

Python製オープンソース交通流シミュレータ UXsim とその応用例

瀬尾 亨[†]
東京工業大学[†]

1 はじめに

自動車交通は今日の都市で重要な役割を果たしているが、渋滞などの問題を抱えている。また、近年はシェアリング、自動運転、電気自動車など革新的な交通技術が登場し、そのような車両を大量に集めて都市規模で活用するための研究の必要性も高まっている。そのような研究のための一手段として、交通流シミュレーションが有用である。

交通流シミュレーションは一般にミクロとマクロの2種に分類される。ミクロは、車両1台1台の挙動を詳細に再現するものである。マクロは、より巨視的な現象、例えば渋滞量の再現に重きを置いたものである。

都市規模の交通の特色は、大規模かつ動的な現象である点である。そのような現象を詳細なミクロシミュレーションで再現するのはデータ準備および計算に多大な労力と費用がかかるが、大局的な交通の評価のためには必ずしもミクロを用いる必要はない。例えば、村田ら¹⁾は、道路課金の決定手法を検証するためにミクロとマクロの交通流シミュレーションを用いた結果、大局的には類似した結果が得られたと報告している。大口ら²⁾は、首都圏の基本道路全てを対象としたマクロ・メソ交通シミュレーションを開発し、交通マネジメントの評価に用いている。Seo and Asakura³⁾は、マクロモデルを活用した都市規模のシェア型自動運転システムのための最適化手法を提案している。

研究に際しては柔軟に使用・改変可能なオープンソースのシミュレータが有用である。例えば、ミクロ交通流シミュレータとしてはSUMO^{*1}が有名である。交通制御用APIも整備されており、近年の自動運転制御などへの応用が多い。また、MATSim^{*2}は交通需要分析に適したシミュレータであり、動的な交通も考慮している。しかし、交通目的に特化したマクロなシミュレータは限られている。

本論文では、著者が新たに開発・公開した純Python製のオープンソース交通流シミュレータ UXsim^{*3}を紹介する。純Python製であるため、使用と理解が容易で、かつ汎用的な機械学習用フレームワークなどの直接の組み込みが可能である。Pythonは一般に計算効率が低いが、マクロシミュレータとすることでその欠点を補っている。そして、大規模な交通分析への応用に資することを意図している。

その交通流シミュレータとしての長所は以下である：

- 動的な交通・渋滞現象を表現できる
 - ネットワークでの経路選択行動を表現できる
 - 個別の車両の移動軌跡を計算できる
 - 純Python製でありPythonコーディングにより柔軟に操作でき、自動化や外部ライブラリの組み込みも容易
 - 上記性質をもつシミュレータとしては計算効率が高い
 - コード行数2300程度（そのうち、可視化などを除いた本質的な部分は約1500）であり、理解や改変が容易
- 一方、現時点で既知の短所は以下である：

- Python製であり一般的な計算効率は低い
- ミクロ交通流シミュレータほど精細ではない
- UIは未実装であり、コーディングにより操作する必要

2 モデル

UXsimは、以下の3モデルを組み合わせてネットワーク交通流を計算する：

- (1) メソスケールの車両追従モデル (KWモデルと等価)
- (2) ノードでの分岐合流モデル
- (3) 各時間帯で瞬間的な最小旅行コスト経路を選択する経路選択モデル

これらのモデルやその統合方法の詳細については文献⁴⁾を参照されたい。

UXsimの入力はネットワーク構造、時間帯別OD需要、各種モデルパラメータである。出力は各車両（車群）の移動軌跡とネットワーク交通状態などである。

本シミュレータが用いる交通流モデル(1)の特筆すべき点は、マクロモデルと等価であるのにも関わらず個別車両の軌跡を明示的に表現できることである。一般的なマクロ交通流モデルであるKWモデル⁴⁾⁵⁾は下式で表される：

$$\frac{\partial k(t, x)}{\partial t} + \frac{\partial Q(k(t, x))}{\partial x} = 0$$

ここに、 $k(t, x)$ は時刻 t 、地点 x の車両密度、 $Q(k)$ は密度が k のときの交通流率を求める流率密度関係であり、この式自体は交通流保存則を意味している。このように、本モデルは交通流を流体と近似し連続的な密度と流率の分布を考えるモデルであるが、 Q が特定の関数形の場合には以下のミクロモデルと等価であることが知られている⁴⁾⁵⁾：

$$X(t + \Delta t, n) = \min \left\{ \begin{array}{l} X(t, n) + u\Delta t, \\ X(t + \Delta t - \tau\Delta n, n - \Delta n) - \delta\Delta n \end{array} \right\}$$

ここに、 $X(t, n)$ は車両 n の時刻 t の位置、 Δt はシミュレーションのタイムステップ幅、 Δn は車群サイズ、 u は最大速度、 τ はドライバー反応時間、 δ は車両サイズである。本モデルは特定車両の位置を逐次的に計算するミク

[†] Toru Seo, Tokyo Institute of Technology

^{*1} <https://sumo.dlr.de/docs/index.html>

^{*2} <https://www.matsim.org/>

^{*3} <https://github.com/toruseo/UXsim>

ロモデルである。右辺第一項は、車両 n が最大速度 u で自由走行する場合を意味する。第二項は、前方車両に近すぎた場合に一定の時間幅 τ ・空間幅 δ をおくように追従走行する場合を意味する。この場合、状況によっては車両 n は減速し、渋滞が発生する。

本モデルは必ずしも全ての車両の軌跡を計算する必要は無く、 Δn を1より大きい値に設定すれば複数の車両をまとめて車群として扱う計算が可能である。その計算方法を取った場合、一般に計算量は $1/\Delta n^2$ に比例し、かつ得られた解（リンク内交通状態）は離散化誤差の範囲内で解析解と一致するため、非常に効率的かつ正確なシミュレーションが可能となる。

3 応用例：深層強化学習による信号制御

UXsim の挙動を確認するため、これが Python 製であることを活かして、代表的な深層学習用ライブラリである PyTorch^{*4} を組み込んだ深層強化学習による信号制御への応用例^{*5}を示す。

交通状況・シミュレーションシナリオは以下の設定とした。4つの交差点が格子状に配置された井型ネットワーク（図1）に、全ての方向から全ての方向に向けて平均して同一の量の交通需要が毎回ランダムに発生するものとする。具体的な仕様は、時間長さ1時間、リンク数24、ノード数12、道路総延長12km、車両台数約8000台、車群サイズ5台、タイムステップ幅5秒である。

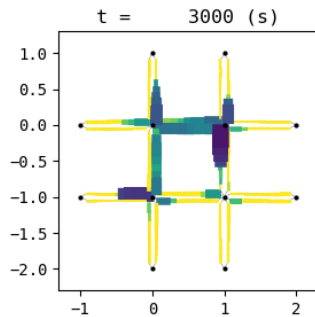


図1: ネットワーク交通状況：線が太いほど車両密度が高く、色が濃いほど平均速度が遅い

まず、UXsim によってシミュレートされた交通流の様子を示す。図1はある時刻の直近2分間平均のネットワーク全体の交通状況である。各交差点で赤信号による待ち行列が発生しており、左上交差点の下方向では青信号でも渋滞のため前に進めないキョースピルオーバー現象が起きている。図2は信号を含む路線の5台毎の車両軌跡である。赤信号により待ち行列が発生・延伸し、青信号になると車両が進んでいる様子がわかる。一部では車両が赤信号で2回待たされる過飽和現象が生じている。この1時間のシミュレーションの計算時間は概ね1-2秒であった。

^{*4} <https://pytorch.org/>

^{*5} Jupyter Notebook 版：https://github.com/toruseo/UXsim/blob/main/demos_and_examples/demo_notebook_03en_pytorch.ipynb

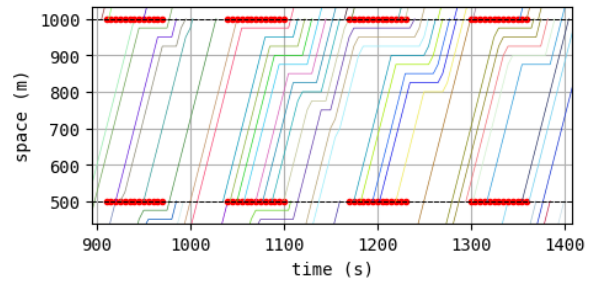


図2: 車両軌跡：横向きの赤太線は赤現示を意味する

次に、深層強化学習による信号制御について述べる。まず、この交通状況は素朴な定周期制御では1時間以内にグリッドロックに陥ってしまう。そこで、状態を各リンクの待ち台数（16次元）、報酬を総待ち台数の変化量、行動を4信号の現示組合せ選択（16選択枝）とし、10秒毎に行動を選択する Deep Q Learning により信号制御を改善する。中間層は4層で各64ニューロンとした。学習を200エピソード実行した様子を図3に示す。ここで、縦軸はトリップ終了車両の平均遅れ時間である。学習の進展とともに遅れ時間が減少し、100エピソード以降は外れ値はあるものの概ね一定の値に収束していることがわかる。200エピソードの学習に要した総計算時間は約13分であった。1エピソードのうち交通シミュレーションに要する時間は1-2秒であったことを考えると、UXsim の計算効率はこの応用のためには良いといえる。計算機環境はCPU3.8GHz*16, GeForce RTX 2060, 32GB RAM であった。

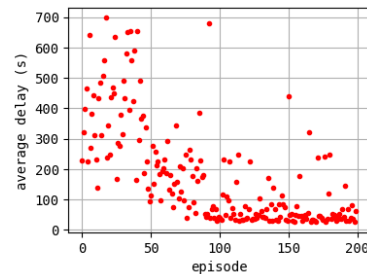


図3: 深層強化学習の進展の様子

参考文献

- 1) 村田航希, 山本規吉, 川上朋也: 需要分布に基づくロードプライシングにおける計算量削減とミクロ交通シミュレーションによる評価, 情報処理学会論文誌, Vol. 63, No. 2, pp. 655-664, 2022.
- 2) 大口敬, 力石真, 飯島護久, 岡英紀, 堀口良太, 田名部淳, 毛利雄一: 首都圏3環状高速道路における交通マネジメント評価シミュレーションの開発, 土木学会論文集D3 (土木計画学), Vol. 74, No. 5, pp. I_1255-I_1263, 2018.
- 3) Seo, T. and Asakura, Y.: Multi-objective linear optimization problem for strategic planning of shared autonomous vehicle operation and infrastructure design, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, pp. 3816-3828, 2022.
- 4) 瀬尾亨: マクロ交通シミュレーション: 数学的基礎理論とPythonによる実装, コロナ社, 2023.
- 5) Laval, J. A. and Leclercq, L.: The Hamilton-Jacobi partial differential equation and the three representations of traffic flow, *Transportation Research Part B: Methodological*, Vol. 52, pp. 17-30, 2013.