渋滞進展を考慮した時間帯別交通均衡配分の効率的解法 － 多起点・１終点の場合*

Some Efficient Algorithms for Semi-dynamic Traffic Equilibrium Assignment with Queue Evolution - Many-to-One O-D Pair Case*

吉相俊**・赤松隆***・和田健太郎****

By Sangjoon KIL**・Takashi AKAMATSU***・Kentaro WADA****

１．はじめに

道路網の計画や近年盛んに行われているTDM施策において，ピーク時の交通状態の予測は不可欠である．しかしながら，従来の静的配分では渋滞状態を表現することができない. 一方, 渋滞状態を表現できる動的配分は, 時々刻々のOD交通量データの入手が困難である.

これらの問題点に対し，近年，渋滞を明示的に表現でき，かつ，モデルの入力データが入手可能な時間帯別交通均衡配分が菊池・赤松モデル[1]，中山モデル[2]が提案されている．菊地・赤松モデルは，一般形の非線形相補性問題（NCP：Nonlinear Complementarity Problem）に帰着可能であり，NCPに対する多くの標準的なアルゴリズムを活用することで，効率的な解法の開発が期待される．また，DP原理が成立しているため，問題を時間帯毎に分割し解くことが可能である．一方，中山モデルは，解の一意性が保証されるという利点がある．しかし，DP原理が成立しないため，全時間帯を同時に計算しなければならない．即ち，中山モデルは大規模問題に適用可能な解法の開発は前者より困難である．

そこで本研究では，菊地・赤松[1]が提案した時間帯別交通均衡配分を対象に，効率的な解法を開発することを目的とする．提示する解法は，1）厳密解への収束の保証ができ，2）大規模ネットワークに適用できるという特徴を持つ．また，数値実験により，ネットワーク規模，混雑状況が，提示した解法の効率性に与える影響を明らかにする．ただし，本研究では，single-commodityフローのみを対象とする．

２．時間帯別交通均衡配分モデル

本研究で対象とするモデルは，基本的に菊地・赤松[1] に示されたものと同一である（詳細は菊地・赤松[1]を参照）. 唯一違う点は，本研究では利用者の経路選択均衡条件が確率的な場合を新たに追加したことである．これは，菊地・赤松モデルでは, 経路選択均衡条件が確定的な場合, multi-commodityフローに対して解の一意性が保証できないためである．

道路網はノード集合 N とそれらを結ぶ有向リンク集合 L からなるネットワークによって表現されるものとする．また，時間の流れをある一定の長さを持つ時間帯ごとに分割し，離散的に考える．このとき，状態変化は時間帯間でのみ起こり，時間帯内では定常状態にあると仮定する．この状況設定の下，時間帯別交通均衡配分モデルでは，以下に示す5つの条件が同時に成立する．

a）各リンクでの待ち行列進展条件

リンク (i,j) の時間帯 t での流入台数を $\lambda_{ij}(t)$ ，流出台数を $\mu_{ij}(t)$ ，待ち行列台数を $x_{ij}(t)$ とすれば，リンクでの時間帯を追った車両台数の保存則を表す条件は，次のように表せる．

$$x_{ij}(t) = x_{ij}(t-1) + \lambda_{ij}(t) - \mu_{ij}(t) \quad \forall (i,j) \qquad (2.1)$$

b）各リンクの容量制約条件

リンク (i,j) のサービス率には, 物理的特性から決まる上限(i.e., 最大流出率： μ_{ij}^* ）があると考える. このとき, 時間帯 t において，待ち行列が存在する場合では流出台数は最大流出率となり, そうでない場合には, 時間帯 t の流入台数 $\lambda_{ij}(t)$ と時間帯 $t-1$ の待ち行列台数 $x_{ij}(t-1)$ の和に等しくなる．即ち，次の条件を満たす.

$$\begin{cases} \mu_{ij}(t) = \mu_{ij}^* & if \quad x_{ij}(t) > 0 \\ \mu_{ij}(t) \le \mu_{ij}^* & if \quad x_{ij}(t) = 0 \end{cases} \quad \forall (i,j) \qquad (2.2)$$

c）各ノードでのフロー保存則

ネットワーク上の各ノードでは，流入台数と流出台数が等しくなる必要がある．よって，各ノードにおけるフロー保存則は次のように表される．

$$\sum_{i \in I(k)} \mu_{ik}(t) - \sum_{j \in O(k)} \lambda_{kj}(t) + q_k(t) = 0 \quad \forall k \ne d \qquad (2.3)$$

ただし，$q_k(t)$ は, 時間帯 t に起点 k を出発して終点に向かうOD交通量である．また，$I(k)$ はノード k へ流入するリンクの上流側ノードの集合，$O(k)$ はノード k から流出するリンクの下流側ノードの集合である.

d）リンク旅行時間とフロー及び待ち行列の関係

リンク (i,j) の旅行時間 $c_{ij}(t)$ は，走行リンクで費やす時間 $m_{ij}(t)$ と待ち行列リンクにおける渋滞待ち時間の和

---

*キーワーズ：交通配分，渋滞，時間帯別配分，NCP

**学生員，東北大学大学院　情報科学研究科　博士課程前期

（仙台市青葉区荒巻青葉6-6，TEL022-795-7507，FAX022-795-7505）

***正員，工博，東北大学大学院教授　情報科学研究科

****学生員，東北大学大学院　情報科学研究科　博士課程前期

として次のように表される．

$$c_{ij}(t) = m_{ij}(\lambda_{ij}) + x_{ij}(t) / \mu_{ij}^{*} \tag{2.4}$$

$$m_{ij}(\lambda_{ij}(t)) = m_0 \left\{ 1 + a\left(\lambda_{ij}(t) / \mu_{ij}^{*}\right)^{b} \right\} \tag{2.5}$$

ここで，m_0 はリンク固有の値であり，$m_{ij}(t)$ はBPR関数で表されるとする．

e）利用者の経路選択均衡条件

利用者の経路選択が確率的である場合を考える．このとき，（リンク・ベースの）利用者のロジット選択は次式で与えられる（導出は，Akamatsu[3]を参照）．

$$c_{ij}(t) + p_j^{\theta}(t) - p_i^{\theta}(t) + \frac{1}{\theta} \ln \frac{\lambda_{ij}(t)}{\sum_k \lambda_{ik}(t)} = 0 \tag{2.6}$$

ここで，θ はロジット・パラメータであり，$p_i^{\theta}(t)$ はノード i から終点までの期待最小経路所要時間である．上記の式(2.6)は，ロジット・パラメータ θ が $\theta \to \infty$ の時，菊地・赤松[1]で示された経路選択条件に一致する．

以上で定式化されたモデルは，時間帯 $t-1$ に関する変数が決まれば，時間帯 t のみを未知変数とした問題として考えられる．即ち，前向きDP原理が成立している．このことを利用すれば，時間帯 $t-1$ に関する変数を所与とした以上の定式化は，次に示すNCPと等価である（詳細は菊池・赤松[1]を参照）．

[NCP]　　Find　$\mathbf{Y} \in \mathbf{\Omega}$　such that

$$\mathbf{Y} \cdot \mathbf{F}(\mathbf{Y}) = 0,\ \mathbf{Y} \geq \mathbf{0},\ \mathbf{F}(\mathbf{Y}) \geq \mathbf{0} \tag{2.7}$$

ここで，

$$\Omega = R_+^{|L|} \times R_+^{|L|} \times R_+^{|N|},\quad \mathbf{Y} = \begin{bmatrix} \mathbf{x} & \boldsymbol{\lambda} & \mathbf{p} \end{bmatrix}^T,$$

$$\mathbf{F}(\mathbf{Y}) = \begin{bmatrix} \mathbf{I} & -\mathbf{I} & \mathbf{0} \\ \mathbf{M}^{-1} & \mathbf{0} & -\mathbf{A}^T \\ -\mathbf{A}_- & \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \\ \mathbf{p} \end{bmatrix} + \begin{bmatrix} \bar{\boldsymbol{\mu}} \\ \mathbf{m} + \boldsymbol{\Pi} \\ -\mathbf{r} \end{bmatrix} \tag{2.8}$$

である．$\mathbf{A}$ はリンク・ノード接続行列であり，$\mathbf{A}_+$（$\mathbf{A}_-$）はリンク・ノード接続行列において -1 (1) 要素を 0 に置き換えた行列である．また，$\mathbf{M}$ は対角要素に μ_{ij}^{*} を持つ $|L| \times |L|$ 次対角行列であり，$\bar{\boldsymbol{\mu}}$，$\boldsymbol{\Pi}$，$\mathbf{r}$ は，各々，

$$\bar{\mu}_{ij} \equiv \mu_{ij}^{*} - x_{ij}(t-1) \tag{2.9}$$

$$r_k \equiv \sum_{i \in I(k)} x_{ij}(t-1) + q_k \tag{2.10}$$

$$\pi_{ij} \equiv (1/\theta) \ln(\lambda_{ij} / \sum_k \lambda_{ik}) \tag{2.11}$$

と定義される要素を持つベクトルである．これにより，モデルの数学的構造を明らかにするのが容易になり，また，効率的なアルゴリズムの開発が可能となる．

3．時間帯別交通均衡配分の効率的解法の提案

本研究で提示する解法は，NCPに対し，適切な条件の下では，収束が保証される枠組に基づいている．具体的には，収束が非常に速い（二次収束）Newton法系統の枠組を活用する．この枠組は，大きく2つに分類できる．第1は，微分不可能な関数を，劣勾配を活用し解こうとするInexact Newton Methodと呼ばれる一群の枠組である．第2は，微分不可能な関数を微分可能な関数に近似し，本来解くべき問題を間接的に解くSmoothing Newton Methodと呼ばれる一群の枠組である．本研究では，それぞれの枠組を時間帯別交通均衡配分へ適用した．ただしここでは，よりロバストな結果が得られた後者のみを示す（Inexact Newton Methodについては，吉等[4]を参照）．

本研究では，Smoothing Newton Methodの枠組の中で，Qi and Liao[5]が提案したアルゴリズムを採用する．このアルゴリズムは，smoothingパラメータを内生変数として扱っており，収束が保証できるという特徴がある．

本稿では，紙面の都合により枠組の詳しい説明は省略する．ここでは，数値計算の際に最も問題となる，解の改訂ベクトルを効率的に計算するための工夫を示す．まず，次のようなNCPを考える．

$$\mathbf{x} \geq 0,\ \mathbf{F}(\mathbf{x}) \geq 0,\ \mathbf{x}^T \mathbf{F}(\mathbf{x}) = 0 \tag{3.1}$$

次に，Qi and Liao[5]での解の改定ベクトル $\mathbf{d}^k$ は，以下の方程式から求められる．

$$\nabla H(\mathbf{z}^k) \Delta \mathbf{z} = -H(\mathbf{z}^k) \tag{3.2}$$

ここで，

$$\mathbf{z} := (\mu, \mathbf{x}) \tag{3.3}$$

$$H(\mathbf{z}) = H(\mu, \mathbf{x}) := \begin{bmatrix} e^{\mu} - 1 \\ \mathbf{\Phi}_{\mu}(\mathbf{x}) \end{bmatrix} = 0, \quad \mu > 0 \tag{3.4}$$

$$\mathbf{\Phi}_{\mu}(\mathbf{x}) := \begin{bmatrix} \phi_{\mu}(\mathbf{x}_1, \mathbf{F}_1(\mathbf{x})) \\ \vdots \\ \phi_{\mu}(\mathbf{x}_n, \mathbf{F}_n(\mathbf{x})) \end{bmatrix} \tag{3.5}$$

$$\phi_{\mu}(\mathbf{x}, \mathbf{F}(\mathbf{x})) := \sqrt{\mu^2 + \mathbf{x}^2 + \mathbf{F}(\mathbf{x})^2} - \mathbf{x} - \mathbf{F}(\mathbf{x}) \tag{3.6}$$

と定義される．また，μ はsmoothingパラメーターである．

上記の解法を本モデルに適用すると，改訂ベクトルを求めるための連立方程式(3.2)のJacobian $\nabla H(\mathbf{z})$ は，以下のように表される．

$$\nabla H(\mathbf{z}) = \begin{bmatrix} e^{\mu} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \gamma_1 & \alpha_1 + \beta_1 & -\beta_2 & \mathbf{0} \\ \gamma_2 & \mathbf{M}^{-1}\beta_1 & \alpha_2 & -\mathbf{A}^T \beta_3 \\ \gamma_3 & -\mathbf{A}_- \beta_1 & \mathbf{A}\beta_2 & \alpha_3 \end{bmatrix} \tag{3.7}$$

ただし，γ_i は，

$$\gamma_i = \frac{\mu}{\sqrt{\mu^2 + \mathbf{Y}_i^2 + \mathbf{F}_i(\mathbf{Y})^2}} \tag{3.8}$$

と表される列ベクトルであり，α_i と β_i $(i = 1,2,3)$ は，それぞれ以下のような対角行列：

$$(\alpha_i)_{k,k} = \frac{(\mathbf{Y}_i)_k \gamma_i}{\mu} - 1,\quad (\beta_i)_{k,k} = \frac{(\mathbf{F}_i(\mathbf{Y}))_k \gamma_i}{\mu} - 1 \tag{3.9}$$

表-1　解法内のパラメーターの設定

パラメーター	値	説明
ν	0.5	step sizeの感度parameter
ω	10^{-4}	step sizeの判定parameter
$l_{\max}$	16	step sizeの判定parameter
α	10^{-4}	smoothing parameterの更新parameter
η	10^{-4}	smoothing parameterの更新parameter

である．以上から，連立方程式(3.2)の係数行列(3.7)は，疎行列で構成されることが分かる．具体的には，リンク・ノード接続行列$\mathbf{A}$，リンク容量$\mathbf{M}$（対角行列），式(3.8)，式(3.9)の疎行列で構成される．ただし，式(3.8)，式(3.9)は，反復計算の中で毎回計算されるものであり，記憶する必要はない．そのため，記憶しなければならないのは，$\mathbf{A}$ と$\mathbf{M}$ のみである．即ち，記憶容量は，高々リンク数のオーダーの約3倍程度である．例えば，Jacobianのオーダーが10^8 の問題に対して，約3×10^4 の記憶容量しか必要としない．

さらに，方程式(3.2)は，ブロック分割して解くことができる．具体的には，まず$\Delta\boldsymbol{\lambda}$ を消去する：

$$\begin{bmatrix} e^{\mu} & \mathbf{0} & \mathbf{0} \\ \dfrac{\alpha_2}{\beta_2}\gamma_1+\gamma_2 & \dfrac{\alpha_2}{\beta_2}(\alpha_1+\beta_1)+\mathbf{M}^{-1}\beta_1 & -\mathbf{A}^T\beta_3 \\ \mathbf{A}\gamma_1+\gamma_3 & \mathbf{A}\alpha_1+\mathbf{A}_{+}\beta_1 & \alpha_3 \end{bmatrix}\begin{bmatrix}\Delta\mu \\ \Delta\mathbf{x} \\ \Delta\mathbf{p}\end{bmatrix} = -\begin{bmatrix} e^{\mu}-1 \\ \mathbf{\Phi}_{\mu,2}(\mathbf{Y})+\dfrac{\alpha_2}{\beta_2}\mathbf{\Phi}_{\mu,1}(\mathbf{Y}) \\ \mathbf{\Phi}_{\mu,3}(\mathbf{Y})+\mathbf{A}\,\mathbf{\Phi}_{\mu,1}(\mathbf{Y}) \end{bmatrix}. \qquad (3.10)$$

式(3.10)の係数行列は，一見密に見えるが，要素毎にみると，式(3.7)とほとんど変わらないことが分かる．これに加え，未知変数のオーダーが小さくなったため，式(3.7)より記憶容量を節約することができる．

上記の$\Delta\boldsymbol{\lambda}$ を消去した式(3.10)から，待ち行列台数の改訂ベクトル$\Delta\mathbf{x}$ と期待最小経路所要時間の改訂ベクトル$\Delta\mathbf{p}$ が決まる．また，消去された$\Delta\boldsymbol{\lambda}$ については，次式：

$$\Delta\boldsymbol{\lambda} = \frac{1}{\beta_2}(r_1\Delta\mu+(\alpha_1+\beta_1)\Delta\mathbf{x}+\mathbf{\Phi}_{\mu,1}(\mathbf{Y})) \qquad (3.11)$$

から，事後的に求めることができる．

4．数値実験による効率性の検証

Smoothing Newton Methodにおいて，解法の効率性（e.g., 収束するまでの反復回数）に影響を与える要因として，(1)ネットワーク規模，(2)混雑度が考えられる．そこで，本章では，それぞれの要因が効率性にどのような影響を与えるか明らかにする．調べる項目は，厳密解へ収束するまでの反復回数，1回当たりの計算時間である．

ここでは，ベース・ケース：

(a)　ネットワーク規模：
5×5 lattice (リンク数：80，ノード数：25).

(b)　リンク容量，リンク自由走行時間：
0～3 の範囲内で，乱数を発生.

(c)　OD交通量：
0～2 の範囲内で，乱数を発生.

(d)　経路選択のロジット・パラメーターθ ：
　ｉ）確定的な場合 ($\theta=\infty$)
　ii）確率的な場合 ($\theta=100$).

を考え，ネットワーク規模に関する実験では(a)の項目を変化させる．具体的には，latticeネットワークの一辺に含まれるノード数を増やし，それに伴いリンク数も増加する．混雑度に関する実験では，まず，ベース・ケースに対し，(c)を2.5，5倍に変化させる．その上で，各々，実現したリンクコストの総和を求め，そのベース・ケースとの比を混雑度と定義する．また，数値実験の際の収束判定は，リンク交通量$\lambda_{ij}(t)$ の乖離割合の最大値：

$$D^n := \max_{ij\in L}(\lambda_{ij}^n-\lambda_{ij}^*)/\lambda_{ij}^* \qquad (4.1)$$

が10^{-4} 以下（誤差0.01%以下），または反復回数が1000回以上になれば，収束したと判断する．解法内のパラメーターの設定は，表-1に従う．なお，数値実験で用いた計算機は，CPU:Pentium 4 @ 2.80GHz, RAM:1GBである．

（1）ネットワーク規模

実験結果を，図-1，図-2に示す．図-1は，各ネットワーク規模における1回当たりの計算時間を示している．縦軸は，1回当たりの計算時間であり，横軸はネットワーク規模（リンク数）を表している．また，図-2は（確率的な場合の）各ネットワーク規模での収束過程を表したもので，縦軸は乖離度D^n，横軸は反復回数を表している．ここで，図-1，図-2に示すケースA～Cは各々対応している．Aはベース・ケースを表しており，B，Cはベース・ケースからリンク数を増加させたケースを表している．

図-1から，1回当たりの計算時間に関しては，確定的な場合でも，確率的な場合でも，ネットワーク規模に関して線形的にしか増加しないといえる．これは，実利用を考えると非常に望ましい性質である．また，図-2から，図-1で示したA～Cのいずれのケースでも同程度の反復回数で収束していることが分かる．即ち，反復回数に関しては，ネットワーク規模が大きくなっても，必ずしも増加しないことが分かる．

（2）混雑度

実験結果を，図-3，図-4に示す．図-3は，各混雑度における1回当たりの計算時間を示している．縦軸は1回当たりの計算時間であり，横軸は混雑度を表している．また，図-4は確率的な場合の各混雑度での収束過程を表したもので，縦軸は乖離度D^n，横軸は反復回数を表している．

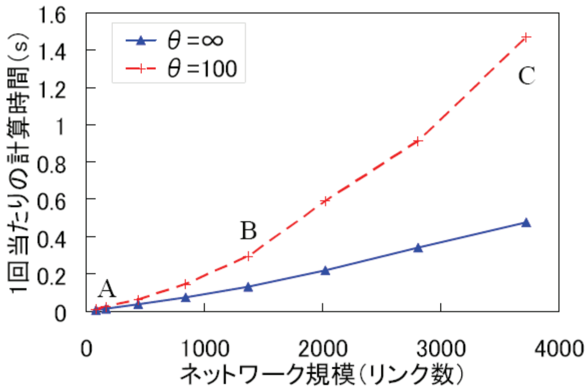

図-1　ネットワーク規模と1回当たりの計算時間

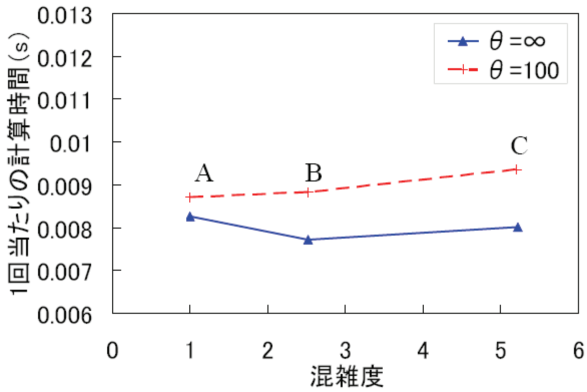

図-3　混雑度と1回当たりの計算時間

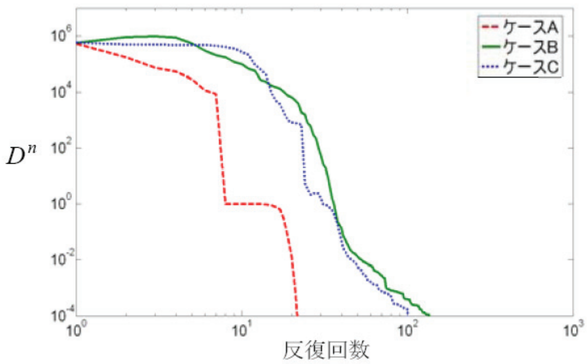

図-2　各ネットワーク規模の収束過程（$\theta=100$）

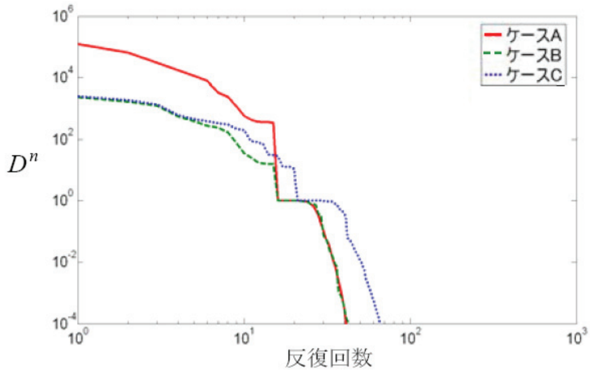

図-4　各混雑度の収束過程（$\theta=100$）

ここで，図-3，図-4のケースA～Cは対応しており，ケースAはベース・ケースを，B，Cは，各々ベース・ケースの混雑度を2.5倍，5倍させた場合に対応している．

図-3より，1回当たりの計算時間は，確定的な場合でも，確率的な場合でも，混雑度とは概ね関係がないという結果が得られた．また，図-4から，図-3で示したA～Cのいずれのケースでも同程度の反復回数で収束していることが分かる．これは，（1）と同様の結果を表しており，本アルゴリズムが，ネットワーク規模に関しても，混雑度に関しても，十分実用的な反復回数で収束することが確認された．

6．おわりに

本研究では，大規模問題に対しても適用可能な，時間帯別交通均衡配分の効率的解法を提示した．より具体的には，本モデルがNCPとして表現可能であることに着目し，NCPに対し収束が保証された枠組である：(1)Inexact Newton Method，(2)Smoothing Newton Methodの2つを活用した（Inexact Newton Methodについては吉等[4] 参照）．また，この枠組を本モデルに適用する際に，大規模な問題に対して適用可能にするため，工夫をした．そして，数値実験により，smoothing Newton methodはロバスト性が高いことを明らかにした．この解法は，非常に実用的であり，大規模ネットワークや非常に混雑している状況でも，十分実用的な反復回数で収束することを確認した．具体的には，次のような結果を得た：

(1)ネットワーク規模によらず，実用的な回数で収束
(2)ネットワーク規模の増加と共に1回当たりの計算時間は線形的にしか増加しない
(3)十分混雑した状況でも，混雑してないときとほぼ同じ反復回数で収束
(4)混雑度と1回当たりの計算時間は無関係.

また，本稿ではsingle-commodityフローの場合のみ説明したが，multi-commodityフローの場合の解法も開発までは既に終わっている．これに関しては，数値実験が終わり次第，報告したい．

参考文献

1）菊地志郎，赤松隆：リンクの流入･流出交通量を内生化した時間帯別交通均衡配分に関する基礎的研究，土木計画学研究・論文集，No.24，pp.577-585，　2006.
2）中山晶一朗：混雑の時間空間移動を考慮した準動的配分モデル，土木学会論文集，vol.64, pp.340-353, 2008.
3）T.Akamatsu ：Decomposition of path choice entropy in general transport networks, Transportation Science, vol. 31, pp.349-362, 1997.
4）吉相俊，赤松隆，菊地志郎，井上紳一：時間帯別交通均衡配分の効率的解法，土木計画学研究・論文集，投稿中.
5）H.Qi，L.Liao：A Smoothing Newton method for general nonlinear complementarity problems ， Computational Optimization and Applications，vol.17，pp.231-253，2000.