離席率を変更させるシミュレーションをおこない、実験結果を図5に示す。

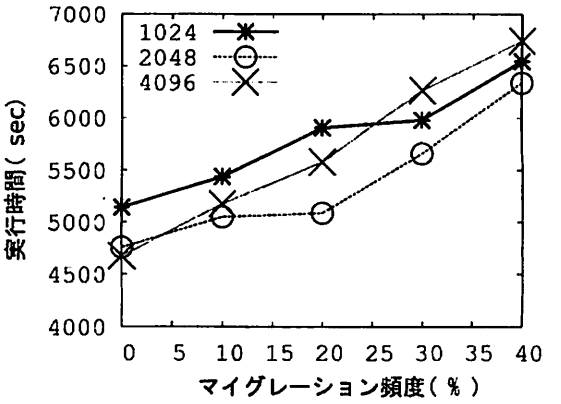

図5 マイグレーション発生時の8192次正方行列並列演算の実行時間の比較

図5より、マイグレーション頻度が増加するに従い、実行時間も増加していることがわかる。特に、分割数が最小となる4096次正方行列では実行時間が2000秒も増えるなど大きな影響がある。一方、1024次正方行列は実行時間の変化が他の分割数と比べてなだらかであったが、マイグレーションが発生しない環境で400秒程度他の分割数よりも実行時間が多く、マイグレーション頻度の高い環境で無いと有効に働かないと推察できる。また、2048次正方行列と

4096 次正方行列ではマイグレーションがない場合に実行時間の差が 90 秒程度あったのに対して、マイグレーション回数が 2 回以上となるマイグレーション頻度 10% 以上の場合には 2048 次正方行列が 4096 次正方行列の実行時間を下回っていることがわかる。このことから、512MB 以上のメモリ領域を用いる 4096 次正方行列にかかるマイグレーション時間が 2048 次正方行列の最終処理の時間を上回ったことがわかる。

この結果から、マイグレーション頻度が 0 から 10% までは 4096 次正方行列に分割し、20 から 40% までは 2048 次正方行列に分割することが最も効率的なジョブ投入となる。この方法をメモリ分割法として、次節で実験・評価する。

6.4 メモリ分割法の実験・評価

利用した 4 台の計算用 PC に 8192 次正方行列を 4 分割 (2048 × 8192 行列) と 8192 正方行列を乗算するジョブを投入し、メモリ分割法の結果と比較する。

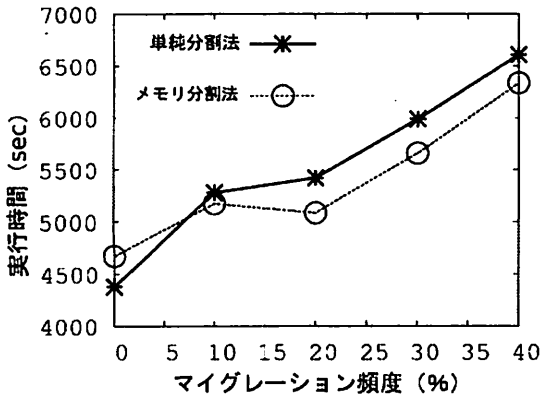

図 6 マイグレーションを想定したプログラムと単純分割の実行時間の比較

結果より、マイグレーションが発生しない環境では 300 秒程度の遅延があるが、マイグレーション頻度が 20% 付近から逆転し、単純な 4 分割よりもよい結果を返す。この原因として、低いマイグレーション頻度では、分割したジョブの統合処理（終了処理）やファイル転送にかかるオーバヘッドにより実行時間が単純な分割よりも上回るが、マイグレーション頻度が高い環境では、マイグレーションにかかるオーバヘッドがメモリ分割法の実行時間を上回ったと考えられる。

7 おわりに

本研究では、遊休計算機を有効活用するための手段である PC グリッドシステムのマイグレーション機能とマイグレーション機能の実現手段である仮想計算機に注目して実験をおこなった。

ユーザの利用を考慮したメモリ分割法により、マイグレーション頻度の高い状況で効率のよい投入ができることが示せた。また、ジョブ分割によるオーバヘッドがジョブ実行に与える影響をみることができた。

しかし、今回のシステムは計算用 PC が 10 台未満の小規模なものだったので、より大規模で広域ネットワークに及んだ場合の影響を今後検討する必要がある。

謝辞

本研究をおこなうにあたり，多くの人々にご協力をいただきました。研究のシステム構築に携わってくださった富士通研究所の皆様、研究の場を提供してくださったソシオネットワーク戦略研究センターの皆様に感謝の意を表します。

また、本研究は、平成 20 年度関西大学重点領域研究助成金において、研究課題「休止中のコンピュータを有効利用するグリッドシステムの構築とその応用」として研究費を受け、その成果を公表するものである。

参考文献

[1] 合田憲人・関口智嗣 編著: グリッド技術入門, コロナ社, (2008).

[2] 秋岡明香, 村岡洋一: グリッド環境での CPU 負

荷予測に基づくネットワーク負荷中期予測電子情報通信学会論文誌, VolJ87-D-I(2004).

[3] F. Berman, H. Casanova, A Chien, K. Cooper, H. Dail, A. Dasgupta, W. Deng, J. Dongarra, L. Johnsson, K. Kennedy, C. Koelbel, B. Liu, X. Liu, A. Mandal, G. Marin, M. Mazina, J. Mellor-Crummey, C. Mendes, A. Olugbile, M. Patel, D. Reed, Z. Shi, O. Sievert, H. Xia and A. YarKhan: New Grid Scheduling and Rescheduling Methods in the GrADS Project, International Journal of Parallel Programming, Vol33,(2005).

[4] Eun-Kyu Byun and Jin-Soo Kim: DynaGrid: A dynamic service deployment and resource migration framework for WSRF-compliant applications, Parallel Computing, Vol33(2007).

[5] Haibo Chen, Jieyun Chen, Wenbo Mao and Fei Yan: Daonity – Grid security from two levels of virtualization, Information Security Technical Report, Volume 12, Issue 3,(2007).

[6] Yan Fei, Zhang Huanguo, Sun Qi, Shen Zhidong, Zhang Liqiang and Qiang Weizhong: An improved grid security infrastructure by trusted computing, Wuhan University Journal of Natural Sciences, Volume 11, Number 6,(2006).

[7] Renato J. Figueiredo, Peter A. Dinda and Jose A. B. Fortes: A Case For Grid Computing On Virtual Machines, Distributed Computing Systems, Vol23 (2003).

[8] FreeBSD jail: http://www.onlamp.com/pub/a/bsd/2003/09/04/jails.html.

[9] グリッド協議会: http://www.jpgrid.org/index.html.

[10] F. Heine, M. Hovestadt, O. Kao and A. Keller: SLA-aware job migration in grid environments, Advances in Parallel Computing, Vol14, (2005).

[11] ITpro 編: すべてわかる仮想化大全 2009, 日経 BP, (2008).

[12] Bruno Richard, Nicolas Maillard, César A.F. De Rose and Reynaldo Novaes: The I-Cluster Cloud: distributed management of idle resources for intense computing Parallel Computing, Vol31(2005)

[13] SETI@home: http://setiathome.berkeley.edu/.

[14] 曽山豊: 企業におけるグリッド・コンピューティングの活用とその成果, グリッド協議会セッション,Grid World 2006(2006).

[15] E.G. Talbi, Z. Hafidi, J-M. Geib: A parallel adaptive tabu search approach, (1998)

[16] 立薗真樹, 中田秀基, 松岡聡: 仮想計算機を用いたグリッド上での MPI 実行環境, SACSIS 2006, (2005).

[17] VMware: http://www.vmware.com/.

[18] Xen: http://www.xen.org/.