

連続事象シミュレーションにおける並列化アルゴリズムの実験評価

滑川光裕[†], 後藤和則^{††}, 森秀樹^{†††}, 佐藤章^{†††}

[†]嘉悦女子短期大学, ^{††}東京職業訓練短期大学校, ^{†††}東洋大学

近年、さまざまな分野で並列、分散処理が行われている。シミュレーションの分野でも、パフォーマンスの向上による高精度化や速度の実用レベルを目指して、さまざまな並列化が行われている。しかしながら、汎用計算機を利用した研究開発では、それらのほとんどが離散事象によるものである。連続事象によるものは、処理効率の向上が期待できないために、あまり扱われていなかった。

我々は、シミュレーションモデルの持つ潜在的な並列性から、同期処理を省略することで、各プロセッサの処理効率を高める手法を提案し、その実験と検証を行った。

A verification of synchronization algorithm for continuous event

Mitsuhiro NAMEKAWA[†], Kazunori GOTO^{††}, Hideki MORI^{†††}, Akira SATOH^{†††}

[†]Kaetsu Women's college, ^{††}Toyo Institute Politechnique Univ., ^{†††}Toyo University

Parallel simulation has contributed to reduction of execution time for carrying simulations for large scale and complex systems. Most of those parallel simulations have been conducted on the systems which have been represented only through discrete changed model. A few algorithms have been proposed for simulation using the continuous changed model. However, an universal and effective algorithm has not yet been established for parallel simulation with continuous changed model or combined model. We propose an new algorithm in synchronization method of parallel simulation with continuous changed model. This paper describes a new algorithm and its verification.

1. はじめに

シミュレーションの処理効率を向上させ、実行時間の短縮を行い、これまで不可能とされていた大規模かつ複雑なシステムにおけるシミュレーションの実行や微視的モデルが要求されるシステムでのシミュレーションを可能とする方法として、並列シミュレーションが注目されている。

並列シミュレーションは、シミュレーションの持つ並列性を抽出し、複数プロセッサに負荷

を分散させることにより、シミュレーション処理の効率を上げるものであるが、これまで行われてきた並列シミュレーションほとんどが離散事象によるものであり、連続事象を対象としたものは、ほとんど見あたらなかった。それは、連続事象の並列シミュレーションが原則的にシミュレーションクロック毎の同期処理を必要とするため、並列化による処理効率が極めて悪いことにその原因があった。

我々は、シミュレーションモデルを3次元空

間の中を流体が流れるような状況を再現するモデルで表現される空間フロー型システムと、ある種のトランザクションがその空間を移動するトランザクションフロー型システムに分類し、社会一般の現象のほとんどを当てはめることの出来るトランザクションフロー型システムにおいて、そのトランザクションの移動予測から同期処理を省略し、効率の良いシミュレーションが可能となる同期方法について提案し、その正当性と有効性を検証した。

2. 並列シミュレーションの方式

並列シミュレーションでは、共通する特定の機能を分離独立させ、それを並列化して処理する方式と、シミュレーション対象となるシステムを空間的に分割し、分割された部分空間ごとに処理を分散させる方式とが、一般に考えられる。前者を機能分散方式、後者を空間分散方式と定義する。

その場合に、負荷が分散された計算機間で、論理的な矛盾をなくし、シミュレーション時刻の順序を正確に保持するための処理が必要となり、そのオーバーヘッド時間が処理効率上大きな問題となる。機能分散でのオーバーヘッド時間は、共通する機能へのアクセス回数とその処理に依存し、空間分散のそれは隣接する部分空間とトランザクションの転移によって決まる。同一機能・性能をもった計算機による並列処理では、どのような分散形態を採ったとしても、シミュレーションのみに関わる処理時間は不変である。したがって、空間分散方式による並列シミュレーションでは、シミュレーション処理時間に無矛盾処理や同期処理のために必要とする準備処理および通信時間、待ち時間などのオーバーヘッド時間の総和を加えたものとなる。

なお、機能分散方式において、共通する機能を処理するための高速処理能力をもつ専用のプロセッサが開発されれば、通信時間に加え、別の観点からの評価も必要となる。

3. 同期処理方式

本章では、並列シミュレーションの同期方法について説明する。

3.1 対象とするシステムと分散方法

自動車を微視的なモデルとして表現される道路交通システムシミュレーションなどは連続事象シミュレーションの典型と考えられる。連続事象シミュレーションの特徴は、システムの変化が主として自動車の連続的な変化(移動)に依存する。したがって、自動車の移動が連続変化モデルとして記述されるシミュレーションとなる。

道路交通システムは、空間分散方式の一つであると考えられるが、ここでは平面性をより強調するために地域分散方式と呼ぶこととする。

この地域分散方式による並列シミュレーションでは、シミュレーション対象となる地域を互いに素な地域に分割する。分割された各地域にはそれぞれプロセッサが割り当てられ、対応する地域のシミュレーションが行われる。したがって、シミュレーション全体を移動する自動車は、地域(プロセッサ)間を渡り歩く(これを転移と定義する)ことになる。しかし、これらの転移は、自動車の密度にもよるが、シミュレーションクロックごとにあるとは限らない。つまり、地域間の自動車の転移は離散的な事象になっていることが分かる。連続事象モデルであるが並列シミュレーションの同期そのものは離散的な事象として表現される。このプロセッサ間の自動車転移が、あたかも単一プロセッサによるシミュレーションと同じ結果になる並列シミュレーション方式が肝心であり、しかもより効果的なシミュレーションが要求される。シミュレーションの品質を保持し、さらに高速処理が要求されることになる。

3.2 転移予測の概念

地域間の自動車転移が、無矛盾的で、同時刻に行われるためには、事前に転移の時刻が正確

に分かっていることが必須である。しかし、一般に自動車の移動は確定的でないので、円滑かつ効率的な転移予測は不可能である。そこで可能な限り、より正確な転移予測を行うことによってそれを達成することを考えた。実際の転移時刻よりも早めの予測は、実行時間の増加につながるが、矛盾のない同期処理が保証される。

ある一つの地域から他の地域への転移は、現在位置・移動方向・移動速度によって、何時ごろ、どの部分空間へ到達できるか、おおよその予測がたてられる。自動車の移動は、ランダムに変化し、それぞれに応じた移動予測が必要である。この転移を円滑に行うためには、両地域間のシミュレーションクロックが同じでなければならない。そのために、図1に示すように、

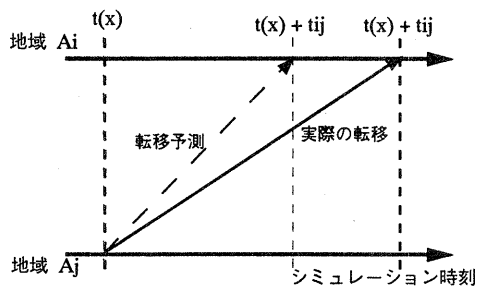


図1 転移の予測と実現

転移予測は実際の転移より早い時刻であることが必須であり、もし、遅い時刻の予測であれば、この方式が適用できなくなる。1回目の予測が外れたときには、図2に示すように、何回かの予測をすることで円滑な転移が実現できる。この方式による同期を行えば、同期時刻までは、各地域が他の地域に関係なく独自のシミュレーションを実行することができ、したがって、毎回のシミュレーションごとの同期処理が不要となる。それによって、通信回数が減り、そのために、全体のシミュレーション実行時間も減少する。

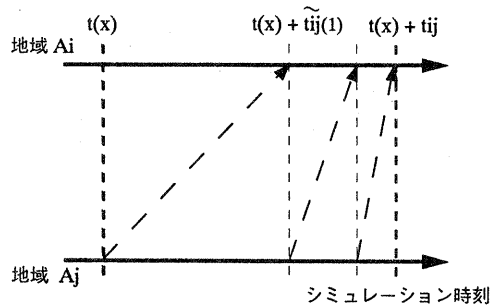


図2 事象予測の繰り返し

4. 連続事象並列シミュレーション

我々は、アルゴリズムの検証を行うために具体的なシステムを対象とする実験を行った。

4.1 道路交通システムモデル

微視的モデルをもつ道路交通シミュレーションでは、信号変化、自動車発生などの離散変化モデルと自動車の連続変化モデルが混在する。自動車の動きを簡略化して記述すれば、離散事象変化モデルとしてとらえることも可能であるが、一般道路、特に渋滞道路を対象とする場合には、自動車モデルは人間の運転モデルを適用したシミュレーションクロックごとの連続変化モデルが必要と考えられる。

4.2 シミュレーション方式

道路交通システムにおける適用では道路交通網を分割することになる。平面的な道路交通網を n 個の互いに素な地域 A_i に分割し、それぞれの地域ごとにプロセッサ P_i を割り当てる。

道路交通システムでは、この分断により、地域間にまたがる道路が分割されることになる。そして転移予測は、道路を走行する自動車がこの分断された部分に差し掛かり、隣接する地域に転移するタイミングを予測することになる。

説明を容易にするために図3のように3つの単純な道路交通網の分割を例として考える。A

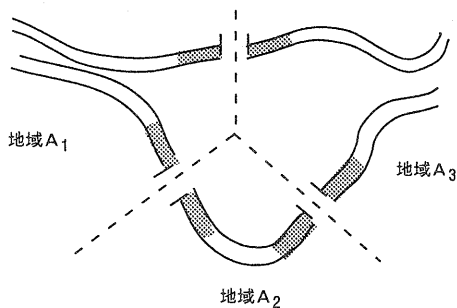


図3 道路ネットワークの地域分割

A_1 、 A_2 、 A_3 の各地域におけるシミュレーションを、 P_1 、 P_2 、 P_3 の各プロセッサが担当することになる。このときに例えば A_2 は、 A_1 と A_3 と隣接しているので、まずはじめにそれぞれの地域間で自動車移動する転移予測を行

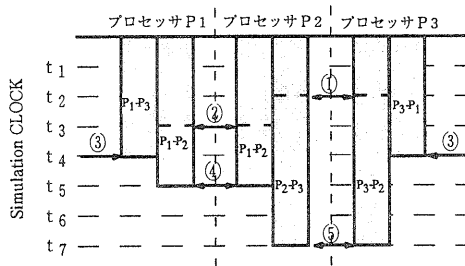


図4 同期処理タイミングの例

う。次に、図4のように A_2 を担当する P_2 が P_1 と P_3 といつ同期処理を行うかというシミュレーションクロックを計算し、それらの中の最早時刻（部分独立時刻）を決定し、その時刻までは A_2 内部のみで処理を行う。いわゆる A_2 の独立時間単位である。この図では、 P_3 との同期処理が先となり時刻 t_2 までは、プロセッサ P_2 が独立して処理を行うことができることを示している。時刻 t_2 になったならば、 P_3 と

の同期処理を行い、次の同期処理時刻を転移予測をもとに決定することになる。この例では、時刻 t_7 まで P_2 と P_3 は、お互いに独立してシミュレーション処理できることになっている。

5. 実験の評価方法

提案したアルゴリズムの検証とその評価を行うために、段階を踏んだ2種類のシミュレーションシステムを作成し、それぞれのシミュレーションを行なった。一つは、一定地域を対象とする微視的なモデルによる道路交通システムの具体的なシミュレーションである。1台のプロセッサによる単体のシミュレーションおよび負荷分散・並列化した2台のプロセッサ上で、それぞれシミュレーションを行い、それらを比較することによってアルゴリズムの検証を行った。このシミュレーションを「実際の並列シミュレーション」と称する。

もう一つは、並列化に対するアルゴリズムの有効性を確認・評価するためのシミュレーションである。実際のシミュレーションから、並列シミュレーションでのオーバーヘッド時間の構造的なデータ、およびマイクロモデルによる道路交通シミュレーション実行に要した種々のデータを取得し、バーチャルな新たなシミュレーションを1台のプロセッサ上で行った。これを便宜上、「バーチャルな並列シミュレーション」と称する。このシミュレーションでは、パラメータ値を変更することによって並列化の効果を容易に評価することができる。

並列化の効果を評価するために、バーチャルなシミュレーションを行った。並列しているプロセッサ間でのシミュレーション時刻の同期に関係する部分、たとえば同期に必要とする処理や通信などを重点とするモデルから構成されている。このシミュレーションでは、プロセッサの台数をパラメータで指定することによって、それに対応した分散並列環境が自動的に作成され、評価のためのシミュレーションが実行される。

5.1 並列シミュレーションのトランザクション分布

一定地域を分散して並列シミュレーションする場合に、分散化された各プロセッサにトランザクションの分布が均等化されていれば、並列化の効果は大きくなる。しかし、一般的には、トランザクションの移動がランダムであるために、トランザクションの分布に偏りがでてくる。そのために、提案したアルゴリズムの一般性を確認するために、「トランザクションがほぼ均等化している場合」と「トランザクションが不均等化している場合」の2種類のケースについてのシミュレーションを行った。

シミュレーションは、60 kmの円周道路を想定し、それを等分に分割して並列化し、走行速度は、(40 ± 5) km/hの1様分布による走りとした。「トランザクションがほぼ均等化している場合」では、自動車は一定分布で発生させ円周道路を走行する。一方、「トランザクションが不均等化している場合」では、不均等な分布が特定の地域に2倍近いトランザクションを処理させる状況を作る環境を設定した。

5.2 評価方法

並列化の効果を評価するための一般的な方法は、シミュレーション実行時間の速度向上率である。われわれは、速度向上率に加え、アルゴリズムの効果を確認する意味で、オーバーヘッド時間の原因となる、プロセッサ間通信回数と同期のための相手待ち時間を解析・評価した。

6. 実験の評価

図5に負荷がほぼ均等化している場合の結果を示した。新アルゴリズムの有効性を見るために、完全同期型との比較をした。新アルゴリズムは、並列数の増加に伴い、そのシミュレーション実行時間が1台から2台へは53.3%、2台から4台へは51.5%というように、プロセッサ数の増加にほぼ比例して効率が良くなっている。完全同期の場合は、1台から2台、

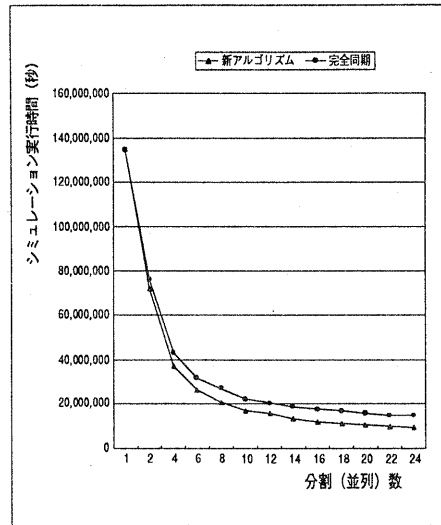


図5 シミュレーション結果 (負荷がほぼ均等の場合)

2台から4台ともに56.4%となっており、新アルゴリズムの方が、より効率化が図られていることがわかる。この傾向は、台数が多くなるほど顕著に表れている。アルゴリズムの違いによる効率化を分かりやすくするために、図6

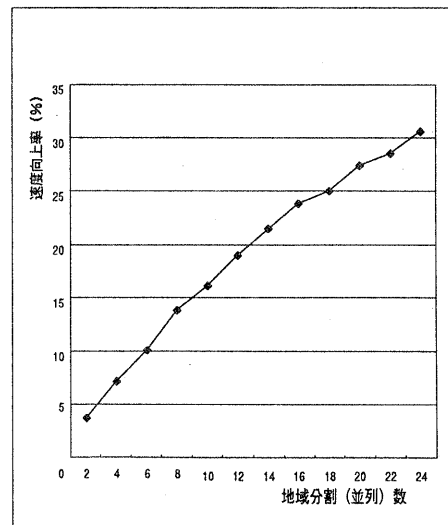


図6 完全同期方式に対する新アルゴリズムの速度向上率

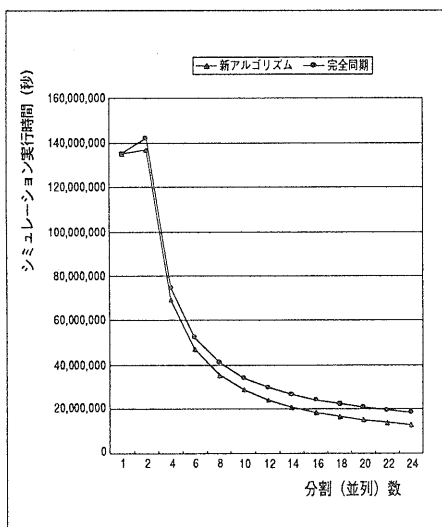


図7 シミュレーション結果 (負荷不均等の場合)

に完全同期に対する新アルゴリズムの速度向上率のグラフを示した。台数に比例して速度向上率が大きくなっていることがわかる。

図7にトランザクションが不均等化されている場合のシミュレーション結果を示す。2台のプロセッサの場合に、単一プロセッサよりシミュレーション実行時間が大きくでている。これは、負荷が不均等なプロセッサに引きずられ、通信時間に依存した一時的な現象である。この場合でも、新アルゴリズムの方が完全同期型よりもよくなっている。

全般的には、図5と図7とを比較して分かるように、トランザクションが均等な分布をしている場合より新アルゴリズムの効果が大きくでていることがわかる。

7. おわりに

本研究では、連続事象並列シミュレーションについて、より品質を保証し、より処理効率の良いシミュレーションが行える新しい同期方式を、渋滞道路交通システムに適用し、連続事象並列シミュレーション環境において実験を行

い、処理効率の効果を検証した。

この新しい連続事象並列シミュレーションの同期方式は、トランザクションフロー型のシステムで、この種のシステムは社会一般に多数存在している。たとえば、鉄道輸送システムから、海の中に生存している魚の生態系のような自然システムなどまで含まれる。

今後は、各種システムについての応用を検討し、より良品質の、より高い効率の同期方式による汎用化と効率化のための解析を行なう予定である。さらに、こうした並列シミュレーションの同期方式をより容易に、より効率良く実験できる仮想的なシミュレーションシステムに対する研究も合わせて行う予定である。

参考文献

- [1] 高井峰生, 山城登久二, 成田誠之助: 離散事象並列シミュレーションにおける保守的同期手法の評価, 情報処理学会論文誌 Vol.37 No.11, 1996.11
- [2] Hassan, R., : Parallel Simulation Using Conservative Time Windows. Winter Simulation Conference, 709-717, 1992
- [3] 猪飼國夫, 板倉直明他: メタ論理を構成することを考慮したハードウェアファジィ推論ユニットの製作, 第5回インテリジェントシステムシンポジウム論文集, 1995
- [4] 猪飼國夫, 本多中二他: 微視的モデルによる道路交通シミュレーションシステム, シミュレーション学会誌「シミュレーション」, Vol.16 No.8, 1997.8
- [5] Misra, 1986, Distributed Discrete-Event Simulation., ACM Comp., surveys. Vol18, pp.39-65.
- [6] M.Namekawa, A.Satoh, et al. : Clock Synchronization Algorithm for Parallel Road Traffic Simulation, International Congress on Modelling and Simulation, pp.1426-1431, 1997.12