

大規模道路ネットワークシミュレータDEBNetSの改良と検証^{*1}

*Improvement and Verification of Dynamic Traffic Flow Simulator "DEBNetS" ^{*1}*

菊池輝^{*2}, 加藤義昭^{*3}, 馬淵透^{*3}, 藤井聡^{*4}, 北村隆一^{*5}

By Akira KIKUCHI^{*2}, Yoshiaki KATO^{*3}, Toru MABUCHI^{*3}, Satoshi FUJII^{*4}, Ryuichi KITAMURA^{*5}

1. はじめに

近年のコンピュータ演算処理能力の著しい向上により、シミュレーションによる交通需要分析も実用の範疇に入り、時間軸上における個人の生活行動の再現や交通流の再現を図る研究が盛んに行われている。交通需要解析におけるシミュレーションの利点は、時間軸を明示的に組み込むことが可能であるほか、交通行動の背後にある複雑な意思決定過程を考慮できる等の点で、旧来の集計的需要予測手法の限界・課題を克服する手法と言える¹⁾。また近年では、従来の集計的あるいは非集計的予測手法に対するシミュレーション手法の代替可能性が議論されるとともに、現在ではシミュレーションでしか評価しえない政策や評価指標が増えつつあることも指摘されている²⁾。

これまでの研究で、交通の発生・分布・分担を担う生活行動（交通行動）シミュレーションと、配分を担う交通流シミュレーションが開発されてきた。これら2つのシミュレーションモデルが取り扱う交通現象は、相互に影響を及ぼしあうため、近年注目を浴びているTDM等の政策を適切に評価するためには、この相互作用を考慮したモデルシステムが必要となる。

DEBNetS (Dynamic Event-Based Network Simulator) は、上述のような、生活行動シミュレータに基づいた総合的な需要解析のためには道路網上での自動車トリップの干渉現象に伴う移動時間の増加（いわゆる混雑現象）を考慮することが不可欠であるとの認識のもとで開発された交通流シミュレータである。

DEBNetSの特徴は、大都市圏の一般街路を含めた大規模なネットワークを、計算対象とすることが可能な点であり、これまでに京都市³⁾や大阪市⁴⁾を対象とした評価計算に適用されている。大阪市の事例では、3000リンク以上のネットワークを対象としている。DEBNetSのモデル上の特徴は、イベントスキニング法を採用することで時間経過を連続的に表現している点である。それ故、生活行動シミュレータで計算される一人一人の生活行動における自動車トリップの出発時刻をデータとして読み込むことで、一人一人の到着時刻を連続時間軸上で計算することが可能となる。

このDEBNetSと生活行動シミュレータの統合システムを用いた交通需要予測の適用事例の結果より、DEBNetSに関する以下の課題点が明らかとなった。

① 渋滞の延伸現象を再現する必要がある

② 計算時間を短縮する必要がある

前者に関して、DEBNetSはリンク上に存在可能な車両台数についての制約条件を考慮しておらず、渋滞が上流に延伸する現象を再現することができない。ネットワーク上の交通流を忠実に再現するにあたり、渋滞延伸現象の再現はDEBNetSの大きな課題である。また後者の計算時間に関して、都市圏レベルを計算対象とした場合、24時間のシミュレーションの実行にPCで20時間弱を要している。アルゴリズムの改良による計算コストの削減も課題として挙げられる。

本研究では、上述の課題点への対応を図るとともに、交通流シミュレーションモデルとしての検証（verificationおよびvalidation）を行う。

2. DEBNetSの改良

(1) 改良前のDEBNetSの概要

DEBNetS は、道路網を構成するリンクを複数の

*1 キーワーズ：ネットワーク交通流、配分交通、シミュレーション

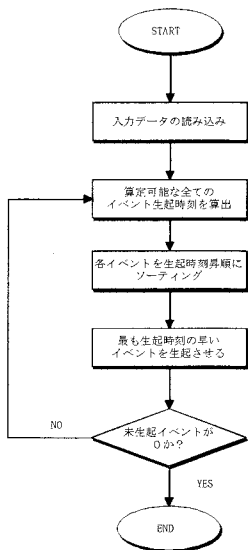
*2 正員、工博、京都大学工学研究科土木システム工学専攻

*3 非会員、オムロン株式会社 PITS 統轄事業部

*4 正員、工博、京都大学工学研究科土木システム工学専攻

*5 正員、Ph.D. 京都大学工学研究科土木システム工学専攻
(京都市左京区吉田本町、TEL075-753-5136, FAX075-753-5916)

道路セグメントに分割した上で、各車両あるいは車両群（以下、これをパケットと呼称する）の道路網上の挙動を、道路網に関する情報と自動車交通需要に関するデータに基づいて、以下のイベントスキニング法により、シミュレーション内で時間を更新させつつ、道路網上の交通状態の変化を再現している：1）各パケットの発生・到着、および、セグメント間の移動をイベントとして定義、2）各イベントの生起時刻に基づきシミュレーション時刻（VT）に最も近い未処理イベントを検出、3）そのイベントに対応した道路セグメント内の車両台数を更新する一方で、VT をそのイベントの生起時刻に更新する。なお、各パケットの道路セグメント内の挙動は、道路セグメント内の交通状態が一様であると見なした上で、KV 曲線を用いてその所要時間を決定することにより再現している。また簡便に経路選択行動を再現するために、各運転者はリンクから流出するたびに、経路選択を行い、その時点での目的地までの最短時間経路に流入するものとしている。このとき、各ドライバーが保有する旅行時間情報は全てのドライバーについて共通であるとして計算コストの軽減を図っている。旅行時間は、一定時間間隔毎にその時点での交通状態から算定している。図－1に DEBNetS の計算手順を示す。



図－1 改良前 DEBNetS の計算手順

(2) 渋滞の延伸現象の再現

各リンクの最下流セグメントの更に下流側に「流出セグメント（長さ 0）」を新たに付加し、このセグメント内で信号制御および渋滞列形成を処理することにする。流出セグメントに入ってきた車両は、まず渋滞が発生している（＝流出セグメントに 1 台以上存在する）かどうか判定（渋滞判定）し、発生していれば渋滞列最後尾に加わる。また渋滞が発生していなければ、次に、下流側リンクが受入可能（＝先詰まりがない）であるかどうかを判定し（受入判定）、不可能であれば当該リンクの流出セグメントに渋滞列を形成する。受入可能であれば、次に当該リンクの流出容量を超過していない（ここでは予め断面容量と青時間比等から 1 信号サイクル内に流出できる台数が決定されている）かどうかを判定（流出判定）し、超過している場合は渋滞列に加わり、超過していなければ次のリンクへ移動する。

ここで、渋滞列を形成あるいは渋滞列に加わることを「渋滞イベント」、次のリンクに移動することを「流出イベント」と呼ぶ。また流出セグメント内へ移動するイベントを「交差点流入イベント」と定義し、従来の DEBNetS で設定したイベントと合わせ総イベント数は 6 種類となる。

信号の影響に関しては、流出イベントを行う全ての車両に平均待ち時間を付加することで表現する。なお、この方法で形成される渋滞列は Horizontal Queue ではなく、Vertical Queue であり、流出セグメント以外の道路セグメントにおける KV 曲線は自由流、すなわち渋滞を考慮しないものを用いる。

交差点付近における処理の手順を図－2に示す。図－2中の流出予定時刻算出式は、その車両の状態によって以下のように異なる。

① 渋滞判定により渋滞を形成する場合：

$$\text{流出予定時刻} = \text{現在時刻} + \text{渋滞長} / \text{流出容量}$$

② 受入判定により渋滞列を形成する場合：

$$\text{流出予定時刻} = \text{現在時刻} + \text{一定時間}$$

③ 流出判定により渋滞列を形成する場合：

$$\text{流出予定時刻} = \text{次の信号サイクル開始時刻} + \text{渋滞長} / \text{流出容量}$$

なお、②での一定時間を用いているのは、下流側セグメントが受入可能となる時刻を予測出来ないためである。今後は、下流側の交通状態を加味してこれ

を予測する等の改良が必要であろう。

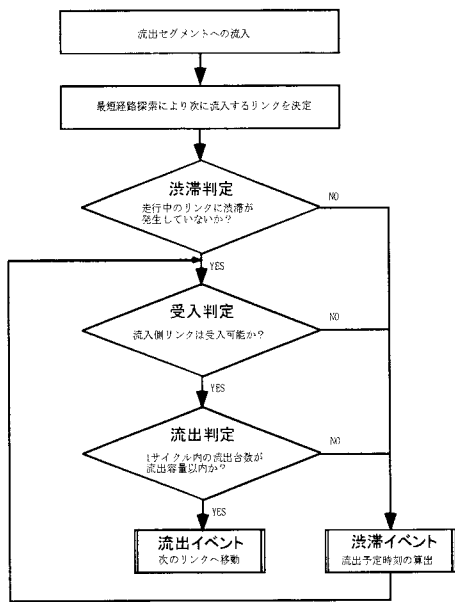


図-2 交差点付近における処理の手順

(3) 計算時間の短縮のためのアルゴリズム改善

従来のDEBNetSの時間進行はイベントスキャンニング方式であった。そのため車両データは常にイベント生起時刻の昇順にソートされている必要がある。すなわち何らかのイベント生起後に次のイベント生起時刻を算出し、それをもとに全車両を生起時刻順にソーティングしなくてはならない。全体の車両数が少ない場合はあまり問題とならないが、車両数が多い場合、このソーティングの実行時間は不安定で、シミュレーション全体に大きな影響を及ぼすと思われる。そこで、イベントスキャンニング方式にスキャンニング・インターバル（離散化した時間帯）を導入し、このスキャンニング・インターバルごとに車両データを管理することにする。そしてこの管理された車両のイベントを順次生起させ、生起後にはその車両の次のイベント生起時刻を算定し、スキャンニング・インターバルとの管理関係を更新する。

例えば、図-3で示すように、シミュレーション時刻が S1～S2 の間にイベントが発生する車両は 3 台 (T1,T2,T3)、同様に S2～S3 の間には 1 台 (T4)、S3～S4 の間には 2 台 (T5,T6)、S4～(S5) の間にイベントが発生する車両は 2 台 (T7,T8) となっているときに、T1、T2、T3 の車両のイベント

を順次生起させ、各車両の次のイベント生起時刻を算定し、その生起時刻を含むスキャンニング・インターバルを求める。たとえば T1 の車両の次のイベント生起時刻が、S3～S4 に含まれるとすると、その車両データは、S1 のカテゴリーから解放され、S3 のカテゴリーに管理されることになる。

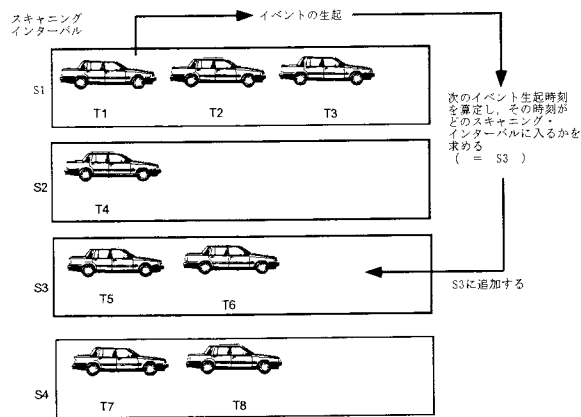


図-3 改良後の車両データの管理方法

このようにして、ソーティングの時間負荷を軽減することで、通常のイベントスキャンニング方式、および、通常のピリオディックスキャンニング方式よりも、計算効率の良い処理が可能となる。

(4) 改良効果の確認

a) 渋滞の延伸現象の再現

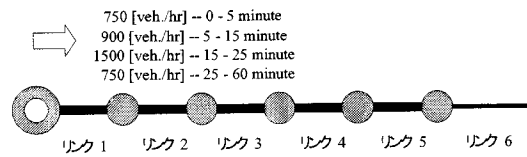


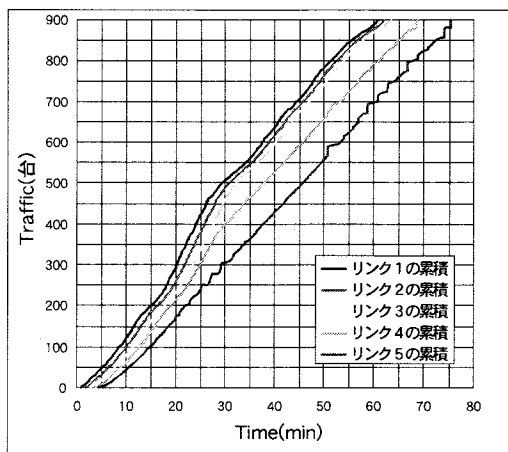
図-4 渋滞延伸現象の再現確認のためのネットワーク

交通流シミュレーションモデルの標準検証プロセス⁴⁾に基づいて、渋滞の延伸が適切に表現できているか否かを確認した。図-4のような、容量2,200[台/時]の6本のリンクが直列につながった単路状のネットワークにおいて、最下流のリンクがボトルネックになるよう設定し、図中に示されるような時間ごとに変化する需要を与えたときの各リンクの流入交通量を観測した。ボトルネック容量は800[台/時]に設定している。表-1にネットワークの設定を示す。

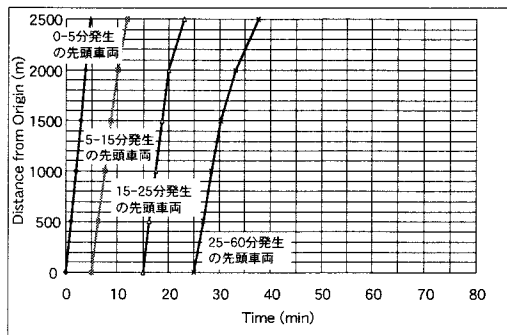
表－1 ネットワークの設定

	リンク長	リンク容量
リンク1～5	500m	2,200[台/時]
リンク6		800[台/時]

図－5に各リンクの流入累積交通量を、図－6に各時間帯（0-5,5-15,15-25,25-60min）に発生した先頭車両の位置を示す。これらの図より、ボトルネックで発生した渋滞が上流側リンクに延伸していくことが見てとれる。



図－5 各リンクの流入累積交通量



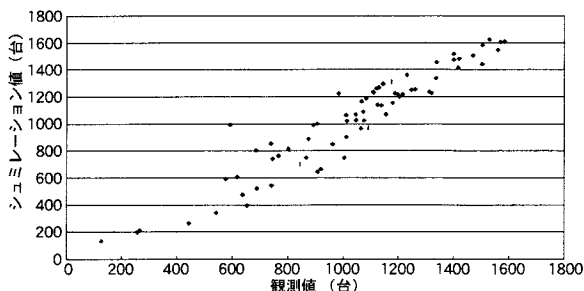
図－6 先頭車両の軌跡

b) 計算時間の短縮

従来の DEBNetS の事例研究として報告されている大阪市への適用事例⁵⁾（リンク数：2994，ノード数：1050，セントロイド数：292，車両データ：990,575 トリップ）では計算時間が 17 時間 31 分^[注1]であったが，これと同一の条件で，上記の改良を施した後の計算時間は 2 時間 25 分^[注2]となり，約 7 分の 1 に短縮し改善効果が確認できた。

3. 実ネットワークを用いた検証

実データを用いたシミュレーションモデルの検証（Validation）のために公開されている吉祥寺BMデータ⁶⁾を用いて，DEBNetSを実行し，各リンクの交通量を実データと比較したところ，相関係数が 0.95 となり，概ね現況を再現できていることが確認された（図－7）。10 分間交通量に基づいた時間帯別の相関係数についても，リンク交通量がない状態から計算を開始したために 7 時 50 分台では低い，それ以降では約 0.83 と良好であった。



図－7 実ネットワークを用いたリンク交通量の検証

4. おわりに

本研究では，交通流シミュレータ DEBNetS の課題を整理し，その対処を図った。さらにその効果を確認するために仮想ネットワークおよび実ネットワークにおける検証を行った。

注

[注1] PentiumIII 800MHz/2CPUのマシン，SunOS5.8上で実行，最短経路探索の時間間隔は15分。

[注2]実行環境は注1と同じ，最短経路探索の時間間隔は5分。

参考文献

- 1)北村隆一：交通需要予測の課題－次世代手法の構築にむけて，土木学会論文集，No. 530/IV-30，pp. 17-30，1996。
- 2)藤井聡：交通計画におけるシミュレーション手法の適用可能性，土木計画学研究・論文集，No.16，pp. 19-34，1999。
- 3)藤井聡，菊池輝，北村隆一：マイクロシミュレーションによる CO₂ 排出量削減に向けた交通施策の検討：京都市の事例，交通工学，Vol. 35，No. 4，pp. 11-18，2000。
- 4)道路交通シミュレーションシステムクリアリングハウス：URL: <http://trans1.ce.it-chiba.ac.jp/ClearingHouse/>
- 5)菊池輝，藤井聡，白水靖郎，北村隆一：大規模ネットワークにおける交通流シミュレータ DEBNetS の現況再現について，第 20 回交通工学研究発表会論文報告集，pp. 49-52，2000。