反復的データ分割による数値シミュレーションの静的負荷分散

藤村 佳克，市川 周一
豊橋技術科学大学　工学研究科　知識情報工学専攻

Iterative Partitioning Scheme for Static Load-Balancing of Numerical Simulations

Yoshikatsu Fujimura and Shuichi Ichikawa
Dept. of Knowledge-based Information Engineering, Toyohashi University of Technology

並列処理や分散処理において，通信時間を考慮して計算負荷を分散することは必須である．市川ら [1][2] は，偏微分方程式の並列求解システム NSL を例として，通信と計算の双方を考慮して実行時間を最小化する静的負荷分散法を示した．この研究では，複数の矩形のブロック (格子点配列) 上でそれぞれ計算が行われ，ブロック間でデータを交換するために通信が発生するという計算モデルが用いられている．しかし上記 2 つの研究はデータ分割を基本としているため，ブロック数 m よりも要素プロセッサ数 n の方が多い場合しか扱えない $(m \leq n)$．

この制約を除去するため箱詰めモデルによる負荷分散 [3] を試みた．この方法は基本的に実行時間最小化マルチプロセッサスケジューリング問題の一種になる．この方法で一定の成果を得ることはできたが，研究 [1][2] とは逆に，$m \geq n$ という条件下でしか満足な負荷分散を行えないという問題がある．

そこで本研究では，箱詰めによる負荷分散とデータの自動分割を組み合わせて，m と n の関係によらず適用可能な静的負荷分散手法を検討する．基本的なアイディアは以下の通りである．

箱詰め法では m 個のブロックを n 個のプロセッサに振り分けるため，$m < n$ なる条件下では m 個のプロセッサだけが利用される．このとき最も計算時間の大きいブロックがボトルネックとなり全体の処理時間を決定する．そこで最も計算時間の大きいブロックを等分割し，ブロックを $m + 1$ 個に増やしてから改めて n 個のプロセッサに詰め直す．この処理を繰り返して実行時間を改善する．

1 回の繰り返し毎にブロック数が 1 増加するため，箱詰めによる負荷均衡は行いやすくなり，プロセッサ間の計算時間の差は減少してゆく．一方，ブロックを分割することによりブロック間通信が新たに発生するので，最適なブロック数を超えると大局的には実行時間は増加に転じる．そこで実行時間が増加をはじめたら繰り返しを止めて終了する．この方法は一種の反復改善法 (Iterative Improvement) なので，以下 Improve と呼ぶ．

もちろん実際の実行時間の増減はそれほど単純ではない．問題が離散最適化であること，箱詰めにヒューリスティックを使っていることなど，色々な要因により細かく変動する (多くの局所解をもつ)．そこで一度実行時間が増加しても，局所解に落ちることを避けるため更に分割を続けて改善がないか調べる．1 回の悪化まで許して分割を続ける方法を Improve1 とし，以下 4 回悪化しても分割を続ける Improve4 までを検討の対象とした．

図は本手法に基づく負荷分散結果の一例である．能力が不均一なプロセッサ 8 台 (性能比 4:3:2:1 で各 2 台) を用いて負荷分散を行った場合の結果で，横軸はブロック数 m，縦軸は実行時間比 (一番速いプロセッサ 1 台で実行した場合の実行時間を 1 として正規化した結果) を表している．Approx5, Approx5+Local は過去の研究 [3] で行った箱詰め法による結果である．今回の反復改善においても何らかの箱詰めアルゴリズムは必要なので，箱詰めには Approx5 を用いた．

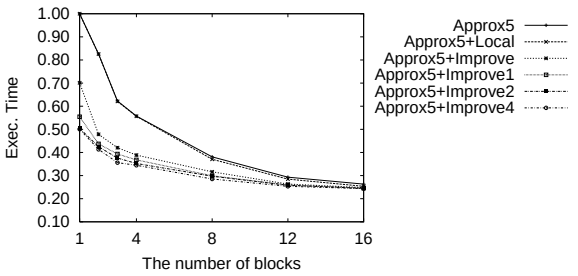

図 1: 負荷分散結果の一例

本研究の手法では，過去の箱詰め法 [3] で満足に扱えなかった $m \leq n$ の領域において良好な負荷分散結果を得ている．局所解を避けるために，2 回程度までは解が悪化しても反復改善を続けると良いこともわかる．実行時間の下界は明らかに 0.2 であるが，本研究のような単純な手法でも充分精度の良い負荷分散が得られている．

本手法による負荷分散の計算時間は，図に示したような条件下において Pentium-II 400 MHz の PC を用いて 0.1 ~ 10 秒程度である．実行時間の面でも充分実用的な手法であるといえる．

参考文献

[1] 市川周一，川合隆光，島田俊夫: 組合せ最適化による並列数値シミュレーションの静的負荷分散，情報処理学会論文誌，Vol.39，No.6，pp.1746-1756 (1998)．

[2] 山下真史，市川周一 :分散処理環境における数値シミュレーションの静的負荷分散手法，情処研報 99-HPC-77, pp. 179–184 (1999)．

[3] 藤村佳克，市川周一 :並列数値シミュレーションの静的負荷分散手法の拡張について，情処研報 99-HPC-77, pp. 185–190 (1999)．