情報爆発時代におけるモデルベース資源選択による高速な仮想クラスタ構築

山崎翔平[†1] 丸山直也[†1] 松岡聡[†1,†2]

1. はじめに

情報爆発時代において，大規模な計算処理能力と効率的な分散計算資源の共有を実現する仮想クラスタが注目されている[1)-3)]。仮想クラスタとは，複数の物理クラスタから構成されているが，ユーザからは論理的に一つのクラスタに見えるように仮想化された計算クラスタである．複数の物理クラスタを統合することにより，大規模な計算処理能力を達成することができる．また，仮想マシン (VM) を用いているために，既存の環境とは独立にユーザ毎にカスタマイズされた計算環境を提供することができることから，グリッド環境上の計算資源を効率良く共有することが可能である．

大規模グリッド環境上で仮想クラスタを用いるためには，スケーラブルかつ高速な構築が必要である．また大規模環境ではハードウェア性能が各ノードで不均質となりうるため，ノード選択によっては性能が極端に低いノードによって全体の構築時間が大幅に増加しうる．これは、仮想クラスタ構築時間では構築に最も時間のかかる計算ノードに律速されるからである．具体的には，予備評価実験では同一クラスタ内でさえ100秒以上の差が生じることが観測されている．

仮想クラスタの実装におけるいくつかの論点は，Krsul ら[3)]，Foster ら[2)]，西村ら[4)] によって議論されているが，これらはいずれも均質な環境を仮定しており、不均質環境上での計算資源選択について考慮していない．

2. モデルベース資源選択による高速な仮想クラスタ構築

我々は，仮想クラスタ構築時間を予測するモデルに基づいた計算ノード選択による高速な仮想クラスタ構築を提案する．本手法では，各計算ノードにおいてユーザにカスタマイズされた VM を配備するプロセス (VM セットアッププロセス) を5つの論理的なステップに分割し，ステップごとに実行時間予測モデルを構築した．これらのモデルでは各計算ノードの性能 (CPU 周波数，ディスク読み書き速度，インストールパッケージ容量など) をパラメータとしており，各パラメータの線形結合で各ステップの実行時間を表現する．モデルの決定係数は，仮想クラスタを実際に構築し，その性能データを基に重回帰分析を行う．また，性能データを取得する際には，既存の仮想クラスタインストーラ VPC[4)] を用いた．

VM セットアッププロセスは以下の5つの論理的なステップに分割される．

(1) **Package Download**
(2) **Package Transfer**
(3) **Package Installation**
(4) **Configuration**
(5) **VM Boot**

Package Download は，各サイトの代表ノードが追加でインストールするパッケージをダウンロードするステップである．各サイトにはあらかじめ VM イメージが配備されているが，ユーザが要求するパッケージのうち不足分を追加でインストールする必要がある．*Package Transfer* は，ダウンロードしたパッケージを，代表ノードからターゲットとなる各計算ノードに転送するステップである．*Package Installation* は，各計算ノードにおいて，代表ノードから転送されたパッケージをインストールするステップである．*Configuration* は，IP アドレスなど各 VM の情報や各ソフトウェアの設定ステップであり，最後の *VM Boot* は，カスタマイズ処理を完了した VM を立ち上げるステップである．

各ステップにおいて仮定するモデル式は図1の通りである．*PkgSize* は，追加でインストールするパッケージサイズ (MB) を表しており，*TransferOrder* は，代表ノードから各計算ノードへとパッケージが転送される際の転送順序を示す．*CPUFreq* は CPU 周波数 (GHz) を表し，*DiskRead* はディスク読み込み速度 (MB/s)，*DiskWrite* はディスク書き込み速度 (MB/s) を表す．仮想クラスタ構築実験のデータを基に，これらの係数を決定した．その際，モデルの精度を表す統計指標である自由度修正済み決定係数では全体として高い数値を得ることができ，モデルの精度は高いと言える．

3. 評価実験

生成したモデルを用いた資源選択方の有効性を示すために，既存の仮想クラスタインストーラ VPC にモデ

†1 東京工業大学
†2 国立情報学研究所

Step1	Download	$\alpha_1(PkgSize)+\varepsilon_1$
Step2	Transfer	$\alpha_2(PkgSize)+\beta_2(TransferOrder)+\varepsilon_2$
Step3	Installation	$\alpha_3(PkgSize)+\beta_3(CPUfreq)^{-1}+\gamma_3(DiskWrite)^{-1}+\varepsilon_3$
Step4	Configuration	$\alpha_4(PkgSize)+\beta_4(CPUfreq)^{-1}+\gamma_4(DiskWrite)^{-1}+\varepsilon_4$
Step5	Booting VM	$\alpha_5(PkgSize)+\beta_5(CPUfreq)^{-1}+\gamma_5(DiskRead)^{-1}+\varepsilon_5$

図 1　各ステップにおいて仮定するモデル式

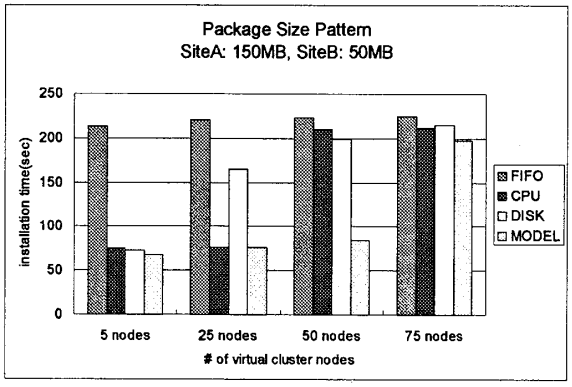

図 2　ダウンロードパッケージサイズがサイト毎に異なる場合の 4 つの選択法の比較

ルベースの資源選択機能を拡張し，評価実験を行った．拡張版 VPC では，計算ノード選択方法の比較のために，提案するモデルを活用する方法を含む，1)FIFO, 2)CPU, 3)DISK, 4)MODEL の 4 つの選択ポリシーを実装した．もっとも単純な手法である FIFO では，登録されたリスト順に計算ノードを選択する．CPU ポリシーや DISK ポリシーでは，各計算ノード上の CPU 周波数やディスク性能を比較し，性能の良い順に選択する．MODEL ポリシーでは，ユーザ要求に対して，各計算ノードにおけるモデル値を計算し，性能の良い順に選択する．

実験では，東工大，松岡研究室の PrestoIII クラスタを 2 つに分割し，擬似的に 2 サイトを用意した (SiteA, SiteB と呼ぶ)．インストールパッケージ容量，仮想クラスタ構成ノード数，実装した 4 つの選択ポリシーを変化させながら，仮想クラスタ構築の実験を行い、その構築時間を計測した．各サイトには 50 台ずつ，計 100 台の計算ノードを用いた．

4. まとめと今後の課題

実験の結果，MODEL ポリシーは，各サイトでインストールパッケージ容量が異なる場合に特に有効であることが分かった．図 2 は，SiteA で 150MB，SiteB では 50MB のパッケージをインストールする必要がある場合の結果である．MODEL ポリシーは、FIFO ポリシーに比べて最大 68 %，CPU ポリシーに比べて最大 60 %，DISK ポリシーに比べて最大 58 %の構築時間短縮を実現できることが分かった．MODEL ポリシーは，各サイトにおけるインストールパッケージ容量を考慮するため，SiteB 上のノードから仮想クラスタを構築する。一方、MODEL 以外の選択ポリシーでは 2 つのサイトにまたがる仮想クラスタを構築するために、構築時間が大きく増加してしまう。

ノード毎の性能が不均質な大規模グリッド環境においても安定して高速な構築を行うために，構築時間のモデル化に基づいた資源選択手法の提案した．本手法では，構築プロセスを 5 つの論理的なステップに分割し，ステップごとに実行時間予測モデルを構築した．

評価実験の結果，モデルに基づく選択法は，各サイトでインストールパッケージ容量が異なる場合に特に有効であり，最も単純な手法に比べて最大 68 %の構築時間短縮を実現することが分かった．

今後の展望としては，より大規模かつ分散した環境での本手法を評価することを考えている．また，仮想クラスタ構築時間だけでなく，ジョブ実行時間を最適化するために，投入されるジョブの特性を考慮した仮想クラスタの初期配置，マイグレーション機能を活用した実行中における仮想クラスタの再配置を検討している．

謝　辞

本研究の一部は科学研究費補助金特定領域研究 (18049028) の補助による．

参 考 文 献

1) Figueiredo, R., Dinda, P. and Fortes, J.: A case for grid computing on virtual machines, *Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS)*, pp.550–559 (2003).

2) Foster, I., Freeman, T., Keahy, K., Scheftner, D., Sotomayer, B. and Zhang, X.: Virtual Clusters for Grid Communities, *Proceedings of the 6th IEEE International Symposium on Cluster Computing and the Grid (CCGRID'06)*, Singapore, pp.513–520 (2006).

3) Krsul, I., Ganguly, A., Zhang, J., Fortes, J. A.B. and Figueiredo, R.J.: VMPlants: Providing and Managing Virtual Machine Execution Environments for Grid Computing, *Proceedings of the 2004 ACM/IEEE conference on Supercomputing (SC2004)*, Pittsburgh, PA, pp.7–18 (2004).

4) Nishimura, H., Maruyama, N. and Matsuoka, S.: Virtual Clusters on the Fly — Fast, Scalable, and Flexible Installation, *Proceedings of the 7th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid'07)*, pp.549–556 (2007).