TOWARDS AN OPTIMAL ALGORITHM FOR COMPUTING FIXED POINTS: DYNAMICAL SYSTEMS APPROACH, WITH APPLICATIONS TO TRANSPORTATION ENGINEERING

Ruey L. Cheu
Dept. Civil Engineering
Univ. of Texas at El Paso
USA
rcheu@utep.edu

Gang Xiang
Philips Healthcare Informatics
El Paso, Texas
USA
gxiang@acm.org

Vladik Kreinovich
Dept, Computer Science
Univ. of Texas at El Paso
USA
vladik@utep.edu

Abstract

In many practical problems, it is desirable to find an equilibrium. For example, equilibria are important in transportation engineering.

Many urban areas suffer from traffic congestion. Intuitively, it may seem that a road expansion (e.g., the opening of a new road) should always improve the traffic conditions. However, in reality, a new road can actually worsen traffic congestion. It is therefore extremely important that before we start a road expansion project, we first predict the effect of this project on traffic congestion.

When a new road is built, some traffic moves to this road to avoid congestion on the other roads; this causes congestion on the new road, which, in its turn, leads drivers to go back to their previous routes, etc. What we want to estimate is the resulting equilibrium.

In many problems – e.g., in many transportation problems – natural iterations do not converge. It turns out that the convergence of the corresponding fixed point iterations can be improved if we consider these iterations as an approximation to the appropriate dynamical system.

Key words

transportation engineering, fixed point, dynamical systems

1 Many practical situations eventually reach equilibrium

In many real-life situations, we have dynamical situations which eventually reach an equilibrium.

For example, in *economics*, when a situation changes, prices start changing (often fluctuating) until they reach an equilibrium between supply and demand.

In *transportation*, when a new road is built, some traffic moves to this road to avoid congestion on the other roads; this causes congestion on the new road, which, in its turn, leads drivers to go back to their previous routes, etc. [Sheffi, 1985].

2 Specific challenges of transportation applications

Intuitively, it may seem that a road expansion (e.g., the opening of a new road) should always improve the traffic conditions. However, in reality, a new road can actually worsen traffic congestion. Specifically, if too many cars move to a new road, this road may become even more congested than the old roads initially were, and so the traffic situation will actually decrease – prompting people to abandon this new road. This possible negative effect of a new road on congestion is a very well known "paradox" of transportation science, a paradox which explains the need for a detailed analysis in the planning of the new road; see, e.g, [Sheffi, 1985; Ahuja, Magnanti, and Orlin, 1993].

3 It is often desirable to predict the corresponding equilibrium

For the purposes of the long-term planning, it is desirable to find the corresponding equilibrium. For example, for the purposes of economic planning, it is desirable to know how, in the long run, oil prices will change if we start exploring new oil fields in Alaska. For transportation planning, it is desirable to find out to what extent the introduction of a new road will relieve the traffic congestion, etc.

In order to describe how we can solve this practically important problem, let us describe this equilibrium prediction problem in precise terms.

4 Finding an equilibrium as a mathematical problem

To describe the problem of finding the *equilibrium* state(s), we must first be able to describe *all possible* states. In this paper, we assume that we already have such a description, i.e., that we know the set X of all possible states.

We must also be able to describe the fact that many states $x \in X$ are not equilibrium states. For example, if the price of some commodity (like oil) is set up too high, it will become profitable to explore difficult-to-extract oil fields; as a new result, the supply of oil will increase, and the prices will drop.

Similarly, as we have mentioned in the main text, if too many cars move to a new road, this road may become even more congested than the old roads initially were, and so the traffic situation will actually decrease – prompting people to abandon this new road.

To describe this instability, we must be able to describe how, due to this instability, the original state x gets transformed in the next moment of time. In other words, we assume that for every state $x \in X$, we know the corresponding state $f(x)$ at the next moment of time.

For non-equilibrium states x, the change is inevitable, so we have $f(x) \neq x$. The equilibrium state x is the state which does not change, i.e., for which $f(x) = x$. Thus, we arrive at the following problem: We are given a set X and a function $f : X \to X$; we need to find an element x for which $f(x) = x$.

In mathematical terms, an element x for which $f(x) = x$ is called a *fixed point* of the mapping f. So, there is a practical need to find fixed points.

5 Fixed points in transportation engineering: static case

To describe traffic, we divided the urban area into zones. For every two zones i and j, we find the number of drivers d_{ij} who need to go from zone i to zone j. For each road link, we find the road capacity c – the number c of cars per hour which can pass through this road link. The travel time t along a link of length L with a speed limit s is usually described by the Bureau of Public Roads (BPR) formula

$$t = t^f \cdot \left[1 + a \cdot \left(\frac{v}{c}\right)^\beta\right], \qquad (1)$$

where v is a volume along this link, $t^f = L/s$ is a *free-flow time*, $a \approx 0.15$, and $\beta \approx 4$.

When a new road is built, some traffic moves to this road to avoid congestion on the other roads; this causes congestion on the new road, which, in its turn, leads drivers to go back to their previous routes, etc. These changes continue until there are alternative routes in which the overall travel time is larger.

Eventually, this process converges to an *equilibrium*, i.e., to a situation in which the travel time along all used alternative routes is exactly the same – and the travel times along other un-used routes is higher; see, e.g., [Sheffi, 1985].

There exist efficient algorithms which, given the traffic demand (i.e., the values d_{ij}) and the road capacity, computes the corresponding equilibrium [Sheffi, 1985]. This algorithm computes the traffic volume along each road link, the travel time between every two zones, etc.

6 Fixed points in transportation engineering: dynamic case

To apply the above algorithm, we must know, at each time interval t, the values $d_{ij}(t)$. The problem is that if we build a new road, these values may change. For example, if the driver needs to be at work at 8:00am, and the travel time to his or her destination is 30 minutes, then the driver leaves at 7:30 am. However, if a new freeway decreases the expected travel time to 15 minutes, then the driver will leave at 7:45 am instead of the original 7:30 am. In this case, the corresponding value $d_{ij}(7{:}30)$ decreases while $d_{ij}(7{:}45)$ increases. We need to take this choice of departure time into account.

In transportation engineering, the most widely used model for describing the general choice (especially the choice in transportation-related situations) is the *logit* model. In the logit model, the probability of departure in different time intervals is determined by the utility of different departure times to the driver. According to this model, the probability P_t that a driver will choose the t-th time interval is proportional to $\exp(u_t)$, where u_t is the expected utility of selecting this time interval. The coefficient at $\exp(u_t)$ must be chosen from the requirement that the sum of these probabilities be equal to 1. So, the desired probability has the form $P_t = \exp(u_t)/s$, where $s \stackrel{\text{def}}{=} \exp(u_1) + \ldots + \exp(u_n)$.

To apply the logit model, we must be able to estimate the utilities of different departure time choices. According to [Noland and Small, 1995], the utility u_i of choosing the i-th time interval is determined by the following formula:

$$u_i = -0.1051 \cdot E(T) - 0.0931 \cdot E(SDE)$$

$$-0.1299 \cdot E(SDL) - 1.3466 \cdot P_L - 0.3463 \cdot \frac{S}{E(T)}, \qquad (2)$$

where $E(T)$ is the expected value of travel time T, $E(SDE)$ is the expected value of the wait time SDE when arriving early, $E(SDL)$ is the expected value of the delay SDL when arriving late, P_L is the probability of arriving late, and S is the variance of the travel time.

Let us assume that we know the original traffic pattern x, i.e., the values $d_{ij}(t)$ for each time interval t.

For each time interval t, we can solve the traffic assignment problem corresponding to this time interval. From the resulting traffic assignment, we can compute the values of the desired auxiliary characteristics, and thus, estimate the expected utility u_t of departing at this time interval t. The logit formula $P_t = \exp(u_t)/s$ enables us to compute the probability P_t that the driver will actually select departure time interval t.

The probability P_t means that out of N drivers who travel from the given origin zone to the given destination zone, $N \cdot P_t$ leave during the t-th time interval. The overall number of drivers who leave from the given origin zone to the given destination zone can be computed by adding the corresponding values for all time intervals. Multiplying this sum by P_t, we get the new value. These new values form the new traffic pattern $f(x)$. The question is to find the equilibrium pattern x for which $x = f(x)$.

7 The problem of computing fixed points

Since there is a practical need to compute the fixed points, let us give a brief description of the existing algorithms for computing these fixed points. Readers interested in more detailed description can look, e.g., in [Berinde, 2002].

8 Straightforward algorithm: Picard iterations

At first glance, the situation seems very simple and straightforward. We know that if we start with a state x at some moment of time, then in the next moment of time, we will get a state $f(x)$. We also know that eventually, we will get an equilibrium. So, a natural thing to do is to simulate how the actual equilibrium will be reached.

In other words, we start with an arbitrary (reasonable) state x_0. After we know the state x_k at the moment k, we predict the state x_{k+1} at the next moment of time as $x_{k+1} = f(x_k)$. This algorithm is called *Picard iterations* after a mathematician who started efficiently using it in the 19 century.

If the equilibrium is eventually achieved, i.e., if in real life the process converges to an equilibrium point x, then Picard's iterations are guaranteed to converge. Their convergence may be somewhat slow – since they simulate all the fluctuations of the actual convergence – but eventually, we get convergence.

9 Situations when Picard's iterations do not converge: economics

In some important practical situations, Picard iterations do not converge.

The main reason is that in practice, we can have panicky fluctuations which prevent convergence. Of course, one expects fluctuations. For example, if the price of oil is high, then it will become profitable for companies to explore and exploit new oil fields. As a result, the supply of oil will drastically increase, and the price of oil will go down. Since this is all done in a unplanned way, with different companies making very rough predictions, it is highly probable that the resulting oil supply will exceed the demand. As a result, prices will go down, oil production in difficult-to-produce oil areas will become unprofitable, supply will go down, etc.

Such fluctuations have happened in economics in the past, and sometimes, not only they did not lead to an equilibrium, they actually led to deep economic crises.

10 Situations when Picard's iterations do not converge: transportation

Similar situations happen in transportation as well. Indeed, let us assume that x is a current traffic. Based on the current traffic, we will have some congestions and traffic delay. As a result, at the next moment, the drives choose different routes, different departure times etc. – to avoid the delays experienced the day before. This next traffic pattern will be denoted as $f(x)$. Our objective is to find out the equilibrium state, i.e., the state for which $f(x) = x$.

A seemingly natural idea it to start with some traffic pattern x_0, then observe the pattern at the next day $x_1 = f(x_0)$, the pattern two days later $x_2 = f(x_1)$, etc., until these patterns converge. In practice, however, this sequence may not converge.

This lack of convergence can be illustrated on a "toy" example in which we have a single origin, single destination, and two possible departure times. Let us assume that the work starts at 8 am, that the free-flow traffic time is 30 minutes, and that we consider two possible departure times 7:30 am and 7:15 am.

For simplicity, let us consider the situation when the traffic suddenly increases. In the original traffic pattern x_0, congestion was low, so practically everyone could reach the destination in 0 minutes. In this original situation, everyone leaves at 7:30 am and nobody leaves at 7:15 am.

- For those departing at 7:15 am, there was no traffic, so the travel time was equal to the free-flow time of 30 minutes.
- The drivers departing at 7:30 am face a much heavier traffic, so we get a traffic congestion. As a result of this congestion, the travel time increased to 45 minutes.

So:

- drivers who leave at 7:15 am spend only 30 minutes in traffic and arrive 15 minutes early, while
- drivers who leave at 7:30 am spend 45 minutes on the road and are 15 minutes late.

It is reasonable to assume that the penalties for being late are heavy, while penalties for arriving early are much lighter. As a result, the next day, almost everyone leaves at 7:15 am and practically no one leaves at 7:30 am.

In this new traffic pattern x_1:

- for those departing at 7:30 am, there is no traffic, so the travel time is equal to the free-flow time of 30 minutes;
- the drivers departing at 7:15 am face a much heavier traffic, so we get a traffic congestion; as a result of this congestion, the travel time increases to 45 minutes.

So:

- drivers who leave at 7:30 am spend only 30 minutes in traffic and arrive on time, while
- drivers who leave at 7:15 am spend 45 minutes on the road.

Since we assumed that there is a penalty for spending extra time on the road, in the next moment of time, we are back to the original arrangement x. These "flip-flop" changes continue without any convergence.

11 How can we handle these situation: a natural practical solution

If the natural Picard iterations do not converge, this means that in practice, there is too much of a fluctuation. When at some moment k, the state x_k is not an equilibrium, then at the next moment of time, we have a state $x_{k+1} = f(x_k) \neq x_k$. However, this new state x_{k+1} is an not necessarily closer to the equilibrium: it "over-compensates" by going too far to the other side of the desired equilibrium.

For example, we started with a price x_k which was too high. At the next moment of time, instead of having a price which is closer to the equilibrium, we may get a new price x_{k+1} which is too low – and may even be further away from the equilibrium than the previous price.

In practical situations, such things do happen. In this case, to avoid such weird fluctuations and to guarantee that we eventually converge to the equilibrium point, a natural thing is to "dampen" these fluctuations: we know that a transition from x_k to x_{k+1} has gone too far, so we should only go "halfway" (or even smaller piece of the way) towards x_{k+1}.

How can we describe it in natural terms? In many practical situations, there is a reasonable linear structure on the set X on all the states, i.e., X is a linear space. In this case, going from x_k to $f(x_k)$ means adding, to the original state x_k, a displacement $f(x_k) - x_k$. Going halfway would then mean that we are only adding a half of this displacement, i.e., that we go from x_k to $x_{k+1} = x_k + \frac{1}{2} \cdot (f(x_k) - x_k)$, i.e., to

$$x_{k+1} = \frac{1}{2} \cdot x_k + \frac{1}{2} \cdot f(x_k). \qquad (3)$$

The corresponding iteration process is called *Krasnoselskii iterations*. In general, we can use a different portions $\alpha \neq 1/2$, and we can also use different portions α_k on different moments of time. In general, we thus go from x_k to $x_{k+1} = x_k + \alpha_k \cdot (f(x_k) - x_k)$, i.e., to

$$x_{k+1} = (1 - \alpha_k) \cdot x_k + \alpha_k \cdot f(x_k). \qquad (4)$$

These iterations are called *Krasnoselski-Mann iterations*.

12 Practical problem: the rate of convergence drastically depends on α_i

The above convergence results show that under certain conditions on the parameters α_i, there is a convergence. From the viewpoint of guaranteeing this convergence, we can select any sequence α_i which satisfies these conditions. However, in practice, different choice of α_i often result in drastically different rate of convergence.

To illustrate this difference, let us consider the simplest situation when already Picard iterations $x_{n+1} = f(x_n)$ converge, and converge monotonically. Then, in principle, we can have the same convergence if instead we use Krasnoselski-Mann iterations with $\alpha_n = 0.01$. Crudely speaking, this means that we replace each original step $x_n \to x_{n+1} = f(x_n)$, which bring x_n directly into x_{n+1}, by a hundred new smaller steps. Thus, while we still have convergence, we will need 100 times more iterations than before – and thus, we require a hundred times more computation time.

Since different values α_i lead to different rates of convergence, ranging from reasonably efficient to very inefficient, it is important to make sure that we select *optimal* values of the parameters α_i, values which guarantee the fastest convergence.

13 First idea: from the discrete iterations to the continuous dynamical system

In this section, we will describe the values α_i which are optimal in some reasonable sense. To describe this sense, let us go back to our description of the dynamical situation. In the above text, we considered observations made at discrete moments of time; this is why we talked about current moment of time, next moment of time, etc. In precise terms, we considered moments t_0, $t_1 = t_0 + \Delta t$, $t_2 = t_0 + 2\Delta t$, etc.

In principle, the selection of Δt is rather arbitrary. For example, in terms of prices, we can consider weekly prices (for which Δt is one week), monthly prices, yearly prices, etc. Similarly, for transportation, we can consider daily, hourly, etc. descriptions. The above discrete-time description is, in effect, a discrete approximation to an actual continuous-time system.

Similarly, Krasnoselski-Mann iterations $x_{k+1} - x_k = \alpha_k \cdot (f(x_k) - x_k)$ can be viewed as a discrete-time approximations to a continuous dynamical system which leads to the desired equilibrium. Specifically, the difference $x_{k+1} - x_k$ is a natural discrete analogue of the derivative $\frac{dx}{dt}$, the values α_k can be viewed as discretized values of an unknown function $\alpha(t)$, and so the corresponding continuous system takes the form

$$\frac{dx}{dt} = \alpha(t) \cdot (f(x) - x). \quad (5)$$

A discrete-time system is usually a good approximation to the corresponding continuous-time system. Thus, we can assume that, vice versa, the above continuous system is a good approximation for Krasnoselski-Mann iterations.

In view of this fact, in the following text, we will look for an appropriate (optimal) continuous-time system (5).

14 Scale invariance: natural requirement on a continuous-time system

In deriving the continuous system (5) from the formula for Krasnoselski-Mann iterations, we assumed that the original time interval Δt between the two consecutive iterations is 1. This means, in effect, that to measure time, we use a scale in which this interval Δt is a unit interval.

As we have mentioned earlier, the choice of the time interval Δt is rather arbitrary. If we make a different choice of this discretization time interval $\Delta t' \neq \Delta t$, then we would get a similar dynamical system, but described in a different time scale, with a different time interval $\Delta t'$ taken as a measuring unit. As a result of "de-discretizing" this new system, we would get a different continuous system of type (5) – a system which differs from the original one by a change in scale.

In the original scale, we identified the time interval Δt with 1. Thus, the time t in the original scale means physical time $T = t \cdot \Delta t$. In the new scale, this same physical time corresponds to the time

$$t' = \frac{T}{\Delta t'} = t \cdot \frac{\Delta t}{\Delta t'}. \quad (6)$$

If we denote by $\lambda = \frac{\Delta t'}{\Delta t}$ the ratio of the corresponding units, then we conclude that the time t in the original scale corresponds to the time $t' = t/\lambda$ in the new scale. Let us describe the system (5) in terms of this new time coordinate t'. From the above formula, we conclude that $t = \lambda \cdot t'$; substituting $t = \lambda \cdot t'$ and $dt = \lambda \cdot dt'$ into the formula (5), we conclude that

$$\frac{1}{\lambda} \cdot \frac{dx}{dt'} = \alpha(\lambda \cdot t') \cdot (f(x) - x), \quad (7)$$

i.e., that

$$\frac{dx}{dt'} = (\lambda \cdot \alpha(\lambda \cdot t')) \cdot (f(x) - x). \quad (8)$$

It is reasonable to require that the optimal system of type (5) should not depend on what exactly time interval Δt we used for discretization.

15 Conclusion: optimal Krasnoselski-Mann iterations correspond to $\alpha_k = c/k$

Since a change of the time interval corresponds to re-scaling, this means the system (5) must be scale-invariant, i.e., to be more precise, the system (8) must have exactly the same form as the system (5) but with t' instead of t, i.e., the form

$$\frac{dx}{dt'} = \alpha(t') \cdot (f(x) - x). \quad (9)$$

By comparing the systems (8) and (9), we conclude that we must have

$$\lambda \cdot \alpha(\lambda \cdot t') = a(t') \quad (10)$$

for all t' and λ. In particular, if we take $\lambda = 1/t'$, then we get $\alpha(t') = \frac{\alpha(1)}{t'}$, i.e., $\alpha(t') = c/t'$ for some constant c $(= \alpha(1))$.

With respect to the corresponding discretized system, this means that we take $\alpha_k = \alpha(k) = c/k$.

Comment. The formula $\alpha_k = c/k$ is not exact: it comes form approximating the actual continuous dependence by a discrete one. This approximation makes asymptotic sense, but this formula cannot be applied for $k = 0$. To make this formula applicable, we must start with $k = 1$ – or, equivalently, start with $k = 0$ (since this is how most descriptions of iterations work), but use the expression $\alpha_k = c/(k+1)$ instead.

16 Reasonable choice of the constant c

As we have mentioned, a reasonable idea is to use Picard iterations. This is not always a good idea, because we may get wild fluctuations. However, it makes some sense to start with the Picard iteration first, to get away from the initial state.

Picard iterations correspond to $\alpha_k = 1$; so, if we want $\alpha_0 = 1$, i.e., $c/(0+1) = 1$, we must take $c = 1$. The resulting iterations take the form

$$x_{k+1} = \left(1 - \frac{1}{k+1}\right) \cdot x_k + \frac{1}{k+1} \cdot f(x_k). \quad (11)$$

17 Selection of $\alpha_k = 1/(k+1)$ in the general case: commonsense interpretation

The above formula (corresponding to $c = 1$) has a natural commonsense interpretation.

Namely, in Picard iterations, as a next iteration x_{k+1}, we take $f(x_k)$. When there are wild oscillations, these iterations do not converge. We expect, however, that these oscillations are going on around the equilibrium point. So, while the values x_i are oscillating and not converging at all, their averages $\frac{x_0 + \ldots + x_k}{k+1}$ and the corresponding values $\frac{f(x_0) + \ldots + f(x_k)}{k+1}$ will be getting closer and closer to the desired equilibrium. Thus, if we want to enhance convergence, then, instead of taking $f(x_k)$ as the next iteration, it makes sense to take an *average* of the previous values of $f(x_k)$:

$$x_{k+1} = \frac{f(x_0) + \ldots + f(x_{k-1}) + f(x_k)}{k+1}. \quad (12)$$

Let us show that this idea leads exactly to our choice $\alpha_k = 1/(k+1)$. Indeed, from

$$x_k = \frac{f(x_0) + \ldots + f(x_{k-1})}{k}, \quad (13)$$

we conclude that

$$f(x_0) + \ldots + f(x_{k-1}) = k \cdot x_k, \quad (14)$$

hence

$$f(x_0)+\ldots+f(x_{k-1})+f(x_k) = k \cdot x_k + f(x_k) \quad (15)$$

and thus,

$$x_{k+1} = \frac{f(x_0) + \ldots + f(x_k)}{k+1} = \frac{k \cdot x_k + f(x_k)}{k+1} =$$

$$\left(1 - \frac{1}{k+1}\right) \cdot x_k + \frac{1}{k+1} \cdot f(x_k). \quad (16)$$

18 Selection of $\alpha_k = 1/(k+1)$ in transportation problems: commonsense interpretation

For transportation problems, this interpretation can be made even more natural. Indeed, when deciding on the best route and on the best departure time x_{k+1} for the moment $k+1$, a reasonable driver takes into account not only the traffic pattern x_k at the previous day, but also the traffic patterns at all previous days. A reasonable idea is to base traffic decisions on the average past traffic, i.e., on $e_k \stackrel{\text{def}}{=} \frac{1}{k} \cdot \sum_{i=1}^{k} x_i$. In other words, instead of the previously considered divergent iterations $x_{k+1} = f(x_k)$, we should consider new iterations $x_{k+1} = f(e_k)$.

This formula leads to the new average

$$e_{k+1} = \frac{1}{k+1} \cdot \sum_{i=1}^{k+1} x_i = \frac{1}{k+1} \cdot \left(\sum_{i=1}^{k} x_i + x_{k+1}\right) =$$

$$\frac{1}{k+1} \cdot (k \cdot e_k + x_{k+1}). \quad (17)$$

Since $x_{k+1} = f(e_k)$, we thus conclude that

$$e_{k+1} = \left(1 - \frac{1}{k+1}\right) \cdot e_k + \frac{1}{k+1} \cdot f(e_k), \quad (18)$$

which is exactly the optimal iteration process.

19 This selection works well

Our experiments on the "toy" road network and on the actual El Paso road network confirmed that this procedure converges [Cheu et al., 2007; Cheu et al., 2008].

The choice $a_k = 1/k$ have been successfully used in other applications as well; see, e.g., [Su and Qin, 2006] and references therein.

Acknowledgments

This work was supported in part by NSF grants HRD-0734825, EAR-0225670, and EIA-0080940, by Texas Department of Transportation grant No. 0-5453, by the Japan Advanced Institute of Science and Technology (JAIST) International Joint Research Grant 2006-08, and by the Max Planck Institut für Mathematik.

References

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993) *Network Flows: Theory, Algorithms and Applications*, Prentice-Hall, Englewood Cliffs, NJ.

Berinde, V. (2002) *Iterative approximation of fixed points*, Editura Efemeride, Baia Mare.

Cheu, R., Kreinovich, V., Chiu, Y.-C., Pan, R., Xiang, G., Bhupathiraju, S. V., and Manduva, S. R. (2007) *Strategies for Improving Travel Time Reliability*, Texas Department of Transportation, Research Report 0-5453-R2.

Cheu, R., Kreinovich, V., Modave, F., Xiang, G., and Magoc, T. (2008) How to estimate, take into account, and improve travel time reliability in transportation networks, In: Muhanna, R. L. and Mullen, R. L. (eds.) *Proceedings of the International Workshop on Reliable Engineering Computing REC'08*, Savannah, Georgia, February 20–22, pp. 289–332.

Noland, R. B. and Small, K. A. (1995) Travel time uncertainty, departure time choice, and the cost of morning commutes, *Transportation Researh Record*, **1493**, pp. 150–158.

Sheffi, Y. (1985) *Urban Transportation Networks*, Prentice Hall, Englewood Cliffs, NJ.

Su, Y. and Qin, X. (2006) Strong convergence theorems for asymptotically nonexpansive mappings and asymptotically nonexpansive semigroups. *Fixed Point Theory and Applications*, Article ID 96215, pp. 1–11.