

グリッドコンピューティングで解く企業利得評価モデル

坂本憲昭* 武智一貴* 小沢和浩* 新村隆英*

An economy model calculated using Grid Computing

* N. Sakamoto, K. Takechi, K. Ozawa and T. Niimura

Abstract: The profit valuation model of a company based on game theory is useful for strategy analysis of a company. In order to use this model, the value of eight parameters must be calculated. We use very long time for calculation because the search is an *NP* problem.

On the other hand, we own grid computing environment as the technique of realizing improvement in the speed of computation time. In this research, we calculate this problem by a grid. By the conventional research, we required four hours in order to calculate one data. By using a grid, we can obtain the result of the data of 20 simultaneously.

Key Words: Grid Computing, Game theory, Economic model

1. はじめに

ゲーム理論に基づく農薬市場における企業の利得評価モデルは、企業がその市場に参入して利得を得るか損失となるかなどの戦略分析に役立つ。また、すでにその業界で利得を得ている企業について、その組織形態の分析などに用いられる。ところで、このモデルが成立しているとみなすためには、目的関数の値をほぼゼロにする8個のパラメータ値を求める必要がある。本研究の目的は、このパラメータを探索することである。

パラメータ探索の難しさは、8個という多変数からわかるように *NP* 問題であることと、その他にモデルが関数というよりも計算アルゴリズムであり、非線形・離散・非凸関数である。また、事前に検討した結果では目的関数の最小値が多峰性関数と似たような様相にあると思われ、さらに、各パラメータに相互干渉がある。例えば、8個のパラメータを $b_1 \sim b_8$ と表現した場合、 $b_2 \sim b_8$ を固定して b_1 を目的関数が最小値になるように決定した後、同様に b_2 を探索した結果、その b_1, b_2 の組み合わせは最小値にならない。

このような経済モデルで解を求める類似研究では、Nelder-Mead Simplex¹⁾、シミュレーティッド・アニーリングなどが適用されているが、解の発見は偶然的であり、局所解と思われる。著者らが研究中の本稿に示すモデルにおいても、満足する解を得ることができていない。また、厳密解を得るために、パラメータの刻み幅を例えば0.01として、すべての組み合わせを調べるには計算時間が 10^{30} [sec]以上となり無限大となる。一方、単純に乱数による探索によれば、0.03%の確率で満足する解を得ている。しかしながら、1データを計算するのにほぼ1日を要しており、研究が遅々として進まない状況になっている。

そこで、計算時間の短縮を実現する手法として、パソコンのクラスタ化やグリッドコンピューティング環境（以下、グリッド）がある。クラスタ化は専用のソフトウェアやハードウェアが必要であり導入は容易ではないが、グリッドの場合、安価な汎用パソコンを用いて構築が可能であり、ミドルウェアも比較的低コストである。著者らはこの環境をパソコン実習室に導入し、教室を利用しない時間帯や夜間、休日を利用して、電力市場における価格予測計算の時間短縮²⁾、鉄鋼業の加熱炉におけるモデル計算時間の短縮と導入および保守コスト低減の検証³⁾を行ってきた。

そこで、本稿ではグリッドでモデル計算をすることにより、従来では1日に1データの計算結果しか得られなかったが、20データの結果を容易に得られることを示し、研究の迅速化と大学等におけるパソコン実習室の有効活用となるグリッド環境導入の利点を示す。

2. 農薬市場利得算出モデル

モデルの目的関数の主要な部分を式(1)~(5)に示す。この他に計算途中での並べ替えやデータの取捨選択などの処理をおこなう。

$$v = \sum_{j=1}^5 \{ \sum_{m=1}^{357} (\bar{E}_m - E_m) Z_{j,m} / 357 \} \quad (1)$$

\bar{E} は既知データ、 Z は外生変数で、ともにあらかじめ別途与えられる。

$$E_m = \sum_{k=1}^{100} n / 100 \quad (2)$$

$$n = \max \{ n \mid \sum_{n=1}^{16} I(\sum_{l=1}^{16} \pi) \geq n \} \quad (3)$$

$$I(\pi) = \begin{cases} 1 & \text{if } \sigma > 0 \\ 0 & \text{if } \sigma \leq 0 \end{cases} \quad (4)$$

$$\sigma = \sum_{j=1}^6 \beta_j X_{j,\ell,m} + \beta_7 \ln(n+1) + \beta_8 e_{1,m,k} + \sqrt{1 - \beta_8^2} e_{2,m,\ell,k} \quad (5)$$

n は整数、 e_1 は市場特有のかく乱変数（外乱）、 e_2 は企業特有のかく乱変数（外乱）とともに正規分布にしたがう確率変数で与える。 X は売上高、従業員数、専業・多角化などのタイプ、などの企業組織のデータであり、

* 法政大学

* Hosei University

(Received March 30, 2009)

あらかじめ別途与えられる。 $\beta_1 \sim \beta_8$ が探索するパラメータであり、 β_3 は負、 β_8 は $-1 \sim +1$ 、それ以外は $-50 \sim +50$ の範囲である。

$v = 0$ にする $\beta_1 \sim \beta_8$ を求めることが目的であるが、現実解として $v \leq 1.5$ が所要値である。

3. グリッドコンピューティング環境

著者らの実習室はグリッドを導入し、空き時間を利用していくつかのモデル計算時間の短縮に成果をあげている。グリッドを実現するAD-POWERs (大日本印刷株式会社)⁴⁾ は、処理に時間を要するジョブや計算を同一ネットワークセグメント内のパソコンに分散処理させるミドルウェアであり、Windowsパソコンのスクリーンセーバとして稼動する。導入は、市販のアプリケーションをパソコンにインストールするのと同じで簡単である。

Fig.1 にグリッドのシステム構成、Table.1 にパソコンのスペックを示す。マスターパソコンはデータのとりまとめ役を担うが、特別なパソコンではなく、スクリーンセーバで計算をおこなうボランティアパソコンと同じである。マスターパソコンが故障すれば、ボランティアパソコンの1台をマスターパソコンにすればよい。なお、各パソコンは同一性能である必要はない。

動作の仕組みは、ボランティアパソコンのスクリーンセーバが稼動すると、マスターパソコンから計算するモデルと入力データをもらう。計算終了後、結果をマスターパソコンに送信する。計算すべきデータがなくなるまで (すなわち、入力データに対する計算結果がすべて揃うまで) この繰り返しである。

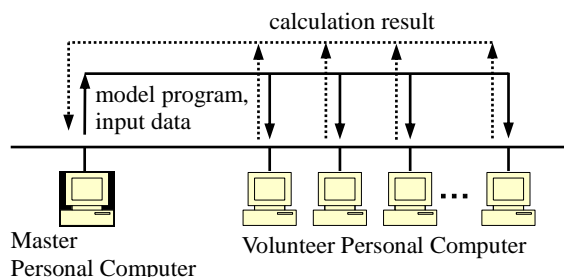


Fig.1 Grid Computing System

Table.1 The specifications of the Grid's PC

Software	OS	WindowsXP
	Grid	AD-POWERs
Hardware	CPU	Celeron 2.93GHz
	Memory	512Mbyte
	Number	20 (same specifications)

実装の作業は、ボランティアパソコンそれぞれにジョブや計算プログラムおよび入力データをインストールする必要はなく、マスターパソコン1台での作業である。その他の処理はすべてグリッドのミドルウェアが行う。プログラム開発および実行のコストは1台で行う場合とほとんど同じである。

なお、計算開始後に授業などが始まってグリッ

ド計算を停止する必要はない。スクリーンセーバになれば再開されるので放置しておけばよい。

4. 計算結果

式(1)の v を最小にする $\beta_1 \sim \beta_8$ を求める手法として、現状では乱数による探索により所要の解を得ている。 $\beta_1 \sim \beta_8$ を乱数で与えることを1回とし、1データに対してこれを10,000回繰り返す。その結果、 $v \leq 1.5$ となるパラメータがこれまでの結果では、平均して3回発見できる。回数を多くすれば、さらに v を最小にする $\beta_1 \sim \beta_8$ を見つけることができるかもしれない。しかしながら、グリッドのパソコンに対して性能が約3倍 (ベンチマークのフリーソフトCrystalMarkによる) のパソコン (スペックをTable.2に示す) を用いても1データに3時間10分を要するため、10,000回は現実的な回数である。このパソコンで20データに各10,000回を計算した場合、約3日間 (63時間20分) を要する。

Table.2 The specifications of the Standalone PC

Software	OS	Windows Vista
	Grid	---
Hardware	CPU	AMD Dual Core 5600+, +20% up
	Memory	2Gbyte
	Number	1

グリッドの場合、1データに4時間35分を要するが (パソコンの性能は約1/3であるが計算時間は3倍にならない)、20台のパソコンがそれぞれ同時に計算するため20データでも所要時間は同じである。

計算の所要時間の比較をTable.3に示す。

Table.3 The comparison of the calculation result.

PC	Number	1 data	20 data
Grid	20	4 hr 36 min	←
Standalone	1	3 hr 10 min	63 hr 20 min

5. おわりに

研究室における高速なパソコンに比べ、グリッドのパソコンは性能が劣るために1データあたりの計算時間は長くなるが、一度に20データの計算を行うことができるため、結果的に約15倍の効率化を図ることが可能となった。

参考文献

- 1) Nelder, J.A and R.Mead : A Simplex Method for Function Minimization, Computer J., Vol.7, 308/313, (1965)
- 2) 新村, 坂本, 小沢 : Grid Computing による電力市場の価格予測, 電気学会電力エネルギー部門大会, 八戸工業大学, 9/12-14, (2007)
- 3) 坂本, 小沢, 新村 : グリッドコンピューティングによる加熱炉の温度計算, 計測自動制御学会産業論文集 (技術・開発ノート), Vol.6, No.17, 135/138, (2008)
- 4) AD-POWERs 紹介サイト:
<http://www.dnp.co.jp/cmc/ad-powers/index.html>