

マルチエージェントシミュレーション統合のための シミュレータアーキテクチャ

中島 悠^{1,a)} 服部 宏充¹

受付日 2012年1月30日, 採録日 2012年7月2日

概要: マルチエージェントシミュレーションは人々の行動が創り出す複雑な社会現象の再現および分析に用いられている。都市における人々の活動は多様であり、対象とする領域の面、対象とする行動の抽象度の面で異なる多くのシミュレーションが実施されている。それぞれの事象は関係があるにもかかわらず、それぞれのシミュレータは個別に構築されてきた。このようなシミュレータという計算可能な形で集積される知識を再利用するために、対象とする問題領域や抽象度が異なる複数のシミュレータを統合することを目指す。本研究では、複数の異質なシミュレータを統合するプラットフォームのアーキテクチャを提案する。さらに、都市交通と都市電力を例に、このアーキテクチャが、抽象度の異なるシミュレータの統合と問題領域の異なるシミュレータの統合を果たせることを示す。

キーワード: シミュレータ統合, シミュレータアーキテクチャ, マルチエージェントシミュレーション, 都市交通, 都市電力

Simulator Architecture for Integrating Multiagent-based Simulations

YUU NAKAJIMA^{1,a)} HIROMITSU HATTORI¹

Received: January 30, 2012, Accepted: July 2, 2012

Abstract: Multiagent-based simulations are regarded as a useful technology for analyzing complex social systems and have been previously applied to various problems. Tackling the problems in an urban environment involves various levels of abstraction and various target domains. Different types of simulator are developed separately by specialists in their respective fields. There is a need to integrate simulators that offer different levels of abstraction and cover the various target domains to cover the multiplicity of actions in an urban environment. This paper introduces the architecture of a simulator integration platform and demonstrates the capabilities of the platform in an inner city traffic and electricity domains.

Keywords: simulation integration, simulator architecture, multiagent simulation, city traffic, city electricity

1. まえがき

マルチエージェントシミュレーション (Multi-agent based simulation: MABS) は都市における複雑な社会現象を分析する有力な技術である。社会では多様な性質を持った主体が自律的に行動し、その行動結果が社会的な現象として現れる。そのため、1人1人の人間をモデル化したエージェ

ントの間のインタラクションを観測する MABS のパラダイムは、社会現象のシミュレーションに適すると考えられている [5]。MABS は、交通、防災、社会科学など様々な分野で適用されてきている [1], [4], [9]。

現代社会において、交通は人々のインタラクションにより形成される複雑な社会現象の1つである。交通という1つの問題領域でも、人々の行動は異なる抽象度でとらえられる。たとえば、抽象度の高い広域交通行動モデルに基づいたシミュレーション [2], [13] がさかんに行われている一方で、抽象度の低い運転行動モデルに基づいたシミュレーション [6], [11] も行われている。しかし、その異なる抽象

¹ 京都大学大学院情報学研究科社会情報学専攻
Department of Social Informatics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan

^{a)} nkjm@i.kyoto-u.ac.jp

度にまたがるシミュレーションの研究は少ない。また、交通は都市における人々の活動の基盤であり、災害時の避難、人同士の接触による疫病伝播といった他領域の問題とも深い関係がある。交通とこのような他領域の問題は相互に関係しているが、シミュレータとして相互に補完しあうような枠組みに関する検討は少ない。

様々な応用分野でシミュレーションによるモデルの検証や事象の再現・分析が行われてきているが、それぞれの研究でそれぞれのシミュレータが作成され、作成されたシミュレータの再利用は難しいのが現状である。そこで、本論文では、シミュレータという実行可能な形で蓄積される諸分野の知見を統合することを目標として、異質なモデルを持つ複数のシミュレータを統合するプラットフォームのアーキテクチャを提案する。本研究の目的は、既存のマルチエージェントシミュレータや新規に作成されるマルチエージェントシミュレータを、独立性を保ちながら簡単に統合するための設計方法を提案することである。

我々は、シミュレーションの差異を問題領域と抽象度の観点からとらえる。都市における人々の行動は、関連するそれぞれの領域で解釈される。たとえば、病院への移動という行動は、交通の観点からは車両の移動として、疫病の観点からは病院における感染者と非感染者の接触として解釈される。このように、同じ行動でも問題領域により異なる解釈がされ、モデル化される。また、都市における人々の行動は、異なる抽象度でモデル化される。たとえば、広域交通を扱う場合には、ネットワーク上を物体が移動するような抽象度の高い形でモデル化され、局所交通を扱う場合には、運転者が道路上で観測/操作を行うような抽象度の低い形でモデル化される。このように、同じ行動でも抽象度の異なるモデル化がされる。

本研究では、異質なモデルを扱うシミュレータの統合に焦点をあて、以下の課題に取り組んだ。

(1) シミュレータの独立性

異なる問題領域のシミュレータは、通常はそれぞれの問題領域で使われる。これらを統合することで複数の問題領域を扱うシミュレーションを行う場合、シミュレータの独立性が保たれていることが望ましい。

(2) シミュレータのインタラクションの簡潔性

異なる問題領域のシミュレーションを統合する場合、それぞれの専門分野の専門家はそれぞれのシミュレータのみに精通している。シミュレータ内部の詳細を知らずに、複数のシミュレータを簡潔にインタラクションさせる必要がある。

2. アーキテクチャ

異質な複数のシミュレータを統合するアーキテクチャとして、レイヤ統合アーキテクチャと外部シミュレータ統合アーキテクチャの2つを提案する。

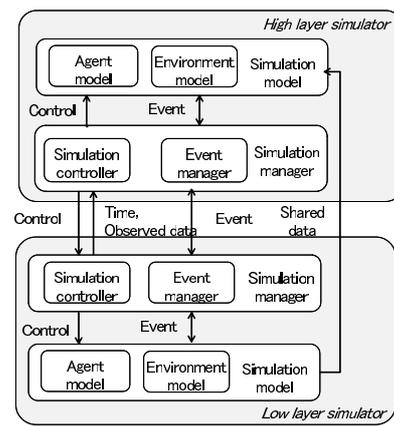


図1 レイヤ統合のアーキテクチャ
Fig. 1 Layer based integration architecture.

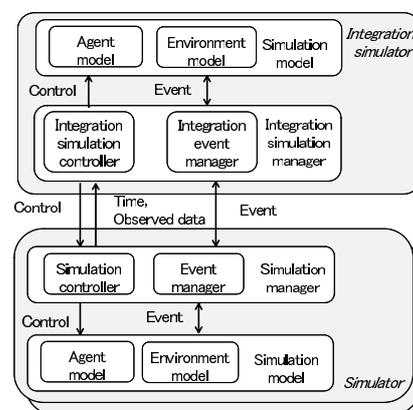


図2 外部シミュレータ統合のアーキテクチャ
Fig. 2 External simulator integration architecture.

2.1 シミュレータ

はじめに、本提案で統合対象とするシミュレータについて述べる。ここで扱われるシミュレータは、通常のシミュレータが備えている初期設定の入力、結果の出力というインタフェースのほかに、(1) 次状態への遷移、(2) 現在時刻の取得、(3) イベントの入出力、(4) イベントの入出力に必要な環境情報の取得というインタフェースを備えているものとする。異なる分野を扱うシミュレータは、それぞれ独立したシステムとして構築される (図1, 図2)。

シミュレーションコントローラ (Simulation controller) は、シミュレーションモデル (Simulation model) を制御する。各シミュレータ (Simulator) は独立して動作するため、それぞれが任意の単位時間で動作することができる。

シミュレーションモデルは、コントローラからの計算依頼を受けると、次状態を計算し、その結果を環境に書き込む。シミュレーションモデルは、人々の行動をモデル化したエージェントモデル (Agent model) と都市環境をモデル化した環境モデル (Environment model) の2つに分けられる。これらのモデルが含むエージェントや施設などのオブジェクトにはIDが割り振られる。

離散事象シミュレーションとして設計される MABS で

は、シミュレーション上で特定の事象が発生するとイベントが発行され、イベントマネージャ (Event manager) により処理される。外部から受けとつても処理可能なイベントを、シミュレータは外部に公開することができる。

2.2 レイヤ統合アーキテクチャ

都市における人々の行動は、異なる抽象度で解釈される。たとえば、地点 A から地点 B への車両の移動を考えると、物体がネットワークの上を流れるような高い抽象度で解釈することもあるし、運転者が道路の形状や周辺車両の状況を観測してハンドルやアクセルの操作を決定し車両を移動させるような低い抽象度で解釈することもある。このような抽象度の異なるシミュレーションの統合を考える。

抽象度の異なるそれぞれのシミュレーションは、抽象度の高い観点からは同一の現象を扱っていることになる。また、同一の現象を扱っていることから、シミュレータの間では、関連の強いデータが存在する。そこで、抽象度の高い上位レイヤ (High layer simulator) から抽象度の低い下位レイヤ (Low layer simulator) を制御し、両シミュレータの接点となるデータを共有するレイヤ状の統合をする (図 1)。上位レイヤのシミュレーションモデルは、より抽象度の低いモデルに置き換えたい状況に関するイベントを公開する。また、その現象に関するデータを共有データ (Shared data) として公開する。

上位レイヤのシミュレーションにおいて、より抽象度の低い計算を必要とする状況となったとき、その状況がイベントマネージャに通知される。それを受けて、上位レイヤのシミュレーションコントローラは、下位レイヤのシミュレーションコントローラを呼び出す。各シミュレータの環境は、基本的にシミュレータごとの独立した環境に保存されるが、抽象度の高いシミュレーションと抽象度の低いシミュレーションの接点となるデータは、両シミュレータの間で共有される。この共有データに対して、下位レイヤのシミュレータは、より抽象度の低いモデルに基づく計算を実施し、結果を書き込む。

2.3 外部シミュレータ統合アーキテクチャ

都市における人々の行動は、関連するそれぞれの領域で解釈される。たとえば、病院への移動という行動は、交通の観点からは車両の移動として、疫病の観点からは病院における感染者と非感染者の接触として解釈される。このような異なる領域の問題を対象とした複数のシミュレータを統合する場合を考える。

対象となる問題領域の異なるシミュレーションの統合では、シミュレーション間の関連は弱く、共有されるデータも少ない。多様なシミュレータとの結合を考えると、シミュレータ間の結合は疎結合であることが望ましい。そこで、外部の統合シミュレータ (Integration simulator) を

使って連携するアーキテクチャとする (図 2)。

外部シミュレータ統合アーキテクチャにおいて、統合シミュレーションコントローラ (Integration simulation controller) は統合される各シミュレータの進捗を調整し、統合イベントマネージャ (Integration event manager) はシミュレータ間のイベントの伝達と変換を行う。

統合シミュレーションコントローラは、各シミュレーションの進捗を調整する。各シミュレータは、統合シミュレータより計算の依頼を受け、次の状態を計算し、計算終了後の現在時刻をコントローラに返す。統合シミュレーションコントローラは、各シミュレータの現在時刻に応じて、次に実行すべきシミュレータを確定する。

統合イベントマネージャは、統合対象となる各シミュレータのイベントマネージャを登録しており、各シミュレータで発生するイベントを取得する。統合イベントマネージャは、取得したイベントの中で必要なものを、それぞれのシミュレータが解釈できる形に変換し、伝達する。これは、シミュレーション間で関係し合う事象と事象を対応づけることに相当する。

各シミュレーションの環境は、シミュレータごとの独立した環境に保存される。その環境は、外部に公開可能な範囲のデータを、読み込みだけ可能な形式で公開する。外部からシミュレーションモデルへ働きかける場合は、イベントの伝達で行う。これにより、シミュレータ間の結合度を下げ、また、外部からの操作によりシミュレーションが破綻することを防ぐ。シミュレータは、公開するイベントの形式や意味に関して明確なインタフェースを外部に提供する。そのインタフェースに従ってイベントを使用することは、イベント送信側シミュレータの責務となる。

2.4 関連研究

これまでに、分散環境下で動作するシミュレーションプラットフォームが提案されている。大規模 MABS 基盤としては、MASON (<http://cs.gmu.edu/~eclab/projects/mason/>)、Repast (<http://repast.sourceforge.net/>)、ZASE [15] があげられる。これらのプラットフォームでは、マルチスレッド処理機能、エージェントデータ管理機能などが備えられており、並列化や分散化に関するシミュレータ開発者の労力を減らすことができる。また、既存のシミュレータを再利用するためのプラットフォームの研究も進められている [12]。このプラットフォームでは、Repast や Ascape (<http://ascape.sourceforge.net/>) で実装したシミュレーションモジュールを分散環境で動作するように統合している。複数の分散したシミュレーションモジュールを統合する研究は、量の面でのシミュレーションの拡張を主眼においている。本論文で提案するプラットフォームの目的は、複数の異質なシミュレータを組み合わせ多様な側面をとらえた都市シミュレーションを構築す

ることであり、これらの既存の基盤とは対象が異なる。

また、ある特定分野の複数のシミュレータを統合する研究として、ロボカップレスキューシミュレータ [9] や統合 ITS シミュレータ [14] に関する研究がある。これらの研究では、防災や交通の分野において実現したい特定のシミュレーションがあり、ドメインに特化したプロトコルを定義することでシミュレータの連携を実現している。それに対して、本研究の提案はドメインを特定しないより抽象度の高い層におけるシミュレータの統合に対するものである。

汎用的にシミュレータを統合する技術として HLA (High Level Architecture) [3] が提案され、IEEE 1516 として採用されている。HLA は分散シミュレータを管理する柔軟な仕様となっており、その規格を満たすシミュレータを RTI (Run-Time Infrastructure) と呼ばれるプラットフォーム上で統合することができる。しかし、HLA は長大であり、その規格を満たすことは容易ではない。また、HLA は、分散オブジェクト管理やオブジェクト所有権管理などを行い、各シミュレータが分散オブジェクトを直接的に操作する設計となっている。これらの性質のため、各シミュレータが多くの操作可能なオブジェクトを公開することになり、シミュレータどうしが密結合しやすくなる。それに対して、提案アーキテクチャは、シミュレータどうしが疎結合になるというメリットがある。しかし、他のシミュレーションに影響を与える際にシミュレーションの介入側はイベントを送信することしかできず、シミュレーションを綿密に組み合わせることはできない。つまり、シミュレータの独立性やシミュレータ統合の容易さの観点からは提案アーキテクチャにメリットがあるが、細かな連携をしようとすると HLA のような形式にメリットがある。

3. 抽象度の異なるシミュレーションの統合

抽象度の異なるシミュレーションの統合として、広域交通における道路ネットワーク移動の簡易的なシミュレーション (Wide-area simple traffic simulator) と局所交通における運転操作の詳細なシミュレーション (Local-area driving behavior simulator) を統合した (図 3)。

広域交通を扱う場合、道路網を単純なリンクと単純なノードからなるネットワークととらえた抽象度の高い環境モデルを扱うことが多い。これは単純な表現方法ではあるが、多数の車両から構成される交通流を分析するには効率的な表現方法であり、これまでの交通工学の研究で使われてきている [2]。しかし、このような抽象度の高い表現は、特定の道路における人間の運転操作を表現するのに向いていない。たとえば、道路幅やレーン数、右折/左折レーンといった詳細な道路構造や、近隣の車両のような周囲の行動主体を表現しにくいからである。そこで、広域の単純な交通行動と局所の詳細な運転行動の関係を扱うシミュレーションを行う場合は、抽象度の異なる 2 つのモデルを組み

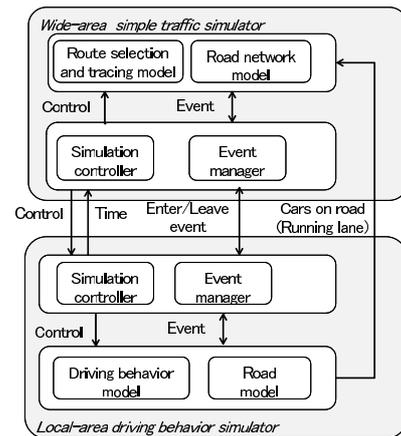


図 3 単純交通シミュレータと詳細交通シミュレータの統合
Fig. 3 Integration wide-area traffic simulator with local-area driving simulator.

合わせる必要がある。

3.1 単純交通シミュレータ

単純交通シミュレータにおいて、交通エージェントは経路計画の選択と経路計画の実行を行う。今回の実装では、道路ネットワークを待ち行列キューを備えたリンクとノードで表現し、その上を交通エージェントが移動する設計とした。単純交通シミュレータの実装には、交通シミュレーションツールキット MATSim (<http://sourceforge.net/projects/matsim/>) を基盤として利用した。

交通センサ情報などからエージェントごとの OD を推定し、エージェントごとに出発地点、経由地点、目的地点を割り当てる。交通エージェントの代表的な振舞いとしては、家を出発し、ショッピングセンタ、会社などを経由して帰宅するというものである。

経路計画選択フェーズにおいて、エージェントは出発地点から目的地点へ到達する経路の選択を行う。エージェントは経路選択のために、シミュレーション環境から道路ネットワークを読み込む。各エージェントは、前日の各道路の通過時間をもとに最短旅行時間となる可能性が高いと予想される経路計画を選択する。

経路計画実行フェーズにおいて、エージェントは決定された経路計画に従って、出発地点から目的地点へ移動する。この移動は、交差点からの道路への進入、道路の移動、道路から交差点への移動として扱われる。移動結果は、道路ネットワーク上での位置情報として環境に保存される。

3.2 詳細交通シミュレータ

詳細交通シミュレータ上のシミュレーションモデルでは、運転者エージェントは道路環境を観測し車両を操作する [10]。詳細交通シミュレータ上の道路には、斜度、曲率、幅など物理的な形状属性が備わっている。各車両の位置は、走行中の道路上での 2 次元座標で表される。道路上の

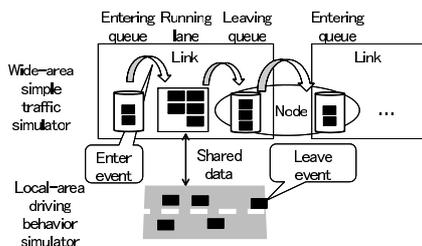


図 4 道路ネットワークと道路における車両の移動

Fig. 4 Behavior of cars on road network and road.

移動は道路ネットワーク上の移動より、短時間で変化する行動であるため、詳細交通シミュレータは単純交通シミュレータより短いシミュレーションステップを備える。

道路上の運転者エージェントは、自身の持つ運転操作モデルに従って、道路環境を観測し、車両を操作する。観測可能な状態は、自車両の状態、周辺車両の状態、道路環境の状態である。自車両の状態として座標、速度、加速度、走行車線などを得ることができる。周辺車両の状態として前後左右の車両の情報を、道路環境の状態として走行中の道路の物理形状を得ることができる。運転者エージェントは、観測した情報を使って意志決定をし、アクセル/ブレーキ/ハンドルの操作を行う。この運転操作に基づき、運転者エージェントの位置が計算される。

3.3 単純交通と詳細交通の統合シミュレーション

単純交通流と詳細運転行動を統合する必要があるのは、局所的な運転行動と広域的な交通流の関係を調べるような場合である。たとえば、筆者らは、運転者の追い抜きに関する選好や希望巡行速度などを含む運転行動モデルと交通流の関係を調べるシミュレーションを実施している [7]。

本研究では、単純交通流シミュレーション上の交通エージェントには、出発地点から目的地までの最短経路探索をするモデルを組み込んだ。また、詳細行動シミュレーション上の運転者エージェントには、「それぞれのエージェントが持つ希望巡行速度より現在速度が遅ければ加速する、前に低速な車両がいるとき追い抜きをする」という運転行動モデルを組み込んだ。詳細運転行動シミュレータと連携させるために単純交通シミュレータに追加した機能は、走行レーンに関するものである。広域の単純交通モデルと局所の詳細運転行動モデルを統合したシミュレーションの進行プロセスを以下に示す (図 4)。

- (1) 単純交通シミュレータにおいて、交通エージェントに 1 日のプランが割り当てられる。
- (2) 単純交通シミュレータにおいて、現在時刻の抽象的な道路ネットワーク上の移動が計算される。交差点の処理に従って、流入キュー (Entering queue) から走行レーン (Running lane)、流出キュー (Leaving queue) から流入キューへ車両を移動させる。
- (3) 走行レーンに進入した車両がいたときは進入イベント

(Enter event) が発生し、単純交通シミュレータは詳細交通シミュレータが呼び出される。単純交通シミュレータは、走行レーン上のエージェントオブジェクトの処理を詳細交通シミュレータに任せる。

- (4) 詳細交通シミュレータは道路上での運転行動を計算する。道路の終端に車両が到達したとき流出イベント (Leave event) が単純交通シミュレータに伝達される。
- (5) 単純交通シミュレータは、道路の終端に到達した走行レーン中の車両を流出キューへ移動させる。
- (6) 単純交通シミュレータの時刻より詳細交通シミュレータの時刻の方が前であるとき、詳細交通シミュレータは時刻を更新し、ステップ (4) へ戻る。
- (7) 1 日の終わりでない場合、単純交通シミュレータは時刻を更新し、ステップ (2) へ戻る。

4. 問題領域の異なるシミュレーションの統合

問題領域の異なるシミュレーションの統合の例として、交通シミュレーションと電力シミュレーションの統合をあげる。これまで都市交通と都市電力という 2 つの問題領域は、市民の観点からは相互に関係がなかった。しかし、電気自動車 (EV) の普及により、都市交通と都市電力に関連が生まれる。たとえば、家庭に高性能な太陽光発電 (PV) パネルが普及すると、設置家庭では電力の余剰が起きる。余剰電力を EV に蓄電し、電力系統へ接続する (V2G: Vehicle-to-grid) ことで、電力の安定供給や、再生可能電力導入の低コスト化への貢献を提言する研究もすでに行われている [8]。

EV の充電という行動は、交通という問題領域においては施設への車両の移動ととらえられ、電力という問題領域においては、電力機器の充電ととらえられる。以下では、このような、問題領域の異なるシミュレーションを統合する場合を考える。

4.1 交通シミュレーション

ここで述べる交通シミュレーションは、3 章と同じものである。交通シミュレータは、到着/出発イベント、目的地変更イベントを公開している。交通エージェントは家を出発し、ショッピングセンタ、会社などを経由して帰宅する。交通エージェントが施設に到着/出発するたびに、到着/出発イベントがイベントマネージャに送信される。

4.2 電力シミュレーション

統合対象のシミュレータとして、施設が持つ電気機器の電力の生成/消費/蓄積/放電に関するモデルを扱う電力シミュレータを新たに作成した。各施設が計算する内容は以下である (図 5)。ステップごとに、太陽光パネルによる発電量、家電による電力消費量が計算される。発電量の方が多かった場合、バッテリーの容量や時間帯ごとの電力の価格

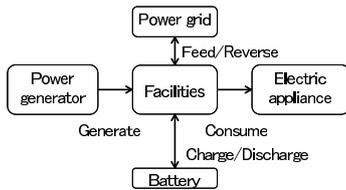


図 5 電力シミュレーションにおける電力の流れ
Fig. 5 Electric flow on electricity simulation.

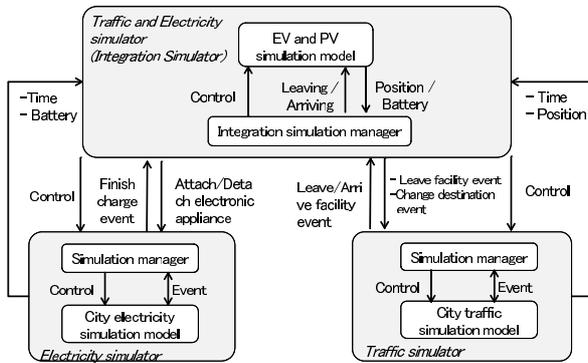


図 6 交通シミュレータと電力シミュレータの統合
Fig. 6 Integration traffic simulator with electronic simulator.

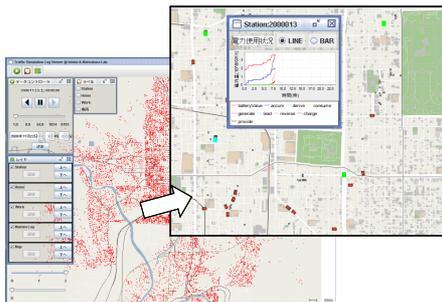


図 7 交通と電力を統合したシミュレーション (左:京都市全域を表示, 右:一部を拡大表示. 車両の位置 (赤い矩形) や施設のバッテリー量を表すグラフを表示)

Fig. 7 Integrated simulation with traffic and electricity simulations.

などを考慮して, 施設は余剰電力を系統電力に戻すかバッテリー (施設設置バッテリーや EV バッテリー) に蓄積するかを選ぶ. 逆に, 消費電力の方が多かった場合, 系統電力から電気を引き込むかバッテリーから電力を引き出すかを選ぶ.

電力シミュレータ上では, 電気機器の接続/離脱のイベントが発生する. このイベントは電力シミュレータの内部で使用されるだけでなく, 外部のシミュレータへも公開される.

4.3 交通と電力の統合シミュレーション

外部統合シミュレータを用いて交通と電力のシミュレーションを統合した. アーキテクチャを図 6 に, シミュレーションの実行画面を図 7 に示す.

統合シミュレーションモデルでは, EV の移動距離を使ってバッテリーの使用量を計算し, EV の充電に関する意志決定を計算する. 統合シミュレータはそれぞれのシミュレー

タ上のオブジェクトを対応づけするモデルとして, 「車両としての電気自動車 ID」と「電力機器としての電気自動車 ID」の対応テーブル, 「出発地/目的地としての施設 ID」と「充電施設としての施設 ID」の対応テーブルを持つ.

統合シミュレーションコントローラは, 交通と電力それぞれのシミュレータの現在時刻を考慮して, それぞれのシミュレータを呼び出す. 呼び出されたシミュレータは, 自身の持つモデルの次状態の計算を行う. 交通と電力という異なる問題領域は, EV の充電施設への到着, 充電施設からの充電および出発という行動をキーにして結びつく. 統合イベントマネージャは, これらの行動に関するイベントの変換と伝達を行う. このシミュレータ間のインタラクションのフローを以下に示す.

- (1) 統合シミュレータ (Integration simulator) は, 交通シミュレータから施設出発のイベント (Leave facility event) を受け取ると, 環境に電気自動車 ID と出発施設を書き込む. EV のバッテリー量が規定量以下であった場合, 近隣の充電施設を検索し, そこへ目的地を変更するイベント (Change destination event) を交通シミュレータへ送る. 統合シミュレータは, 電力シミュレータから対応する ID を持つ EV のバッテリー量を読み出し, 現在の EV のバッテリー量を環境に記録する.
- (2) 統合シミュレータは, 交通シミュレータから施設到着のイベント (Arrive facility event) を受け取ると, 出発地点と施設の距離を計算し, 消費した電力量を計算する. その結果は統合シミュレーション環境に記録される. EV のバッテリー量が規定量以下であった場合, 充電量を含めて施設電源への接続イベント (Attach electronic appliance event) を電力シミュレータへ送る.
- (3) 電力シミュレータにおいて, EV の充電が終わると施設電源からの離脱が行われ, 充電完了イベント (Finish charge event) が送信される. 統合シミュレータは充電完了イベントを受け取り, 該当する EV が充電専用施設に停車していた場合は施設出発のイベント (Leave facility event) を交通シミュレーションに送信する.

PV と EV を同時に扱うシミュレーションをするには, 電力と交通の両方の分野にまたがるシミュレーションをする必要があるが, 両分野の質の高いシミュレータを作成することは難しい. たとえば, 今回用いた MATSim 0.1.1 は, 数万行の Java コードからなり, これと同等の交通シミュレータを実装することはコストが大きい. 提案アプローチのように, 既存のシミュレータを統合して利用できるようにすることで, 新しい社会現象を扱うシミュレーションの構築が容易になる. 電力シミュレータと連携させるために交通シミュレータに修正を加えた点は, 充電と関連した目的地変更イベントの処理である.

5. 実行例と評価

5.1 レイヤ統合アーキテクチャの性能評価

本プラットフォームの性能評価のため、交通シミュレーションに関して抽象度の異なるシミュレータをレイヤ型で統合する前と統合した後で、エージェント数とシミュレーション計算時間の関係を調べた。

このシミュレーションでは、京都市の約 20 km 四方の領域を実験の対象として選び、ゼンリン社の数値地図から約 5 万のリンクと約 10 万のノードを持つ道路ネットワークを生成した。このネットワーク上で、ランダムに OD (Origin-Destination) を生成し、広域の単純交通シミュレーション上の交通エージェントに割り当てた。広域の単純交通シミュレーションのステップは 1 秒、局所の詳細交通シミュレーションのステップは 0.5 秒とした。シミュレーション内時間で 24 時間分の計算を行った。

実験には Corei7 3.02 GHz の CPU, 12 GB のメモリを持つ計算機を用いた。Java VM は Sun JRE 1.6.0.23 (64 bit), MATSim はバージョン 0.1.1 を用いた。また、各シミュレータは、同一 Java VM 上で動作させた。

図 8 に、エージェント数とシミュレーション計算時間の関係を示した。このグラフを見ると、統合前のシミュレーションと統合後シミュレーションの双方において、エージェント数に対して計算時間が線形に増加していることが分かる。また、その差は一定比となっている。

5.2 外部シミュレータ統合アーキテクチャの性能評価

本プラットフォームの性能評価のため、交通と電力という問題領域の異なるシミュレータを外部シミュレータにより統合する前と統合した後で、エージェント数とシミュレーション計算時間の関係を調べた。

交通エージェントの全エージェントを EV 所持者とした。交通エージェントは家庭から勤務地に移動し、家庭に戻ってくる。その間で充電が必要になった場合、充電専用施設に立ち寄る。施設は家庭、勤務地、充電専用施設だけとした。家庭における電力の発生、消費、充放電のシミュレーションは、1 ステップを 10 秒とした。交通シミュレーションに関する設定や計算機環境は、5.1 節と同じである。

図 9 に、エージェント数とシミュレーション計算時間の関係を示した。このグラフでは、統合前の各シミュレーションと統合後のシミュレーションのそれぞれにおいて、エージェント数に対して計算時間が線形に増加することが示されている。また、その差は一定比となっている。シミュレーション統合のオーバーヘッドは、外部と送受信するイベントの処理により生じる。エージェントの意志決定や環境の次状態の計算コストに比べて、イベント処理の計算コストが相対的に非常に小さければ、オーバーヘッドは無視できる。今回の統合シミュレーションでは、エージェント

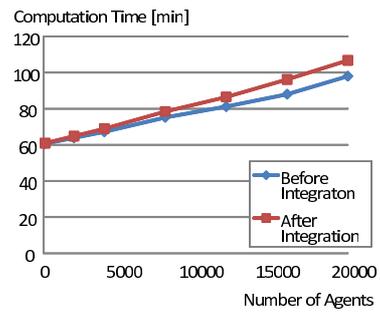


図 8 レイヤ統合アーキテクチャにおけるエージェント数と計算時間の関係

Fig. 8 Performance of layer based integration architecture.

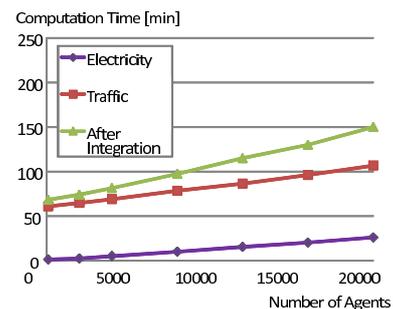


図 9 外部シミュレータ統合アーキテクチャにおけるエージェント数と計算時間の関係

Fig. 9 Performance of external simulator integration architecture.

の意志決定は外部イベント処理に比べて複雑であり、また頻度も高かったため、統合のオーバーヘッドは小さかった。しかし、シミュレータ間でやりとりされるイベントが多いような場合やシミュレータが別の計算機上で動いていて通信にコストがかかるような場合などではシミュレータ連携のオーバーヘッドは増大することになる。

6. 考察

抽象度の異なるシミュレータの連携では、抽象度の高いシミュレータが抽象度の低いシミュレータを呼び出す関係になる。抽象度が異なるだけで同一の現象を扱っているシミュレータを統合する際には、両シミュレータで扱う一部のデータに強い関係が生じる。そこで、上位レイヤから下位レイヤを呼び出し、一部のデータを共有するレイヤ状の統合が適切である。この統合方式では、共有する情報を綿密に制御できるが、シミュレータ間の結合度は高くなる。

問題領域の異なるシミュレーションの統合では、対象や場所の異なるシミュレーションが統合される。この場合、シミュレーションのデータや進行プロセスはそのシミュレータに強く結びつき、シミュレータ間には比較的弱い関係しか生じない。そこで、シミュレーション間の作用をイベントに限定した統合が適切である。この場合、シミュレーション間の作用を詳細に制御することはできないが、シミュレータ間の結合度は低くなる。

複数のシミュレータを統合する場合、シミュレータ上の全オブジェクトを公開し、グローバルな共有環境の下で統合する方法も考えられる。しかし、この方法ではシミュレータ間の結合度が高くなってしまい、統合したシミュレーションを再活用することは難しい。提案するアーキテクチャにおいて、統合されたシミュレータも1つのシミュレータと見なすことができるので、統合されたシミュレータ自体も統合対象となりうる。

7. あとがき

MABS は都市交通など複雑な社会現象を分析する有力な技術であり、これまでに、交通、防災、社会科学など様々な分野で適用されている。これらが対象としている現象は相互に関係があるが、シミュレータとしては相互に補完しあうような枠組みがないのが現状である。本論文では、シミュレータという実行可能な形で蓄積されてきた諸分野の知見を統合することを目標として、異質なモデルを持つ複数のシミュレータを統合するプラットフォームのアーキテクチャを提案し、以下の2つの課題を解決した。

(1) シミュレータの独立性

複数のシミュレータをそれぞれ独立したシステムとして扱い、イベントを基本とした統合をすることで、各シミュレータの独立性を維持した。

(2) シミュレータのインタラクションの簡潔性

シミュレータを連携させる際の接点となるイベントとデータのみを公開させることで、シミュレータ間のインタラクションを簡潔にした。

謝辞 本研究はパナソニック・京都大学共同研究「個の行動・嗜好に基づく群行動のナビ制御による地域 EMS の実現」、JSPS 若手研究 (B) 「マルチモデルに基づく都市シミュレーションプラットフォーム」の助成を受けた。

参考文献

[1] Balmer, M., Cetin, N., Nagel, K. and Raney, B.: Towards Truly Agent-Based Traffic and Mobility Simulations, *Proc. 3rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2004)*, pp.60-67 (2004).

[2] Balmer, M., Meister, K., Rieser, M., Nagel, K. and Axhausen, K.W.: Agent-based Simulation of Travel Demand: Structure and Computational Performance of MATSim-T, *Proc. 2nd TRB Conference on Innovations in Travel Modeling*, pp.1-30 (2008).

[3] Dahmann, J. and Morse, K.: High level architecture for simulation: An update, *Proc. 2nd International Workshop on Distributed Interactive Simulation and Real-Time Applications*, pp.32-40, IEEE (1998).

[4] Deguchi, H., Kanatani, Y., Kaneda, T., Koyama, Y., Ichikawa, M. and Tanuma, H.: Social Simulation Design for Pandemic Protection, *Proc. 1st World Congress on Social Simulation*, Vol.1, pp.21-28 (2006).

[5] Epstein, J. and Axtell, R.: *Growing Artificial Societies: Social Science from the Bottom Up*, MIT Press (1996).

[6] Halle, S. and Chaib-draa, B.: A Collaborative Driving System based on Multiagent Modelling and Simulations, *Journal of Transportation Research Part C*, Vol.13, pp.320-345 (2005).

[7] Hattori, H., Nakajima, Y. and Yamane, S.: Massive Multiagent-Based Urban Traffic Simulation with Fine-Grained Behavior Models, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol.15, No.2, pp.233-239 (2011).

[8] Kempton, W. and Tomic, J.: Vehicle-to-grid power fundamentals: Calculating capacity and net revenue, *Journal of Power Sources*, Vol.144, No.1, pp.268-279 (2005).

[9] Kitano, H., Tadokoro, S., Noda, H., Matsubara, I., Takahashi, T., Shinjou, A. and Shimada, S.: RoboCup rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research, *Proc. IEEE Conference on Systems, Men, and Cybernetics*, Vol.6, pp.739-743 (1999).

[10] Nakajima, Y., Yamane, S. and Hattori, H.: Multi-model Based Simulation Platform for Urban Traffic Simulation, *Lecture Notes in Artificial Intelligence, LNAI 7057*, pp.228-241, Springer (2010).

[11] Paruchuri, P., Pullalarevu, A.R. and Karlapalem, K.: Multi Agent Simulation of Unorganized Traffic, *Proc. 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2002)*, pp.176-183 (2002).

[12] Scerri, D., Hickmott, S., Padgham, L. and Drogoul, A.: An Architecture for Modular Distributed Simulation with Agent-Based Models, *Proc. 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2010)*, pp.541-548 (2010).

[13] Yamashita, T., Izumi, K., Kurumatani, K. and Nakashima, H.: Smooth traffic flow with a cooperative car navigation system, *Proc. 4th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2005)*, pp.478-485 (2005).

[14] 伊川雅彦, 後藤幸夫, 熊澤宏之, 古澤春樹: 異種分散シミュレーションによる ITS 開発環境の構築, *情報処理学会論文誌*, Vol.45, No.12, pp.2805-2814 (2004).

[15] 山本 学, 田井秀樹, 水田秀行: 1 億エージェントを用いたエージェントベースシミュレーションの実現への考察, *電子情報通信学会論文誌 D*, Vol.90, No.9, pp.2423-2431 (2007).

中島 悠

平成 21 年京都大学大学院情報学博士課程修了。博士 (情報学)。日本学術振興会特別研究員 (DC1), 同大学特定研究員を経て, 同大学特定助教。都市上の人間行動のマルチエージェントシミュレーションに興味を持つ。

服部 宏充 (正会員)

平成 16 年名古屋工業大学大学院博士課程修了。博士 (工学)。平成 16 年学振特別研究員 (PD)。リバプール大学, マサチューセッツ工科大学客員研究員を経て, 平成 19 年京都大学大学院情報学研究科助教。マルチエージェントシミュレーションに興味を持つ。