UCI-ITS-TS-WP-05-2

The Dynamic System of the Traffic Assignment Problem: Part II. Computation

UCI-ITS-TS-WP-05-2

Wen-Long Jin

Institute of Transportation Studies, University of California, Irvine
Irvine, CA 92697-3600, USA
wjin@uci.edu

March 2005

Institute of Transportation Studies
University of California, Irvine
Irvine, CA 92697-3600, U.S.A.
http://www.its.uci.edu

The dynamic system of the traffic assignment problem: Part II. Computation

Wen-Long Jin *

March 15, 2005

Abstract

In the preceding paper, we demonstrated the existence of *the* dynamic system of the traffic assignment problem (TAP), called J-system, whose steady states are J-equilibria (JE) and only stable steady states are user-equilibria (UE). Based on this new definition of UE, in this paper, we develop algorithms for computing both JE and UE of general transportation networks. To find JE, we can use Newton's method or the finite difference method; to find UE, we propose a perturbation-based method and a hybrid method incorporating Frank-Wolfe's method. With an example, we show that both Newton's and finite difference methods converge exponentially measured by J-index of convergence, and the perturbation-based method yields UE solutions. We also present methods for solving TAP with variable demand. The finite difference method and the perturbation-based method are intrinsic to J-system and readily applicable for solving TAP of any general networks. The implementation and efficiency of these methods for solving TAP of large-scale transportation networks will be investigated in the future.

Keywords: Static traffic assignment problems, Fixed demand, Variable demand, Newton's method, Finite difference method, J-index of convergence, Perturbation-based method, Hybrid method

*Institute of Transportation Studies, University of California, 522 Social Science Tower, Irvine, CA 92697, USA. Email: wjin@uci.edu. Corresponding author

1 Introduction

In this study we adopt the network notation system given in **Table** 1. We have the following basic relationships:

$$q_{rs} = \sum_k f_k^{rs}, \qquad \forall\ r,s \tag{1a}$$

$$f_k^{rs} \geq 0, \qquad \forall\ k,r,s; \tag{1b}$$

$$q_{rs} \geq 0, \qquad \forall\ r,s; \tag{1c}$$

and

$$x_a = \sum_{r,s} \sum_k f_k^{rs} \delta_{a,k}^{rs}, \qquad \forall\ a \tag{2a}$$

$$c_k^{rs} = \sum_a t_a \delta_{a,k}^{rs}, \qquad \forall\ k,r,s. \tag{2b}$$

In addition, we use abbreviations in **Table** 2.

In (Jin, 2005), based on the definition of J-functions of FIFO violation, the following dynamic system, called J-system, is proposed for static TAP with variable demand,

$$\dot{f}_k^{rs} = -q_{rs} f_k^{rs}(c_k^{rs} - u_{rs}), \quad \forall\ k,r,s \tag{3a}$$

$$\dot{q}_{rs} = -q_{rs}(\sum_k f_k^{rs} c_k^{rs} - q_{rs} u_{rs}), \quad \forall\ r,s \tag{3b}$$

or

$$\dot{f}_k^{rs} = -f_k^{rs} \sum_j (c_k^{rs} - c_j^{rs}) f_j^{rs} - q_{rs} f_k^{rs}(v_{rs} - u_{rs}), \quad \forall\ k,r,s \tag{3c}$$

$$\dot{q}_{rs} = -q_{rs}^2 (v_{rs} - u_{rs}), \quad \forall\ r,s \tag{3d}$$

where the derivative is with respect to a decision variable τ, u_{rs} is the inverse of the demand function associated with O-D pair $r-s$, and v_{rs} is the average O-D travel time defined by

$$v_{rs} = \frac{\sum_j f_j^{rs} c_j^{rs}}{q_{rs}}, \qquad \forall\ k,r,s. \tag{4}$$

Denoting q_{rs} by f_0^{rs}, $\mathbf{f}^{rs} = (f_0^{rs}, \cdots, f_k^{rs}, \cdots)$, and $\mathbf{f} = (\cdots, \mathbf{f}^{rs}, \cdots)$, we can write J-system in the following form

$$\dot{\mathbf{f}} = -\mathbf{J}(\mathbf{f}) \tag{5}$$

whose steady states are called J-equilibria (JE). As shown in (Jin, 2005), J-system is *the* dynamic system of TAP in the sense that the BMW objective function (Beckmann, McGuire, and Winsten, 1956) is its total energy and user equilibria (UE) are its only globally stable steady states.

In this paper, based on the definitions of JE and UE, we propose algorithms for computing them. We start with TAP with fixed demand, in which $\dot{q}_{rs} = 0$ and $v_{rs} = u_{rs}$ for all O-D pairs $r - s$. We show the feasibility of these methods by solving a simple network. Then we propose similar methods for solving TAP with variable demand.

The rest of the paper is organized as follows. In Section 2, we present methods for solving JE for fixed demand case. In Section 3, we propose methods for solving UE for fixed demand case. In Section 4, we present methods for solving TAP with variable demand and some discussions.

2 Algorithms for solving JE

In this section, we present two methods for solving JE of (3) satisfying $\mathbf{J}(\mathbf{f}) = 0$ for fixed demand case. As we know, if the initial value when $\tau = 0$ of $f_k^{rs} = 0$ for any k, r, s, it remains at zero in J-system for $\tau > 0$. That is, if a path is excluded in the initial state, it is always excluded. Thus we only consider those paths with non-zero initial flows. Denoting the number of initially non-empty paths connecting O-D pair $r - s$ by K_{rs} ($\forall\ r, s$), we then have a K_{rs}-dimension hyper-plane $\mathcal{G}_{rs}$, in which $\tilde{\mathbf{f}}_{rs} = (f_1^{rs}, \cdots, f_{K_{rs}}^{rs})$, $q_{rs} = \sum_{k=1}^{K_{rs}} f_k^{rs}$, and $f_k^{rs} \geq 0$ for $1 \leq k \leq K_{rs}$. Further, we define the total hyper-plane as $\mathcal{G} = \prod_{r,s} \mathcal{G}_{rs}$. Therefore, all solutions of (3), including JE, remain in $\mathcal{G}$ for $\tau > 0$.

2.1 Newton's method

Given an initial $\mathbf{f}_1 \in \mathcal{G}$, if $\mathbf{J}(\mathbf{f}_1) \neq 0$, we can find the next candidate $\mathbf{f}_2$ from the following linear equations by linearizing $\mathbf{J}$ at $\mathbf{f}_1$:

$$\mathbf{J}(\mathbf{f}_1) + \nabla \mathbf{J}|_{\mathbf{f}_1} (\mathbf{f}_2 - \mathbf{f}_1) = 0. \tag{6}$$

It is not hard to show that the Jacobian matrix $\nabla \mathbf{J}|_{\mathbf{f}_1}$ is not symmetric. Since $\nabla \mathbf{J}|_{\mathbf{f}_1}$ could be ill-conditioned, it is suggested to use iterative solution methods for non-symmetric matrices, such as GMRES (Trefethen and Bau, III, 1997). Since $\mathbf{f}_2$ might not satisfy (1), we have to adjust it as follows:

$$\bar{f}_k^{rs} = f_k^{rs} \frac{q_{rs}}{\sum_k f_k^{rs}}, \qquad \forall\ k, r, s. \tag{7}$$

Here, the stop condition is based on $\|\mathbf{J}(\mathbf{f})\|_2$, called J-index of convergence,

$$\|\mathbf{J}(\mathbf{f})\|_2 \quad = \quad \sqrt{\sum_{rs} \sum_{k} (J_k^{rs})^2 / \sum_{rs} K_{rs}}, \tag{8}$$

which equals to zero iff $\mathbf{f}$ is a JE.

In Newton's method, the Jacobian $\nabla \mathbf{J}|_{\mathbf{f}_1}$ is a square matrix with dimension of $\prod_{rs} K_{rs} \times \prod_{rs} K_{rs}$, whose $((rs, k), (\bar{r}\bar{s}, \bar{k}))$ element is $\partial J_k^{rs} / \partial f_{\bar{k}}^{\bar{r}\bar{s}}$. For example, for a network with n O-D pairs and m paths for each O-D pair, then the size of the Jacobian matrix is $m^n \times m^n$. Therefore, computational cost will be proportional to $m^{2n\ 1}$, which is very costly for a large network. Here we propose a modified Newton's method, by considering only the diagonal blocks of the Jacobian matrix, $\nabla \mathbf{J}_{rs}|_{\mathbf{f}_1}$, whose (k, j) element is $\partial J_k^{rs} / \partial f_j^{rs}$ $(\forall\ r, s)$. Then we solve the following linear equations

$$\mathbf{J}_{rs}(\mathbf{f}_1) + \nabla \mathbf{J}_{rs}|_{\mathbf{f}_1} (\mathbf{f}_{rs,2} - \mathbf{f}_{rs,1}) = 0, \qquad \forall\ r, s. \tag{9}$$

That is, we update $\mathbf{f}_2$ for each O-D pair separately. In this method, computational cost is proportional to nm^2 in the aforementioned network, which is significantly smaller. However, this method is heuristic and subject to numerical and theoretical examinations.

2.2 Finite difference method

We can discretize J-system, (3), in the decision space as

$$\frac{\mathbf{f}(\tau + \Delta\tau) - \mathbf{f}(\tau)}{\Delta\tau} = -\mathbf{J}(\mathbf{f}(\tau)),$$

[1]Since the Jacobian matrix could be sparse, special treatment can be used to reduce the computational load.

therefore

$$\mathbf{f}(\tau + \Delta\tau) = \mathbf{f}(\tau) - \Delta\tau \mathbf{J}(\mathbf{f}(\tau)). \tag{10}$$

In this way, we can compute $\mathbf{f}(\tau)$ for any τ given an initial guess $\mathbf{f}(0)$. This method is called Euler's method (Strogatz, 1994). Since J-system is stable, its asymptotic solutions converge to steady states, i.e. JE, for very large τ. To have stable numerical method, $\Delta\tau$ should be small enough, which can be determined by analyzing properties of a specific J-system.

To find JE with the finite difference method, we need to re-assign these path flows to links to compute path travel times at each decision step and obtain J-functions $\mathbf{J}$ for the next step. Therefore, the total computational load of the finite difference method is proportional to $\tau/\Delta\tau$ and that of computing J-functions. To have a more efficient method, one approach is to use the fourth-order Runge-Kutta method (Strogatz, 1994), which is still stable for larger decision step $\Delta\tau$.

We can see that the finite difference method is an intrinsic method to J-system for solving its steady states. Note that computational cost in (10) should be smaller than in Newton's method, and we do not need adjustment as in Newton's method since (1) is always satisfied with (10).

3 Algorithms for searching UE

The difference between JE and UE are their stability or existence of shorter empty paths. Based on this difference, we can search for UE from JE solutions by perturbing the latter.

3.1 Perturbation-based method

With any initial guess, by using methods in Section 2 we can find a JE. Further, if we can find empty shorter paths at this JE, we then perturb the JE by shifting proportions of flow from an non-empty path to the shorter empty paths. Then we can find a different JE, since non-UE JE are unstable in these directions (Jin, 2005, Appendix B). We repeat this process until we obtain UE, where J-index is small enough and there are no shorter empty paths. The algorithm is shown in **Table** 3. Note that the initial guess does not have to be all-or-nothing assignment as in Frank-Wolfe's (FW) method (Sheffi, 1984). In stead, with K-shortest paths as initial state, the perturbation-based method could find UE without many perturbations.

3.2 Hybrid methods

Another approach is to use FW method to get close to UE, and then use FW result as an initial guess of J-system to find the corresponding steady states. As we know, FW method usually takes several steps to find alternative paths, but there's zigzagging effect around UE. In the hybrid method, we could take advantage of the solution methods in Section 2 to avoid such effect. If FW method still misses alternative paths, however, we need to repeat using FW method or the perturbation-based process. The hybrid method is shown in **Table** 4.

We can also hybridize the methods for solving JE, especially the finite difference method, with other traditional methods for solving TAP (Patriksson, 1994).

4 A simple example

In this section, we study a simple network given by (Fig. 5.1 (page 114) of Sheffi, 1984), which is shown in Figure 1. For this network, J-functions of FIFO violation can be written as

$$J_1 = x_1 \left(\left(10(1+0.15(\frac{x_1}{2})^4) - 20(1+0.15(\frac{x_2}{4})^4) \right) x_2 + \left(10(1+0.15(\frac{x_1}{2})^4) - 25(1+0.15(\frac{x_3}{3})^4) \right) x_3 \right)$$
$$J_2 = x_2 \left(\left(20(1+0.15(\frac{x_2}{4})^4) - 10(1+0.15(\frac{x_1}{2})^4) \right) x_1 + \left(20(1+0.15(\frac{x_2}{4})^4) - 25(1+0.15(\frac{x_3}{3})^4) \right) x_3 \right)$$
$$J_3 = x_3 \left(\left(25(1+0.15(\frac{x_3}{3})^4) - 10(1+0.15(\frac{x_1}{2})^4) \right) x_1 + \left(25(1+0.15(\frac{x_3}{3})^4) - 20(1+0.15(\frac{x_2}{4})^4) \right) x_2 \right)$$

whose Jacobian matrix of dimension 3×3 can be computed, but not shown here. Therefore, J-system of the network is

$$\dot{x}_1 = -J_1,$$

$$\dot{x}_2 = -J_2,$$

$$\dot{x}_3 = -J_3.$$

4.1 Convergence of methods for solving JE

In this subsection, we use initial state at $(x_1, x_2, x_3) = (3.39, 5.00, 1.61)$ and the corresponding path costs $(c_1, c_2, c_3) = (22.3, 27.3, 35.3)$. This is the first solution with all alternative UE paths in Table 5.4 of (Sheffi, 1984). We will show convergence rates and converging trajectory in the region of $x_1 + x_2 \leq 10$ ($x_3 = 10 - x_1 - x_2$) for Newton's method and the finite difference method.

For Newton's method, the convergence of J-index defined in (8) is shown in Figure 2, and the solution trajectory in Figure 3. We can see that Newton's method converges exponentially, but has zigzagging effect around the steady state.

For finite difference method with $\tau = 0.02$ and $\Delta\tau$ =0.0005, 0.001, and 0.002, the convergence of J-index is shown in Figure 4, and the solution trajectory in Figure 5. From these figures, we can see that as long as the finite difference form of (3) is stable (i.e. $\Delta\tau$ is sufficiently small), convergence rates are similar and solution trajectories are also similar. Especially, convergence rates are constant when solutions get closer to JE and there is no zigzagging effect.

4.2 All JE solutions and perturbation-based method

With either Newton's method or the finite difference method, we start with different combinations of non-zero path flows and find all seven JE for this network as shown in **Table 5**.

In Figure 6, the seven JE are shown. We perturb the first six non-UE JE by shifting

0.05 flow from a nonempty path to each empty shorter path. After perturbation, we use the finite difference method with $\Delta\tau = 0.0005$ and $\tau = 0.1$ to obtain solution trajectories of J-system, which converge to UE, as also shown in Figure 6. This confirms the theory in (Jin, 2005) that only UE are stable steady states. This simple example also demonstrates the feasibility of the perturbation-based method in finding UE.

5 Discussions

In this paper, we studied two methods (Newton's method and finite difference method) for finding JE and two methods (perturbation-based method and hybrid methods) for searching UE. From a numerical example, we show that solutions of JE converge exponentially for both Newton's and finite difference methods, and the perturbation-based method can yield UE by perturbing a non-UE JE. Since Newton's method requires the Jacobian matrix, it only works for simple networks and simple link performance functions. In contrast, the finite difference method is more general, since it can be used as long as J-functions of FIFO violation can be computed. That is, as long as there exists J-system of a traffic assignment problem, the finite difference method and the perturbation-based method are applicable. In this sense, the finite difference method and the perturbation-based method are intrinsic to J-system. Therefore, in the future, for complicated networks or dynamic traffic assignment, we can always use these intrinsic methods to find possible UE solutions.

This study is intended to show the feasibility of J-system formulation in solving the traffic assignment problem. For large-scale road networks, more issues on implementation and efficiency should be investigated in the future. For example, knowledge in dynamic systems and special treatment of special network structure can be incorporated to find UE more efficiently. Also, we may use many other methods for solving nonlinear equations directly, such as S-system method (Savageau and Voit, 1987).

The methods proposed here can also be used to find system optimal assignments and solve any other TAP with J-system formulation. For example, for TAP with variable demand, we can have two types of methods. In one, with an initial guess of $f_0^{rs} = q_{rs}$ (for any r, s) and f_k^{rs} (for any k, r, s), we directly solve (3a) and (3b), by using finite difference method and perturbation-based method. In the other, with an initial guess of q_{rs} (for any r, s), we solve the corresponding UE f_k^{rs} (for any k, r, s) for a fixed demand problem (3c) with $v_{rs} = u_{rs}$, and then update q_{rs} (for any r, s) and f_k^{rs} (for any k, r, s) with the difference between v_{rs} and u_{rs} by (3c) and (3d) respectively.

At the same time, the following questions are worth looking at: Do we expect to have exponential convergence generally? With a nice network, if the initial guess has the same taken paths as UE, will the corresponding JE automatically be UE, without having to use techniques like perturbation? If in an intermediate step, a path is an empty shorter path, will it be a taken path in UE? Do we have limited pieces of JE? The answers to these questions will be helpful to better understand the characteristics of J-system and lead to better solution methods.

References

M. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the economics of transportation.* Yale University Press, New Haven, Connecticut, 1956. Also published as Rand-RM-1488-PR, Rand Corporation, Santa Monica, CA, May 12, 1955.

W.-L. Jin. The dynamic system of the traffic assignment problem: Part I. Theory. 2005. Working paper UCI-ITS-TS-WP-05-01, Institute of Transportation Studies, University of California, Irvine. http://www.its.uci.edu/its/publications/papers/CTSS/UCI-ITS-TS-WP-05-01.pdf.

M. Patriksson. *The traffic assignment problem - models and methods.* VSP, Utrecht, The Netherlands, 1994.

M. Savageau and E. Voit. Recasting nonlinear differential equations as s-systems: A canonical nonlinear form. *Mathematical biosciences*, 87:83–115, 1987.

Y. Sheffi. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Prentice Hall, Englewood Cliffs, NJ, 1984.

S. H. Strogatz. *Nonlinear Dynamics and Chaos: with Applications to Physics, Biology, Chemistry, and Engineering*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1994.

L. N. Trefethen and D. Bau, III. *Numerical linear algebra*. SIAM, Philadelphia, PA, 1997.

$\mathcal{N}$	node (index) set
$\mathcal{A}$	arc (index) set
$\mathcal{R}$	set of origin nodes; $\mathcal{R} \subset \mathcal{N}$
$\mathcal{S}$	set of destination nodes; $\mathcal{S} \subset \mathcal{N}$
$\mathcal{K}_{rs}$	set of paths connecting O-D pair $r-s$; $r \in \mathcal{R}$, $s \in \mathcal{S}$
τ	decision variable, independent of time
$\Delta\tau$	decision step
x_a	flow on arc a; $\mathbf{x} = (\cdots, x_a, \cdots)$
t_a	travel time on arc a;$\mathbf{t} = (\cdots, t_a, \cdots)$
q_{rs}	traffic demand between origin-pair rs; $q_{rs} \equiv f_0^{rs}$
v_{rs}	average travel time for O-D pair $r-s$
u_{rs}	travel time function for variable demand for O-D pair $r-s$
f_k^{rs}	flow on path k connecting O-D pair $r-s$; $\mathbf{f}^{rs} = (f_0^{rs}, \cdots, f_k^{rs}, \cdots)$; $\mathbf{f} = (\cdots, \mathbf{f}^{rs}, \cdots)$
c_k^{rs}	travel time on path k connecting O-D pair $r-s$; $\mathbf{c}^{rs} = (\cdots, c_k^{rs}, \cdots)$; $\mathbf{c} = (\cdots, \mathbf{c^{rs}}, \cdots)$
$\delta_{a,k}^{rs}$	indicator variable: $\delta_{a,k}^{rs} = \begin{cases} 1 & \text{if link } a \text{ is on path } k \text{ between O-D pair } r-s \\ 0 & \text{otherwise;} \end{cases}$
	$\mathbf{\Delta}^{rs} = (\cdots, \delta_{a,k}^{rs}, \cdots)$; $\mathbf{\Delta} = (\cdots, \mathbf{\Delta}^{rs}, \cdots)$
$\mathcal{F}^{rs}$	the set of $\mathbf{f}^{rs}$ satisfying (1)
$\mathcal{F}$	the set of $\mathbf{f}$; $\mathcal{F} = \prod_{rs} \mathcal{F}^{rs}$
$\mathcal{G}^{rs}$	the set of initially non-empty $\mathbf{f}^{rs}$ satisfying (1) and $\mathbf{f}^{rs}(\tau) > \mathbf{0}$ for $\tau = 0$
$\mathcal{G}$	the set of $\mathbf{f}$; $\mathcal{G} = \prod_{rs} \mathcal{G}^{rs}$
K_{rs}	the number of initially non-empty paths connecting O-D pair $r-s$
J_k^{rs}	FIFO violation for flow on path k connecting O-D pair $r-s$;
	$\mathbf{J}^{rs} = (\cdots, J_k^{rs}, \cdots)$; $\mathbf{J} = (\cdots, \mathbf{J}^{rs}, \cdots)$
$\Vert\mathbf{J}\Vert_2$	2-norm of $\mathbf{J}(\mathbf{f})$, defined in (8)

Table 1: Network notation system

TAP	the traffic assignment problem
O-D	origin-destination
BMW	(Beckmann, McGuire, and Winsten, 1956)
FIFO	First-In-First-Out
J-functions	FIFO violation functions
J-system	an autonomous dynamic system of path flows in decision variable τ
JE	J-equilibria, steady states of J-system
UE	User equilibria
J-index	index of convergence, $\Vert\mathbf{J}\Vert_2$
FW	Frank-Wolfe

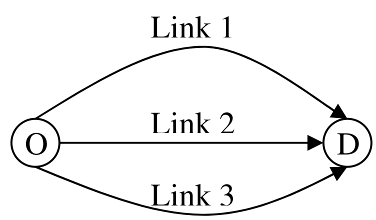
Table 2: Some abbreviations

Initial guess of $\mathbf{f}$, e.g. K-shortest paths for each O-D pair
for $n = 1, 2, 3 \cdots$
    Find JE with the finite difference method (or Newton's method)
    If no shorter empty paths, JE is UE;
    Otherwise perturb JE in the directions of shorter empty paths

Table 3: A perturbation-based method for solving UE as stable steady states of J-system

Use FW method to find a candidate state
for $n = 1, 2, 3 \cdots$
    Find JE from the state given by FW
    If no shorter empty paths, JE is UE;
    Otherwise perturb JE in the directions of shorter empty paths or use FW method again

Table 4: A hybrid method for solving UE as stable steady states of J-system



$$t_1 = 10(1 + 0.15(\frac{x_1}{2})^4)$$

$$t_2 = 10(1 + 0.15(\frac{x_2}{4})^4)$$

$$t_3 = 10(1 + 0.15(\frac{x_3}{3})^4)$$

$$x_1 + x_2 + x_3 = 10$$

Figure 1: An example network

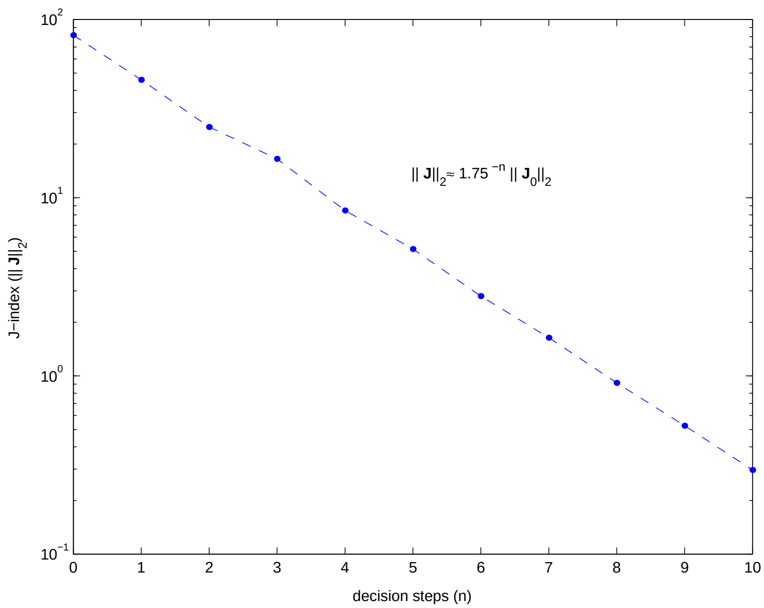


Figure 2: Convergence of Newton's method

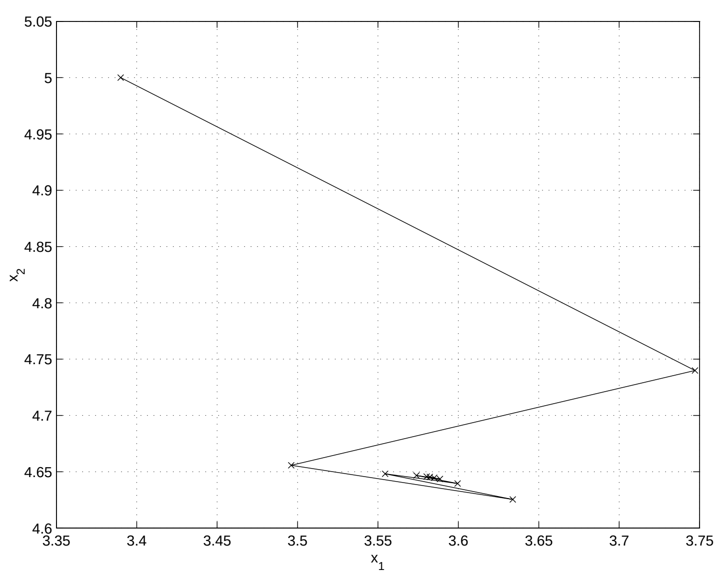


Figure 3: Solution trajectory of Newton's method

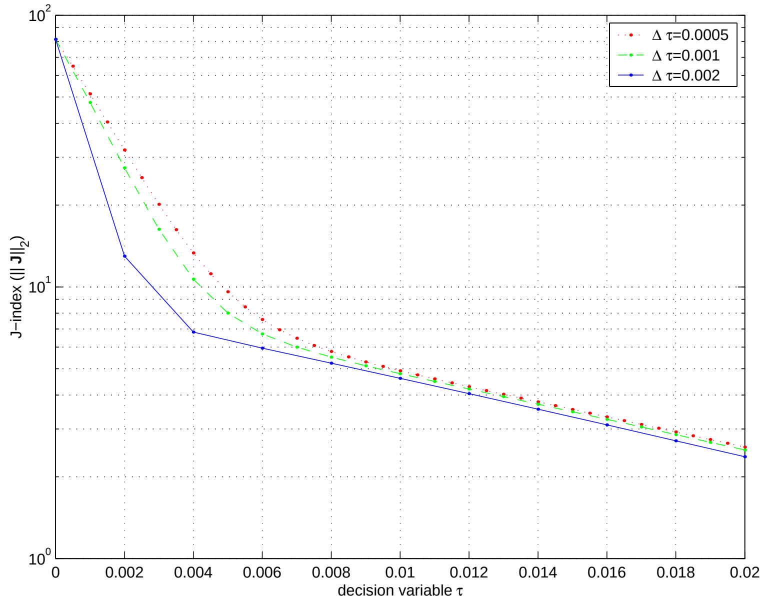


Figure 4: Convergence of the finite difference method

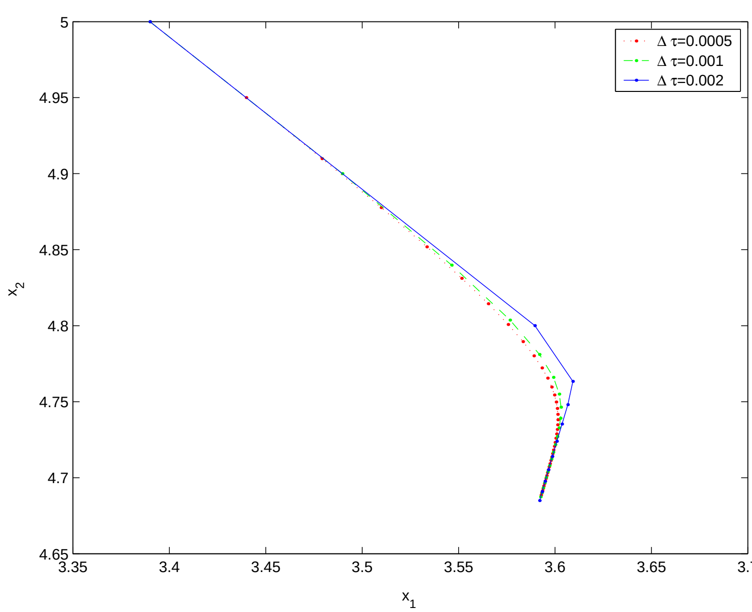


Figure 5: Solution trajectory of the finite difference method with $\tau = 0.02$

JE	link flow			link cost		
1	10	0	0	947.5000	20	25
2	0	10	0	10	137.1875	25
3	0	0	10	10	20	487.9630
4	4.0346	5.9654	0	34.8405	34.8405	25
5	4.7864	0	5.2136	59.2053	20	59.2053
6	0	6.0762	3.9238	10	35.9740	35.9740
7	3.5833	4.6451	1.7716	25.4560	25.4560	25.4560

Table 5: All JE for network in Figure 1

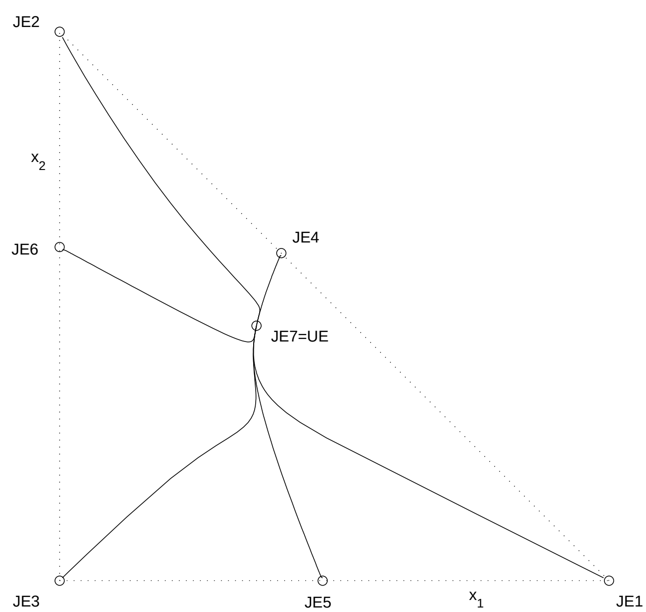


Figure 6: All JE and perturbation of non-UE JE with $\tau = 0.1$ and $\Delta\tau = 0.0005$