

高負荷オブジェクトシミュレーションの動的負荷分散化*

4H-6

畠山正行 石川博行†

茨城大学工学部情報工学科‡
 {masayuki, st94005}@cis.ibaraki.ac.jp

1 はじめに

近年、分散処理計算がさかんに研究されているが、例えば数値シミュレーションを用いて研究をしている人が分散処理をさせたくてもプログラムするのに煩雑な工夫をして実装しなければならない事が多く、十分に広まっているとは言にくい。

そのような問題を解決するために仮想分散計算機環境が我々の研究グループの直井によって開発された[1]が、まだ本格的な数値計算に適用されて本格的に稼働をしていない。そこで、このシステムの機能の検証および多少の性能・特性等の改良を考え実装した。

2 仮想分散計算機システム

昨年開発されたこの仮想計算機システムはオブジェクト指向パラダイムに基づいた機構を導入しており、ユーザにとってネットワークで繋がれた複数の計算機があたかも一つの計算機にアクセスするのと同様の環境を目指したものである。[1] このシステムでは学習機能を持たせて動的負荷分散を実現している。このシステムでは主要要素として以下で述べるケルビムや黑板モデルという考え方を取り入れている。

2.1 ケルビム [1]

ケルビムとはエージェントモデルによって実現された、バックグラウンドで動く分散処理用プロセスである。仮想計算機システム上には、図1のように、2種類のケルビムがあり、一つは実際にメソッドを実行、処理するためのケルビムであり、もう一つは、ケルビムを管理するためのケルビム管理ケルビム (CMC) である。CMCはケルビムの起動やメソッドの解釈を担当し、ケルビムはCMCによって活性化され、CMCからの情報によって処理を行なう。また、これらのケルビムの学習によって環境

内のどの計算機によって実行されるか、またいくつケルビムを活性化するかがCMCによって決定される。

2.2 黑板モデル

ケルビムやCMCはその通信手段として黑板を用いている。従ってそれらの間の通信路は唯一この黑板のみであり、この黑板を通して情報がやり取りされる。黑板にはユーザが直接アクセスできるオブジェクト黑板と、システムが使用する共有黑板がある。この黑板は"gdbs"というデータベースを用いることによって実現している。

図1に簡単な概要を示す。この図はケルビムと黑板の関係を表している。例えばCMCは共有黑板 (Shared Status Blackboard) に見に行き、その情報を基にケルビムインタフェース (図1中の"C-I/F") に必要な数のケルビムを生成させる。

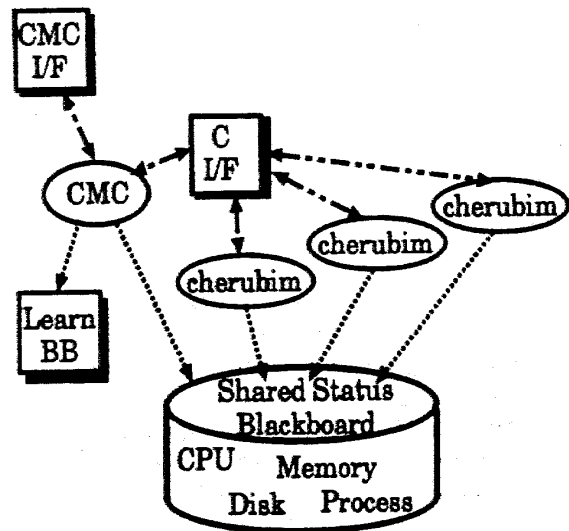


図1: ケルビムと黑板の関係

これらにより、共有黑板にそのマシンの状態を揭示しておき、その情報に従いケルビムの生成を制御することによって動的な負荷分散が可能になっている。つまりそのマシンのリソースの変化によって、処理するためのデータ量の調整が出来る。

しかし、システム的设计上、必ず黑板というものを通さなければならないので、ある計算を行うと

*Distributed, Dynamic Load Balancing System for Heavy Load Object-based Simulations

†Masayuki Hatakeyama, Hiroyuki Ishikawa

‡Department of Computer and Information Sciences, Ibaraki University, 4-12-1 Nakanarusawa, Hitachi-city, 316-8511 JAPAN

きに分散されたオブジェクト間での通信がそれぞれの計算に比べて、頻繁に行われるものはあまり向いていない。しかし反対に1つのオブジェクトがある程度まとまった計算をするものであれば、成果が期待できるといえる。

3 高負荷計算の実装

そこで仮想計算機システムに計算をさせる対象として比較的分割計算し易く、分割されたオブジェクト間の通信量の少ない DVO シミュレーション計算を用いることにした [2]。DVO 計算では、はじめに物理空間をセルと呼ばれる幾つかの微小空間に分割する。物理空間セル内には分子が多数存在し、それぞれの分子の持つ速度は様々である。そこで各セル内では、ボルツマン方程式の衝突計算を離散化する離散化速度座標 (Discrete Velocity Ordinate) という概念を用いて速度空間を分割して離散化計算をする。

ボルツマン方程式は分子の衝突前の状態と衝突後の状態との変化を計算する衝突項計算と、分子の空間的移動を計算する流動項計算からなる。

衝突項の離散分割数にもよるが、計算機間でのセルデータの転送通信時間は数秒のオーダーである。一方、各セル内の衝突項内部計算に要する時間は数百秒以上である [3]。このことにより、本仮想分散計算機上において実現するためには、セルを最小のオブジェクトとみなし、計算量の多い衝突項計算の部分に注目した。つまり、衝突項の部分分散計算することにより、高速化を図る。その概念図を図2に示す。

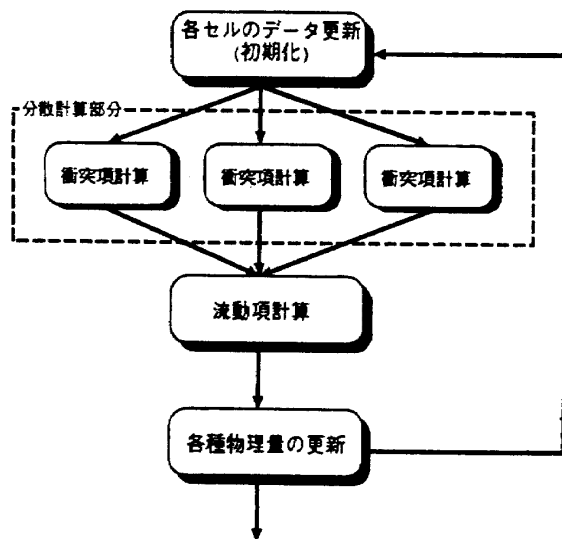


図 2: DVO の分散計算 (1 ステップ)

これらの分析に基づいて元々の計算を分散環境

用書き替え、また仮想計算機のライブラリを組み込んで行った。

4 問題点と改良

当初、実装してみたところこのシステム [1] のままでは計算時間がかなりかかった。この原因は実装されている黑板にはデータベースが用いられているが、各 DVO セル計算データの読み書きだけに要する時間が 2~3 時間以上かかってしまい、実用に耐えないことが判明した。

そこでこの部分の解決策として、この膨大なデータだけは黑板を使用することを断念して、通常ファイルとしてやりとりすることにした。しかしこれだけでは、データの読み書きの部分において同期制御その他の問題が起こると考えられるので、黑板にファイルの読み書きのための実行状況を表示し、それを通してデータの転送をする事にした。勿論そのほかのデータについては従来通り黑板を用いることにした。

この結果、計算時間は飛躍的に速くなり、単独での計算と比べ、2 台で約 1.7 倍の速度の向上になった。また DVO シミュレーションを実装する上において、先ほどの問題を除いては本システムの理想通りに比較的スムーズに行えた。

5 結論

この研究により、直井の開発した仮想計算機システムが本格的な数値計算においても実用的に使えることが分かった。

また PVM などの他の分散システムと比べても、数値計算を実現したいユーザがあまり分散計算の知識がなくても元々の数値計算の考え方をあまり変えずに分散計算が実現できた。

参考文献

- [1] 直井 稔: “エージェントを用いたオブジェクトベースシミュレーション仮想計算機システム”, 平成 8 年度茨城大学大学院理工学研究科情報工学専攻学位論文
- [2] 高阿田 昭宏: “ボルツマン方程式に対する離散的数値解法の研究”, 平成 6 年度茨城大学大学院理工学研究科情報工学専攻学位論文
- [3] 畠山正行, 進藤祐希, 宍戸実, “PVM を用いた高負荷シミュレーションの動的負荷分散の実現”, 情報処理学会第 54 回全国大会, pp.3-445 ~3-446, 5N-5, 1998