

## マルチエージェントシミュレーションにおける 動的経路探索の高速化手法の検討

佐野 義仁†

福田 直樹†

† 静岡大学情報学部

### 1 はじめに

道路交通は、現代における重要な社会システムであるが、大気汚染や交通渋滞などの様々な問題が存在する。そこで、これらの問題を解決するために、交通状況の分析や新しいシステムを導入した際の影響を調べることが重要であり、マルチエージェントを用いたシミュレーションが適用される場合がある。

マルチエージェントシミュレーションではスケラビリティが一つの課題である [4] が、エージェントが状況に合わせて動的に経路探索を行おうとすると処理が複雑になり、シミュレーションの大規模化が容易ではない。そこで、本研究では、コンピュータを構成するパーツの一つである GPU を汎用演算に利用する GPGPU を用いて動的経路探索を並列的に処理することによって高速化を行う。

### 2 関連研究

道路交通などをマルチエージェントを利用したシミュレーションにより解析可能とするシステムとして、ベルリン工科大学とスイス連邦工科大学によって共同開発されている MATSim がある [1]。MATSim は移動主体である車をエージェントとし、そのエージェントの相互作用として交通現象などを再現する。

MATSim では道路網を、リンクとノードを用いて構成する。リンクとノードによって構成された道路網をエージェントが移動する。各エージェントは、それぞれ行動プランを持ち、行動プランに従って行動をする。行動プランで指定された目的地まではダイクストラ法によって決定される。

エージェントの行動プランは XML 形式で記述される。XML 形式で行動プランを記述すると図 1 のようになる。行動プランは滞在状態を表す act, 移動状態を表す leg を交互に記述することによって構成される。行動プランは国設調査に基づく census データに対して、MATSim の API を利用して XML 形式に変換することによって作成することができる。

このような方法によって、MATSim では、およそ 300 万規模のエージェントによるシミュレーション結果を用いた解析事例が報告されている [3]。

<person id="10"> ← 定義するエージェントに割り当てる ID

```
<plan>
  <act type="home" x="15" y="10" end_time="8:00:00" />
  <leg mode="car" />
  <act type="work" x="60" y="25" end_time="17:00:00" />
  <leg mode="car" />
  <act type="home" x="15" y="10" />
</plan>
```

</person>

図 1: 行動プラン記述例

これ以外に、歩行者などの群衆を対象としたマルチエージェントシミュレーションの大規模化も検討されている [4]。この方法では、空間を一つの線としてとらえ、空間内でのエージェントの動きを制限することによって計算処理を軽減し、大規模化を行う。しかしながら、いずれの場合でも、エージェントが動的に経路探索をやり直す場合に対して、効果的に実現する方法は用意されない。

### 3 シミュレーションへの動的経路探索の導入

MATSim では、通常は、日単位でのシミュレーションを行う際に、シミュレーション開始前に各エージェントの持つプランに対してどのような経路を通りプランを実現するかを計算する経路探索処理の実行を行うので、各日内のシミュレーションの途中で動的な移動経路変更を行わない。MATSim 上で、日ごとのシミュレーション内での動的経路探索を行えるようにするために、Within-day replanning API が用意されている [2]。

Within-day replanning API では、API を呼び出したエージェントの周囲の地図データ等を利用して、経路探索が可能である。しかしながら、数百万を超えるエージェントが動作するような環境では、各エージェントが状況に応じて経路探索を動的に行うことは計算負荷が非常に大きくなるため、その負荷を低減し、処理のスケラビリティを向上させるための機構が必要となる。本研究では、そのための機構として、各エージェントの経路探索アルゴリズムを GPGPU 上でも実行可能なプログラムとして記述できるようにするためのフレームワークを準備することを考える。

## 4 GPGPU を利用した動的経路探索処理

OpenCL でのプログラミングモデルでは、GPU 上で実行されるプログラムで行う処理は、あらかじめカーネルプログラムという特殊な形式で実装しておく必要がある。図2は、C 言語上で OpenCL に基づいて経路探索を行うプログラムの実装例である。

本研究では、GPU を利用した演算を行うためのフレームワークとして、OpenCL を利用する。OpenCL を用いたプログラムは一般に C 言語等で記述されるが、MATSim では Java 言語を用いてシミュレーションプログラムを記述するため、Java 言語でも OpenCL を利用できるようにする必要がある。そこで、本研究では Java 言語上で OpenCL を利用できるようにする枠組みである `joctl.org` を用いる。

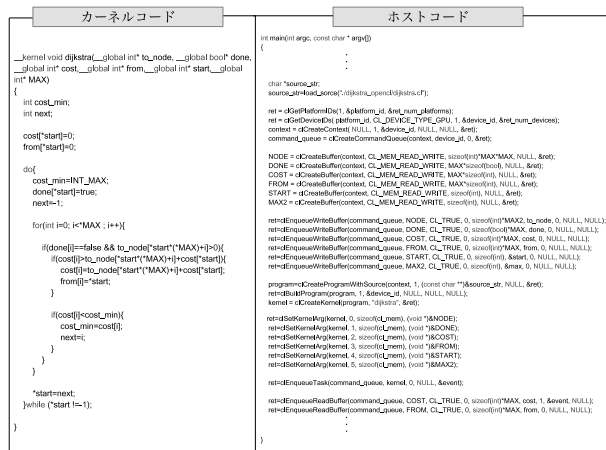


図2: OpenCL 上の C 言語での経路探索のコーディング例

図3は、MATSim への GPU を用いた動的経路探索の実装モデルである。MATSim の Within-day replanning API から `joctl.org` を経由して、GPU 上で実行されるカーネルプログラムが呼び出される。

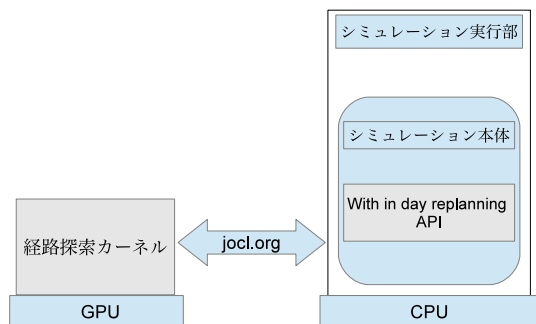


図3: MATSim への GPU 動的経路探索の実装モデル

## 5 経路探索処理のベンチマーク・評価環境

MATSim では、エージェントの動的経路探索以外に他の多くの処理も行っているため、OpenCL による動的経路探索の実装方法やチューニング方法ごとの効果の違いを検証することが、必ずしも容易ではない。本研究では、エージェントの動的経路探索部分の OpenCL 化に対する効果を検証・評価するための実行環境も、別途実装した。本実行環境は C 言語で記述しており、MATSim とほぼ同様の道路網などを入力として受け取り、簡易な評価シミュレーションを行うことが可能となっている。

シミュレータでは、OpenCL を利用して経路探索を行う場合での並列度の向上などを GUI などを用いて確認することができ、GPU を使用するよう指示された場合には、そのパラメータに応じた動的経路探索処理が GPU を利用して行われるようになる。

## 6 まとめ

本研究では、シミュレーションの実行に GPU を演算処理に利用する GPGPU を適用することによってシミュレーションの高速化について検討した。特に、エージェントがシミュレーションの途中に行う動的経路探索について GPGPU を適用することによっての高速化について検討した。

## 参考文献

- [1] M. Balmer, K. Meister, M. Rieser, K. Nagel, and K. W. Axhausen. Agent-based simulation of travel demand: Structure and computational performance of matsim. In *Proc. the 2nd TRB Conference on Innovations in Travel Modeling*, 2008.
- [2] Enrique de la Hoz, Ivan Marsa-Maestre, Miguel A. Lopez-Carmona, and Pablo Perez. Extending matsim to allow the simulation of route coordination mechanisms. In *Proc. The 1st International Workshop on Multi-Agent Smart Computing (MASmart 2011)*, pp. 1–15, 2011.
- [3] Ryo Kanamori, Takayuki Morikawa, and Takayuki Ito. Evaluation of special lanes as incentive policies for promoting electric vehicles. In *Proc. The 1st International Workshop on Multi-Agent Smart Computing (MASmart 2011)*, pp. 45–56, 2011.
- [4] 山下倫央, 岡田崇, 野田五十樹. 大規模群集流動の制御に向けたシミュレーション環境の構築. In *Joint Agent Workshop and Symposium (JAWS)*, 2012.