

# User-Equilibrium Properties of Fixed Points in Dynamic Traffic Assignment\*

David E. Kaufman<sup>†</sup>  
AT&T Labs

Robert L. Smith<sup>‡</sup>  
University of Michigan

Karl E. Wunderlich<sup>§</sup>  
Mitretek Systems, Inc.

December 1, 1997

## Abstract

This paper considers the problem of dynamic traffic assignment under the principle that individual drivers will choose fastest paths, in the dynamic situation where path durations consist of time-dependent link travel times. Rather than constructing a unified model encompassing traffic dynamics and route choice, we decompose the model into an assignment mapping, which identifies the link travel times resulting from an input routing policy, and a routing mapping, which yields fastest-path routings associated with input link travel times. Since time-dynamic link travel times are influenced by route choice, this dynamic situation therefore encompasses predictive routing strategies.

We establish that user-equilibrium routing policies are fixed points of the composition of the routing and assignment functions. After discussing difficulties associated with establishing existence of fixed points under discrete-time modeling and all-or-nothing routing, we present instead new iterative routing mappings for continuous-time multipath routing (the splitting of a single-class flow onto multiple paths), which adjust routing policies more incrementally. We provide sufficient conditions for existence of fixed points in various routing policy domains and offer some suggestions on the computation of these fixed-point policies.

---

\*This research was supported in part by the University of Michigan ITS Center.

<sup>†</sup>Business Operations Analysis, AT&T, Room 2B19, 379 Campus Drive, Somerset, NJ 08873

<sup>‡</sup>Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109-2117

<sup>§</sup>Mitretek Systems, Inc., 600 Maryland Avenue, SW, Suite 755, Washington, DC 20024

# 1 Introduction

An assignment of flow in a traffic network is described as satisfying user equilibrium if no vehicle can alter its route choice so that its trip duration is reduced. This condition models drivers behaving noncooperatively with perfect information about the available route choices. When traffic conditions are considered constant over time, the situation is one of static user equilibrium, which has been studied extensively (see, e.g., Dafermos (1980), Potts and Oliver (1972), Sheffi (1985), and Patriksson (1994)). On the other hand, the problem of dynamic user equilibrium, where traffic conditions change over time, has a variety of formulations and a lack of unifying results.

Classical approaches try to construct a single mathematical model including route choices as decision variables, which are used to compute link travel times and corresponding trip durations. The model is typically given variational constraints and/or an objective function intended to force the trip durations to satisfy user equilibrium. Models proposed in the literature include Janson (1991), Ghali and Smith (1995), and Wie et al (1995) in discrete time and Friesz et al (1993), Ran et al (1993), Smith and Wisten (1995), and Lam and Huang (1995) in continuous time. However, these models suffer from restrictive definitions of equilibrium and conditions on traffic flow dynamics and/or failure to establish existence of equilibrium solutions. The issue of the proper model to realistically capture dynamic traffic flow remains a critical problem that we also do not definitively address in this paper.

As in Janson (1995) and Jayakrishnan et al (1995), we decompose the classical approach into two parts. We consider *assignment mappings*, which give link travel times and trip durations that result from fixed vehicle routings. These assignment mappings are not assumed to follow from any closed-form relationship between traffic flows and link travel times. We also construct *routing mappings*, which determine user-equilibrium routings given fixed link travel times. The resulting routings are not user equilibrated, in general, because they are based on input link travel times that may not recur under the new routings. However, if the new routings would exactly reproduce the input link travel times through the assignment map, then the routings do satisfy user equilibrium. Such routings are described as *fixed points* of the combined routing/assignment mapping. This characterization is used to investigate existence of user-equilibrium solutions through fixed-point theory, considering effects of modeling in discrete or continuous time and with all-or-nothing or multipath routing. An computer implementation of the all-or-nothing routing version together with numerical experience is documented in Wunderlich, Kaufman and Smith (1997).

## 2 All-or-nothing routing

Let the traffic network be  $G = (N, L)$ , where  $N$  is the set of nodes,  $D \subset N$  the set of destinations, and  $L \subset N \times N$  the set of links. With  $\mathfrak{R}_+$  denoting the nonnegative reals, let

$T \subset \mathfrak{R}_+$  be the set of time indices at which route decisions can be made, i.e., the times at which the model permits vehicles to depart nodes. A function  $\pi : N \times D \times T \rightarrow L$  will be called an *all-or-nothing* routing policy, because under such a policy, all the traffic at node  $n \in N$  at time  $t \in T$  with destination  $d \in D$  must choose  $\pi(n, d, t)$  as its next link. (Routings that split the traffic at  $(n, d, t)$  onto two or more next links are called *multipath* policies; see section 3.) We require that policies be feasible in the sense that any link  $\pi(n, d, t)$  is in fact a link  $(n, n') \in L$  outgoing from  $n$ . Denote by  $\Pi$  the set of feasible all-or-nothing policies.

We call  $c : L \times T \rightarrow \mathfrak{R}_+$  a *travel time function*;  $c(\ell, t)$  will be interpreted as the link travel time experienced by a vehicle entering link  $\ell$  at time  $t$ . We require that travel time functions be feasible in the sense that  $t + c(\ell, t) \in T$  for all  $t \in T, \ell \in L$ , i.e., that a vehicle which enters a link at a time in  $T$  will exit the link at a time in  $T$ . It is assumed that any delay at a node will be modeled as part of the preceding link travel time. Denote by  $C$  the set of feasible link travel time functions.

We consider a model of traffic flows in a vehicular network simply as a function  $A : \Pi \rightarrow C$ , called an *assignment mapping*;  $c = A(\pi)$  gives the link travel times that would occur in the traffic network  $G$  when all vehicles in the network are routed according to policy  $\pi$ . The time-dependent travel demands in the network are assumed to be represented as part of the definition of a particular assignment mapping. Note that this permits flexible departure-time choice.

We also introduce  $R : C \rightarrow \Pi$ , called the *routing mapping*, which computes as  $\pi = R(c)$  the time-dependent fastest paths in the network under link travel times  $c$ . The composition of the functions, the combined routing/assignment mapping  $R(A(\cdot)) : \Pi \rightarrow \Pi$ , can be seen as mapping an initial routing policy to a new policy (as illustrated in Figure 1) which would appear optimal to a single vehicle, assuming all other vehicles in the network were following the initial policy. A routing policy satisfying  $R(A(\pi)) = \pi$  will be called a *fixed point* of the routing/assignment mapping.

**Insert Figure 1 about here.**

The problem of calculating time-dependent fastest paths, i.e., evaluating  $R$ , has been explored in Kaufman and Smith (1993), where it was shown that under an intuitive consistency condition, finding the fastest paths for trips starting at any fixed time  $t$  requires no additional computational effort beyond static shortest-path calculation. The nature of the assignment mapping  $A$  is very general; our definition is intended to encompass everything from deterministic mathematical models based on link impedance functions (e.g., Janson (1991), Kaufman, Nonis, and Smith (1996), and Lafortune et al (1994)) to deterministic traffic simulations (e.g., Santiago and Rathi (1990), Van Aerde et al (1989), and Wong (1990)).

We now make formal the characterization of user-equilibrium solutions as fixed points. In the following definition, we refer to  $\pi$  as determining paths, since  $\pi$  and the resulting

travel times  $c = A(\pi)$  determine the complete path from  $n$  to  $d$  at time  $t$  as consisting of the links  $\ell_1 = (n, n') = \pi(n, d, t), \ell_2 = (n', n'') = \pi(n', d, t + c(\ell_1, t))$ , and so on.

**Definition 1** A policy  $\pi \in \Pi$  satisfies *user equilibrium* if, when all vehicles follow  $\pi$ , the ensuing link travel times are such that the path determined by  $\pi$  from any node  $n \in N$  starting at any time  $t \in T$  to any destination  $d \in D$  has duration less than or equal to that of all alternative paths.

This definition takes into account the actual path durations experienced by vehicles which enter the links of the path at various times, like the simultaneous route-departure equilibrium of Friesz et al (1993), rather than the instantaneous equilibrium of Ran et al (1993) which only considers path durations consisting of the link travel times prevailing at the time the trip is begun.

**Theorem 1** *A fixed-point routing policy satisfies user equilibrium.*

**Proof:** We prove the contrapositive. Suppose  $\pi \in \Pi$  does not satisfy user equilibrium. Then under the link travel times  $A(\pi)$  which prevail in the network under  $\pi$ , for some  $n \in N$ ,  $t \in T$ ,  $d \in D$  the trip duration from  $n$  to  $d$  at time  $t$  choosing link  $\ell = \pi(n, d, t)$  leaving  $n$  is greater than the trip duration choosing a different link  $\ell'$  leaving  $n$ . But since  $R$  determines the fastest-path routing under link travel times  $A(\pi)$ , then  $\pi' = R(A(\pi))$  implies  $\pi'(n, d, t) = \ell' \neq \ell = \pi(n, d, t)$ , and thus  $\pi$  is not a fixed point. ■

The theorem motivates us to seek user-equilibrium solutions by seeking fixed points, but does not specify how to compute fixed points. One simple approach would be to execute repeatedly the loop implied by Figure 1. Explicitly, choose an arbitrary initial routing policy  $\pi_0$ , set  $k = 1$ , and proceed iteratively by the steps  $c_k = A(\pi_{k-1}); \pi_k = R(c_k)$ ; increment  $k$ . The iteration terminates when  $\pi_k = \pi_j$  for some  $j < k$ . The iterative method has been applied in Kaufman, Wunderlich and Smith (1991) and Wunderlich, Kaufman and Smith (1997) under the acronym SAVaNT (Simulation of Anticipatory Vehicle Network Traffic), using the INTEGRATION traffic simulation Van Aerde (1989) as the assignment mapping. In a small test network, the iteration frequently terminated with  $j = k - 1$ , i.e., with a fixed point  $\pi_k$ , where  $k$  was typically 10 iterations or less. However, in some experiments,  $j < k - 1$ ; i.e., the method encountered cycling without finding a fixed point, and thus no user-equilibrium solution was found.

Within the generality of the model defined above, existence of fixed points under all-or-nothing routing cannot be guaranteed. In the discrete-time case where  $T$  consists of integers  $1, \dots, t_0$  this is demonstrated trivially by an example consisting only of two nodes and two identical links from node 1 to node 2, where a vehicle platoon wishes to travel from 1 to 2 at time zero; see Figure 2a.

Insert Figure 2 about here.

Only two routing policies exist, one choosing the first link and the other choosing the second. Under the common assumption that link travel times are an increasing function of the load on the link, both policies have the property that the link not chosen appears to give a shorter trip duration, and hence the SAVaNT iteration would cycle between the two policies, neither being a fixed point. The cause of the existence failure is twofold. First is the essential discreteness of all-or-nothing routing, which prevents us from splitting the vehicle platoon to obtain flow balance. This motivates our investigation of multipath routing, which follows. Second, our definition of user equilibrium is somewhat restrictive. In the example of Figure 2a, neither policy is a fixed point because the unused path always appears faster. But if the platoon is allowed to switch from one policy to the other, the travel time experienced by the platoon is unaffected. Both policies would therefore satisfy a weaker definition of user equilibrium, encompassing policies that allow alternate paths to be faster if the time savings vanishes when the route is changed, as in the example. However, the weaker definition would require a more general assignment mapping, producing link travel times corresponding to a multiplicity of “what-if” alternate-route scenarios based on nonincremental reassignments of flow.

It might seem that we could get around the discreteness of all-or-nothing routing by choosing a time interval small enough to allow us to achieve flow balance. The general expression of this approach is the idea of *bang-bang* continuous-time control, in which the time index set  $T$  is an interval  $[0, t_0]$ , partitioned into a number of subintervals. The admissible bang-bang routing policies are those that are all-or-nothing on each of the subintervals of the partition. However, even this generalization is not sufficient to guarantee existence of fixed points. To see this, reconsider the network of our previous example. Let the first subinterval of the partition be  $[0, t_1]$ . Under a bang-bang policy, the route choice must be the same during all of  $[0, t_1]$ , as in Figure 2b. Then no matter how small  $t_1$  is made, the other link appears faster at time  $t_1/2$  and thus the policy is not a fixed point. To generalize further, we must include policies that are not necessarily piecewise constant, leading again to our treatment of multipath routing, which follows.

### 3 Multipath routing

Where route choice has previously been specified as a next link  $\ell \in L$ , it will now be given as a vector of fractional splits. To simplify the notation, assume that  $G$  has exactly  $k$  links departing each node. The results which follow will obviously extend to networks with varying numbers of outgoing links from the nodes. Let

$$\bar{L}_k = \left\{ v = (v_1, \dots, v_k) \in \mathfrak{R}_+^k : v \geq 0, v_1 + \dots + v_k = 1 \right\}$$

be the space of routing split vectors; if  $v = (v_1, \dots, v_k)$  is the routing split vector to be applied to a flow of  $x$  vehicles at node  $n$  at time  $t$  with destination  $d$ , then for  $i = 1, \dots, k$  the destination- $d$  flow onto the  $i$ th link away from  $n$  at time  $t$  is  $v_i x$ . Traffic flows are thus considered to be infinitely divisible.

Define the set of multipath policies by  $\bar{\Pi} = \{\bar{\pi} : N \times D \times T \rightarrow \bar{L}_k\}$ . A multipath policy gives a routing split vector for each decision point  $(n, d, t)$ .  $\bar{\Pi}$  is a multidimensional continuum to which we seek to apply fixed-point theory to conclude existence of fixed points and hence user-equilibrium routing policies.

The property most valuable in establishing existence of fixed points is continuity of the mapping, in this case from  $\bar{\Pi}$  to itself. Now recall our earlier requirement that travel time functions  $c \in C$  be feasible, i.e., that they take on values which ensure that vehicles reach nodes only at times in the index set  $T$ . If  $T$  is discrete, it is apparent that feasibility is incompatible with continuity. Therefore we will confine our attention to the continuous-time case, taking  $T$  to be an interval  $[0, t_0]$  of the real line.

In this section we discuss how to formulate routing mappings so that the combined routing/assignment function is continuous. It is proven that for a broad class of routing mappings, a fixed point satisfies user equilibrium, under the following generalization of Definition 1:

**Definition 2** A policy  $\bar{\pi} \in \bar{\Pi}$  satisfies *user equilibrium* if, when all flows are split at nodes according to  $\bar{\pi}$ , the ensuing link travel times are such that the various paths determined by  $\bar{\pi}$  from a given node  $n \in N$  starting at a given time  $t \in T$  to a given destination  $d \in D$  have equal duration, and all alternative paths not employed under  $\bar{\pi}$  have at least that duration.

We first establish this equivalence for the simpler case  $k = 2$ . We then discuss approaches to casting the whole of the problem in a form that meets topological requirements sufficient to establish existence of fixed points. Finally, we return to the details of the more challenging case  $k > 2$ .

### 3.1 An iterative routing mapping

To apply fixed-point theory, we need a continuous combined routing/assignment mapping. It is not our purpose here to concentrate on the assignment half of the loop; we simply assume a continuous assignment mapping  $\bar{A} : \bar{\Pi} \rightarrow C$  giving the link travel times that follow from a given multipath routing policy. It then suffices to construct a continuous routing function, since the composition of continuous functions is continuous.

In the all-or-nothing case, the routing function  $R$  simply produces a single fastest path for each decision point. Applying  $R$  to the multipath case, calculating fastest paths based solely on time-dependent link travel times  $c \in C$ , would always compute new routing split vectors as unit vectors  $(1, 0, 0, \dots, 0)$ ,  $(0, 1, 0, \dots, 0), \dots$ , i.e., an all-or-nothing choice (except perhaps when there are ties for optimal next link). As we have seen, this is inherently discontinuous;

the routing map must be made to use more information. The new routing map will be *iterative* and can be seen as part of a SAVaNT-type loop, with the difference that it takes a previous routing policy as its additional input. This mapping has the form  $\bar{R} : \bar{\Pi} \times C \rightarrow \bar{\Pi}$ , compared to the all-or-nothing mapping  $R : C \rightarrow \Pi$ .

Consider a “current” policy  $\bar{\pi} \in \bar{\Pi}$  and let  $c = \bar{A}(\bar{\pi})$  be the resulting link travel times. For a particular decision point  $(n, d, t)$ , let  $\ell_1, \dots, \ell_k$  be the links leaving  $n$ . We begin by calculating the time-dependent fastest path from  $(n, t)$  to  $d$  exactly as in the all-or-nothing case. But now we also store a byproduct of that computation, namely the dynamic programming (DP) slacks  $s = (s_1, \dots, s_k)$  associated with  $\ell_1, \dots, \ell_k$ . The DP slacks are defined as follows: if  $q$  is the duration of the fastest path from  $(n, t)$  to the destination, then  $q + s_i$  is the duration of the fastest path from  $(n, t)$  to the destination *choosing  $\ell_i$  as the next link of the trip*. Put another way,  $s_i$  is the unnecessary delay incurred by choosing  $\ell_i$  as the next link, hence  $\ell_i$  is an optimal next link under  $c$  if and only if  $s_i = 0$ . The following properties hold:

- Each slack vector is continuous as a function of  $c$ , under the additional restriction that  $c$  is time-continuous, i.e., that  $c(\ell, t)$  is continuous with respect to  $t$  for all  $\ell \in L$ .
- $s_1, \dots, s_k \geq 0$ , and  $\min\{s_1, \dots, s_k\} = 0$ .

The iterative routing mapping uses the slacks to adjust the current policy; the greater the potential delay of a suboptimal path, the more the current split vector will be adjusted against the first link of that path. The fundamental *balancing* property which is the goal of this adjustment process can be stated in general terms.

**Definition 3** A continuous function  $f : \bar{L}_k \times \mathfrak{R}_+^k \rightarrow \bar{L}_k$  is a *balancing function* if for all  $v \in \bar{L}_k$  and  $s \in \mathfrak{R}_+^k$  with  $\min\{s_1, \dots, s_k\} = 0$ ,

$$v = f(v, s) \text{ if and only if } v_i s_i = 0 \text{ for } i = 1, \dots, k.$$

Interpreting  $v$  and  $s$  as the routing split vector and slack vector for the links outgoing from a certain node, Definition 3 says that given  $v$  and  $s$ , a balancing function  $f$  yields the same routing split  $v$  if and only if the links actually used by  $v$  are already optimal according to  $s$ .

We are interested in iterative routing mappings  $\bar{R}$  which compute each of the new routing splits by balancing functions.

**Definition 4** Let an input multipath policy  $\bar{\pi}$  and link travel times  $c$  be given. Suppose that for every decision point  $(n, d, t)$ , the output policy  $\bar{\pi}' = \bar{R}(\bar{\pi}, c)$  is such that  $v' = f(v, s)$ , where  $s$  is the vector of slacks at  $(n, d, t)$  under travel times  $c$ , the routing split vectors  $v$  and  $v'$  are the  $(n, d, t)$  components of  $\bar{\pi}$  and  $\bar{\pi}'$  respectively, and  $f$  is a balancing function. Then  $\bar{R}$  is called a *balancing routing mapping*.

Under these definitions, when the routing mapping is balancing, the set of fixed points of the combined multipath routing/assignment mapping is exactly the set of user-equilibrium solutions.

**Theorem 2** *If  $\bar{R}$  is balancing, then a routing policy  $\bar{\pi} \in \bar{\Pi}$  satisfies user equilibrium if and only if  $\bar{\pi} = \bar{R}(\bar{\pi}, \bar{A}(\bar{\pi}))$ .*

**Proof:** Let  $c = \bar{A}(\bar{\pi})$  and  $\bar{\pi}' = \bar{R}(\bar{\pi}, c)$ . By definition,  $\bar{\pi}$  satisfies user equilibrium if and only if for each decision point  $(n, d, t)$ ,  $\bar{\pi}(n, d, t) \equiv v$  is such that  $v_i > 0$  only if the  $i^{\text{th}}$  link away from  $n$  is optimal at time  $t$  according to the link travel times  $c$ . With  $s$  as the vector of slacks at  $(n, d, t)$  under  $c$ , this is the same as saying that  $v_i > 0$  only if  $s_i = 0$ , or equivalently that  $v_i s_i = 0$ . But since  $\bar{R}$  is balancing, Definitions 3 and 4 imply that this latter condition holds if and only if  $v = v' \equiv \bar{\pi}'(n, d, t)$ .

Therefore user equilibrium is identical with the condition that  $\bar{\pi}$  and  $\bar{\pi}'$  agree at every decision point  $(n, d, t)$ , i.e., that  $\bar{\pi}$  is a fixed point. ■

To make the nature of balancing routing mappings more clear, let  $k = 2$  and consider

$$f((v_1, v_2), (s_1, s_2)) = \left( \frac{v_1 + s_2}{1 + s_1 + s_2}, \frac{v_2 + s_1}{1 + s_1 + s_2} \right)$$

which is continuous on  $\bar{L} \times \mathfrak{R}_+^2$  and can easily be shown to be balancing. The effect of the function is to adjust the previous routing split vector  $v$  so that it more strongly favors the optimal next link choice. For example, if  $s_1 > 0$ , then link 2 is the sole optimal next link and  $s_2 = 0$ . In this case, the proportion of flow to link 2 is increased by adding  $s_1$  to  $v_2$ , followed by division by  $1 + s_1 + s_2$  to regain normalization. The only fixed point is thus  $v = (0, 1)$ , i.e., all flow routed to link 2, which is exactly the user-equilibrium routing when link 2 is uniquely optimal. On the other hand, if  $s_1 = s_2 = 0$ , i.e., when both links are optimal, the current routing splits are unchanged and all routing splits  $v$  are both fixed points of  $f$  and user-equilibrium routings.

The choice of  $f$  above is far from unique, as illustrated by the following result. We define the general normalization function  $g : \mathfrak{R}_+^k \rightarrow \mathfrak{R}_+^k$  by  $g(x_1, \dots, x_k) = \left( \frac{x_1}{x_1 + \dots + x_k}, \dots, \frac{x_k}{x_1 + \dots + x_k} \right)$ .

**Lemma 1** *Let  $h : \mathfrak{R}_+ \rightarrow \mathfrak{R}_+$  be continuous and such that  $h(x) = 0$  if and only if  $x = 0$ . Then  $f : \bar{L}_2 \times \mathfrak{R}_+^2 \rightarrow \bar{L}_2$  defined by  $f(v, s) = g(v_1 + h(s_2), v_2 + h(s_1))$  is a balancing function.*

**Proof:** Suppose  $v_1 s_1 = v_2 s_2 = 0$  and  $\min\{s_1, s_2\} = 0$ . If  $s_1 > 0$  then  $v_1 = s_2 = 0$ ,  $v_2 = 1$ , and  $f(v, s) = g(0, 1 + h(s_2)) = (0, 1) = v$ . Similarly, if  $s_2 > 0$  then  $f(v, s) = (1, 0) = v$ . And if  $s_1 = s_2 = 0$  then  $f(v, s) = v$  immediately.

Conversely, suppose for all  $v \in \bar{L}_2, s \in \mathfrak{R}_+^2$  with  $\min\{s_1, s_2\} = 0$  that  $v = f(v, s)$ . If  $s_1 > 0$ , then  $s_2 = 0$ , so  $v_1 = \frac{v_1}{1 + h(s_1)}$ , thus  $v_1 = 0$ . Similarly, if  $s_2 > 0$ , then  $v_2 = 0$ . Thus  $v_1 s_1 = v_2 s_2 = 0$ . ■



Our previous balancing function example is simply the case  $h(x) = x$ . Another example, perhaps more sophisticated, would be  $h(x) = e^{\alpha x} - 1$ , where  $\alpha$  is any positive constant. This choice of  $h$  would avoid strongly suboptimal links much more assiduously, while using a finer adjustment for nearly optimal links.

From a practical standpoint, it seems most sensible to require additionally that  $h$  be increasing, as in the examples given above. Under this condition,  $\bar{R}$  always directs flow away from links which currently appear suboptimal, with the strength of the adjustment increasing with the suboptimality of the link. It remains true in general that the new routing policy may violate user equilibrium more strongly than the old policy, due to the changes in link travel times caused by the routing adjustment. Regardless of whether  $h$  is increasing, Theorem 2 assures us that when a fixed-point policy is found, it satisfies user equilibrium and predicts the link travel times which occur under this equilibrium routing. Note also that the flexibility in the choice of balancing function can be exploited in analyzing a single network; Definition 4 permits a different balancing function to be applied to each different decision point  $(n, d, t)$ , if such a course would prove useful.

Surprisingly, implementing balancing functions when  $k > 2$  is more difficult. We return to this issue in section 3.3.

### 3.2 Existence of fixed points

The restrictions placed on the space of routing policies in order to prove existence of fixed points (i.e., user-equilibrium solutions) affect the potential difficulty of computing fixed points. The weakest assumption we make is to permit all policies in  $\bar{\Pi}$  as previously defined. This explicitly includes, for example, policies for which the routing split at a given location  $n$  and destination  $d$  is not continuous with respect to time.

**Theorem 3** *If  $\bar{R} : \bar{\Pi} \times C \rightarrow \bar{\Pi}$  is a balancing routing mapping and  $\bar{A} : \bar{\Pi} \rightarrow C$  is continuous and yields time-continuous link travel time functions, then there exists in  $\bar{\Pi}$  a fixed point of the combined routing/assignment map  $\bar{R}(\cdot, \bar{A}(\cdot))$ .*

**Proof:** See appendix.

The applicability of Theorem 3 to computation is unclear, because the proof relies on giving  $\bar{\Pi}$  the Cartesian product topology over its indices, including uncountably many time points in  $[0, \bar{T}]$ . This topology is therefore not metrizable (cf. Munkres (1979)). In other words, there exists no distance function in  $\bar{\Pi}$  which is compatible with this existence theorem. This will hinder efforts to prove that a proposed algorithm converges to a fixed point.

The obvious first choice for a solution algorithm is the method of successive approximation, in which the SAVaNT loop is explicitly executed as discussed in section 2. The ideal

situation is when a metric  $d(\cdot, \cdot) : \bar{\Pi} \times \bar{\Pi} \rightarrow \mathfrak{R}_+$  is available and it can be proved that  $\bar{R}$  is a *contraction mapping*, i.e., there exists  $0 \leq \alpha < 1$  such that

$$d(\bar{R}(\bar{\pi}_1, \bar{A}(\bar{\pi}_1)), \bar{R}(\bar{\pi}_2, \bar{A}(\bar{\pi}_2))) \leq \alpha d(\bar{\pi}_1, \bar{\pi}_2)$$

for all  $\bar{\pi}_1, \bar{\pi}_2 \in \bar{\Pi}$ . This would imply that successive approximation converges to a fixed point from any starting point, and incidentally that this fixed point is unique (cf. Saaty (1964) ).

To enable the application of this property or other metric analyses, we will impose a number of additional restrictions. First, we must consider only those routing policies in  $\bar{\Pi}$  for which each routing split vector varies continuously over time. Second, the set of routing policies is required to be equicontinuous (cf. Dunford and Schwartz (157,IV.6.6]). Equivalently, each split vector of each policy in the routing space is required to be Lipschitz continuous with respect to time, with a uniform Lipschitz constant characterizing all split vectors at all times. (The uniform Lipschitz property is most easily pictured when applied to a class of differentiable functions over time. The requirement is then simply that a single constant bounds the magnitude of the derivatives of each function over all times.) Restricting attention to an equicontinuous policy space allows the use of the metric topology determined by the supremum norm

$$\|\bar{\pi}\| = \sup_{\substack{n \in N \\ d \in D \\ t \in T}} \|\bar{\pi}_{ndt}\| \tag{1}$$

where the norm on the right-hand side is the Euclidean norm of the particular routing split vector  $\pi_{ndt}$ .

For each  $\epsilon \geq 0$  define  $\bar{\Pi}_\epsilon$  as the set of routing policies in  $\bar{\Pi}$  whose component functions are Lipschitz continuous with uniform Lipschitz constant  $\epsilon$ . Then the policies in  $\bar{\Pi}_\epsilon$  are equicontinuous. To obtain fixed points in this space, we must impose additional restrictions so that  $\bar{\Pi}_\epsilon$  is closed under the mapping  $\bar{R}(\cdot, \bar{A}(\cdot))$ .

The precise form of the closure restriction is potentially quite complicated. The assignment mapping  $\bar{A}$  must be restricted so that the link travel times  $c$  resulting from a policy  $\bar{\pi} \in \bar{\Pi}_\epsilon$  cannot have a time rate of change with magnitude sufficient to produce a routing policy outside of  $\bar{\Pi}_\epsilon$ . Thus, the network is assumed to have a bound on its time-sensitivity of link travel times to travel demands. However, the exact nature of this bound depends on the balancing function used to compute a new routing split vector from a previous split vector and its associated DP slacks. Using the notation of Lemma 1, a sufficient sensitivity restriction for the case  $h(x) = x$  might be inadequate for the case  $h(x) = e^{\alpha x} - 1$ , which performs stronger adjustments on input routing policies. Therefore, to ensure closure of  $\bar{\Pi}_\epsilon$  under  $\bar{R}(\cdot, \bar{A}(\cdot))$ , it will be necessary to consider not just particular aspects of  $\bar{R}$  and  $\bar{A}$ , but also their interdependence.

However, under sufficient closure restrictions, existence of fixed points can be verified under more convenient topological conditions.

**Theorem 4** *Under the restrictions of Theorem 3, and the additional requirement that  $\bar{\Pi}_\epsilon$  is closed under the routing/assignment mapping  $\bar{R}(\cdot, \bar{A}(\cdot)) : \bar{\Pi}_\epsilon \rightarrow \bar{\Pi}_\epsilon$ , then there exists in  $\bar{\Pi}_\epsilon$  a fixed point of  $\bar{R}(\cdot, \bar{A}(\cdot))$ .*

**Proof:** See appendix.

Since the metric topology on  $\bar{\Pi}_\epsilon$  induced by the supremum norm on  $\bar{\Pi}_\epsilon$  is complete, convergence of a sequence of policies  $\{\bar{\pi}_{ndt}^j\}_{j=1}^\infty$  to a fixed point can be established by examining  $\|\bar{\pi}_{ndt}^{j_1} - \bar{\pi}_{ndt}^{j_2}\|$  as  $j_1, j_2 \rightarrow \infty$ .

In summary, Theorem 3 gives sufficient conditions for existence of user-equilibrium routing policies within an essentially arbitrary set of multipath policies, while Theorem 4 gives sufficient conditions, more difficult to satisfy, for existence of user-equilibrium policies in a smaller domain of equicontinuous routing policies, in which algorithmic analysis may be easier.

When the context is such that existence cannot be established, the balancing-function approach may be a valuable heuristic for seeking multipath user equilibrium as previous applications of SAVaNT have been used to seek all-or-nothing equilibria. Both applications are heuristic only in their lack of a convergence guarantee; Theorems 1 and 2 guarantee that if convergence occurs, the resulting fixed point satisfies user equilibrium.

### 3.3 General balancing functions

This section completes the presentation of the balancing-function approach, as a computational heuristic at the very least, by proving constructively the existence of balancing routing mappings when there are more than two links outgoing from the various nodes of the network. Following the construction, the balancing property is proven in Theorem 5 as the counterpart for  $k > 2$  of Lemma 1.

The idea behind the balancing functions of Lemma 1 is to add a term to each component of the current routing split vector, the term being zero for suboptimal links and positive for optimal links. The result is then normalized by applying  $g$ . The natural extension of this approach when  $k > 2$  would be to compute new split vectors as

$$F(v, s) = g \left( v_1 + \min_{j \neq 1} h(s_j), \dots, v_k + \min_{j \neq k} h(s_j) \right)$$

since  $\min_{j \neq i} h(s_j) = 0$  for every suboptimal next link  $i$ . However, if two distinct next links  $\ell_{i_1}$  and  $\ell_{i_2}$  are tied as the optimal next link, then  $s_{i_1} = s_{i_2} = 0$ . As a result,  $\min_{j \neq i} h(s_j) = 0$  for all  $i = 1, \dots, k$ , and thus  $F(v, s) = v$ . For example, if  $v = (1, 0, 0)$ ,  $s = (3, 0, 0)$ , and  $h(x) = x$ , then  $(v, s)$  clearly violates equilibrium, yet  $F(v, s) = v$ .

Defining  $s^* = \max_{i=1, \dots, k} s_i$ , another approach is to use  $s^* - s_i$  in place of  $\min_{j \neq i} h(s_j)$ , i.e.,

$$F(v, s) = g(v_1 + h(s^* - s_1), \dots, v_k + h(s^* - s_k))$$

since this can be relied on to increase the routing split for optimal links (unless all links are optimal). But where our previous attempt failed because some fixed points are not equilibria, this one fails because some equilibria are not fixed points. For example, take  $v = (0, 0, 1)$ ,  $s = (5, 3, 0)$ , and  $h$  as before. This routing split satisfies equilibrium, yet  $F(v, s) = g(0, 2, 5) \neq v$ .

We can resolve the matter by applying the balancing functions from the two-link case in an inductive fashion, based on a decomposed representation of the routing split. We assume without loss of generality that  $k = 2^p$  for some positive integer  $p$ . (Otherwise, simply insert dummy links whose flows are constrained to be zero.) The routing split will now be specified by a collection of routing split vectors in  $\bar{L}_2$ .

At the first level, we divide the list of outgoing links into two groups  $\ell_1, \dots, \ell_{\frac{k}{2}}$  and  $\ell_{\frac{k}{2}+1}, \dots, \ell_k$ . The proportions of flow directed onto each group are denoted as  $v_1^1$  and  $v_2^1$  respectively. At the second level, we divide each of these groups in two, and provide split vectors subdividing the flow within each group. Thus  $v_1^2$  and  $v_2^2$ , which sum to 1, are the proportions of  $v_1^1$  to be directed onto link groups  $\ell_1, \dots, \ell_{\frac{k}{4}}$  and  $\ell_{\frac{k}{4}+1}, \dots, \ell_{\frac{k}{2}}$  respectively. Similarly,  $v_3^2$  and  $v_4^2$ , which sum to 1, give the proportions of  $v_2^1$  to be directed onto the remaining two quarters of the outgoing link set.

In general, the representation consists of vectors  $v^m = (v_1^m, \dots, v_{2^m}^m)$  at each level  $m = 1, \dots, p$ , where  $(v_{2j-1}^m, v_{2j}^m) \in \bar{L}_2$  for all  $j = 1, \dots, 2^{m-1}$  and  $m = 1, \dots, p$ . We also define  $v^0 = 1$ . These are composed into a one-level, i.e., link-by-link, split vector  $\hat{v} \in \bar{L}_k$  by

$$\hat{v}_i = \prod_{m=1}^p v_{\lceil \frac{j}{2^{p-m}} \rceil}^m, \quad i = 1, \dots, k. \quad (2)$$

The multi-level representation, with parentheses indicating the pairs within each level, is illustrated for  $k = 8$  in Figure 3 in the form of a tree with each leaf  $v_i^p$  corresponding to outgoing link  $\ell_i$ .

**Insert Figure 3 about here.**

The set of nodes reached in the tree by tracing arrows in reverse from a leaf (including the leaf itself) will be called the *ancestors* of that leaf. Thus, the ancestors of  $v_i^p$  are simply the terms being multiplied on the right-hand side of (2).

The new routing split will be computed by updating pairs proceeding down the tree, using the balancing functions already available for split vectors in  $\bar{L}_2$ . Given any pair  $(v_{2j-1}^m, v_{2j}^m)$ , the new routing  $(u_{2j-1}^{m+1}, u_{2j}^{m+1})$  is obtained by adjusting the apportionment between the two link groups to favor more strongly the group which contains the optimal choice from among the union of the two groups.

We now denote the DP slacks in the original representation as  $\hat{s} = (\hat{s}_1, \dots, \hat{s}_k)$ . Whether a link group contains an optimal link is indicated by the minimum of its slacks. Thus we

define slack vectors for each level,  $s^m = (s_1^m, \dots, s_{2^m}^m)$ , by

$$s_j^m = \min_{i=1, \dots, 2^{p-m}} \left\{ \hat{s}_{i+2^{p-m}(j-1)} \right\}$$

and from these, we consider optimality strictly within pairs of link groups by defining  $t^m = (t_1^m, \dots, t_{2^m}^m)$  for  $m = 1, \dots, p$ , where

$$(t_{2j-1}^m, t_{2j}^m) = (0, s_{2j}^m - s_{2j-1}^m) \quad j = 1, \dots, 2^{m-1}; m = 1, \dots, p$$

under the assumption that the first of each pair of groups contains a link which is optimal among the union of the two groups. (This assumption can be satisfied without loss of generality by renumbering the complete set of outgoing links so that  $\hat{s}_i$  is increasing in  $i$ .) The multi-level slacks can be computed quickly by going up the tree, under the relation

$$s_j^m = \min \left\{ s_{2j-1}^{m+1}, s_{2j}^{m+1} \right\}.$$

The multi-level split vector elements are then updated in pairs, using the class of balancing functions  $f$  established in Lemma 1. The new routing, denoted  $(u^1, \dots, u^p) = F((v^1, \dots, v^p), (s^1, \dots, s^p))$ , is computed by the finite procedure

$$(u_{2j-1}^m, u_{2j}^m) = f((u_{2j-1}^m, u_{2j}^m), (t_{2j-1}^m, t_{2j}^m)) \quad j = 1, \dots, 2^{m-1}; m = 1, \dots, p.$$

This induces a routing split  $\hat{u}$  in the original routing space  $\bar{L}_k$ , computed from  $(u^1, \dots, u^p)$  in the manner of equation (2).

Note that this composition is not one-to-one; it may be that  $(u^1, \dots, u^p) = F((v^1, \dots, v^p), (s^1, \dots, s^p)) \neq (v^1, \dots, v^p)$ . Theorem 5 employs the stricter condition, proving the actual implemented routing  $\hat{v}$  remains fixed if and only if it produces user equilibrium.

**Theorem 5** *Let  $\hat{u}$  be the composition in  $\bar{L}_k$  of  $(u^1, \dots, u^p) = F((v^1, \dots, v^p), (s^1, \dots, s^p))$ , and let  $\hat{v}$  be the composition of  $(v^1, \dots, v^p)$ . Then  $\hat{u} = \hat{v}$  if and only if  $\hat{v}_1 \hat{s}_1 = \dots = \hat{v}_k \hat{s}_k = 0$ .*

**Proof:** Suppose  $\hat{v}_i \hat{s}_i > 0$  for some  $i$ . Then by (2), the ancestors of  $v_i^p$  are all positive. Proceed upward in the tree from  $v_i^p$  until reaching the first ancestor  $v_j^m$  such that  $s_j^m = 0$ . Now examine the pair  $(v_{2j-1}^{m+1}, v_{2j}^{m+1})$  which splits flow at  $v_j^m$ . The corresponding pair of slacks  $(s_{2j-1}^{m+1}, s_{2j}^{m+1})$  must have one zero element and one positive element, or else the first zero-slack ancestor would have occurred either higher or lower in the tree than level  $m$ . The slacks were assumed to increase from left to right within  $s^m$ , thus  $s_{2j-1}^{m+1} = t_{2j-1}^{m+1} = 0$  and  $s_{2j}^{m+1} = t_{2j}^{m+1} > 0$ .

The latter element of the pair is the ancestor of  $v_i^p$ , i.e.,  $v_{2j}^{m+1} > 0$ , hence  $v_{2j}^{m+1} t_{2j}^{m+1} > 0$ . By Definition 3,

$$(u_{2j-1}^{m+1}, u_{2j}^{m+1}) = f((v_{2j-1}^{m+1}, v_{2j}^{m+1}), (t_{2j-1}^{m+1}, t_{2j}^{m+1})) \neq (v_{2j-1}^{m+1}, v_{2j}^{m+1}).$$

In other words, the total flow onto the union of the two link groups of the pair has been given a new allocation between the two groups. This must cause an actual difference between the routing split compositions  $\hat{v}$  and  $\hat{u}$  unless that total flow onto the union is zero. But as observed previously, the ancestors of  $v_j^m$  are positive, so the total flow is positive and  $\hat{u} \neq \hat{v}$ .

Conversely, suppose  $\hat{u} \neq \hat{v}$ . Then there must be some routing split pair where  $(u_{2j-1}^{m+1}, u_{2j}^{m+1}) \neq (v_{2j-1}^{m+1}, v_{2j}^{m+1})$  and  $v_j^m$  and its ancestors are positive. Then by Definition 3,  $v_j^m t_j^m > 0$  for one element of that pair. Therefore

$$0 < t_j^m \leq s_j^m = \min_{i=1, \dots, 2^{p-m}} \left\{ \hat{s}_{i+2^{p-m}(j-1)} \right\},$$

and thus the slack is positive at every leaf beneath  $v_j^m$ . Furthermore, since the elements of every routing split pair sum to 1, there is beneath  $v_j^m$  at least one leaf  $v_i^p$  whose ancestors (including itself) are all positive. Thus  $\hat{v}_i \hat{s}_i > 0$ . ■

Finally, it should be noted that the multi-level computations should not typically be burdensome. For example, in a detailed metropolitan model we will usually have  $k \leq 8$ , i.e.,  $p \leq 3$ .

## 4 Computational considerations and discrete time approximation

We have established the existence of user-equilibrium solutions by characterizing user equilibrium by a fixed-point property. It remains to discuss computing, at least approximately, these fixed points.

The limited-sensitivity requirement ensuring that  $\bar{\Pi}_\epsilon$  is closed under the routing/assignment map allows routing policies to be considered in a metric space, tending to facilitate proofs of algorithmic convergence. That is, it is easiest to prove convergence of the sequence of policies produced by an algorithm by proving convergence to zero of the sequence of distances between the policy iterates and some limiting policy.

However, in computational practice, it will be necessary to further restrict attention to routing policies that can be represented finitely. For example, we may seek to impose restrictions on  $\bar{A}$  and  $\bar{R}$  so that a space of piecewise linear routing policies is closed under the routing/assignment map. It would then be possible to complete an iteration of successive approximation by computing new routing split vectors only at the finite set of time points between which the piecewise linear routing policy interpolates.

Viewed another way, this approach can be viewed as being based in a discrete-time approximation. In discrete time, a routing policy consists of routing split vectors at a finite number of time points. The routing space is then finite-dimensional, which simplifies the

task of proving existence of fixed points. Under a Euclidean metric on this finite-dimensional space, the set of feasible routing policies is closed and bounded, hence compact, and convex. To conclude existence of fixed points by Schauder's theorem (see Appendix), it remains only to ensure that the routing/assignment map is continuous.

Since the routing/assignment map is the composition of several functions which perform intermediate tasks, we would prove its continuity by proving the continuity of the intermediates. Among these intermediates is the function giving DP slacks from link travel times  $c = \bar{A}(\bar{\pi})$ . For the DP slack function to be continuous, the link travel times  $c$  must range over a continuum, rather than be restricted to a discrete set  $T$ . We may assume that this is an intrinsic property of  $\bar{A}$ , or that it is the result of interpolating between discrete-time link travel times determined by  $\bar{A}$ . In either case, however, the balancing function is being applied only at the discrete time points. The cost of the convenience gained by discrete-time approximation is therefore loss of the equivalence between fixed points and user-equilibrium solutions over the rest of the time continuum.

Future work in discrete time approximation would begin with establishing error measures of the distance from equilibrium of such fixed points, for instance time integrals of individual delay due to suboptimal route choice. Analysis of individual assignment models would then be directed toward identifying a sufficiently fine time discretization so as to achieve a specified bound on the error of the associated discrete-time fixed-point solutions.

## 5 Conclusions

We have provided a general formulation of the dynamic traffic assignment problem as one of finding a fixed point of an iterative routing/assignment transformation. We also established sufficient conditions for such a fixed point, and hence user equilibrium, to exist. The important next step is to develop efficient algorithms that demonstrably converge to such a fixed point. Since the routing procedure is analogous to the best-reply correspondence in game theory, the field of dynamic games should provide some insights into this question.

## References

- [1] Dafermos, S.C. (1980) Traffic equilibrium and variational inequalities. *Transportation Science* **14**, 42–54.
- [2] Dunford, N. and Schwartz, J.T. (1957) *Linear Operators – Part I: General Theory*. Interscience Publishers Inc., New York.
- [3] Friesz, T.L., Bernstein, D., Smith, T.E., Tobin, R.L. and Wie, B.W. (1993) A Variational inequality formulation of the dynamic network user equilibrium problem. *Operations Research* **41**, 179–191.
- [4] Ghali, M. and Smith, M. (1995) A model for the dynamic system optimum traffic assignment problem. *Transportation Research* **29B**, 155-170.
- [5] Janson, B.N. (1991) Dynamic traffic assignment for urban road networks. *Transportation Research* **25B**,143–161.
- [6] Janson, B. N. (1995) Network design effects of dynamic traffic assignment. *Journal of Transportation Engineering* **121**, 1-13.
- [7] Jayakrishnan, R., Tsi, W.K. and Chen, A. (1995) A Dynamic traffic assignment model with traffic-flow relationships. *Transportation Research* **3C**, 51-72.
- [8] Kaufman, D.E., Wunderlich, K.E. and Smith, R.L. (1991) An Iterative routing/assignment method for anticipatory real-time route guidance. *Proceedings, SAE Vehicle Navigation and Information Systems Conference* **P-253**, 701–708.
- [9] Kaufman, D.E., Nonis, J. and Smith, R.L. (1996) A Mixed integer-linear programming model for dynamic traffic assignment. Submitted to *Transportation Research B*.
- [10] Kaufman, D.E. and Smith, R.L. (1993) Fastest paths in time-dependent networks for intelligent vehicle-highway systems application. *IVHS Journal* **1**, 1–11.
- [11] Lafortune, S., Sengupta, R., Kaufman, D.E. and Smith, R.L. (1994) Dynamic system-optimal traffic assignment using a state space model. *Transportation Research* **27B**, 451–472.
- [12] Lam, W. and Huang, H-J. (1995) Dynamic user optimal traffic assignment model for many to one travel demand. *Transportation Research* **29B**, 243-259.
- [13] Munkres, J.R. (1975) *Topology: A First Course*. Prentice-Hall, Inc., Englewood Cliffs, N.J.
- [14] Patriksson, M. (1994) *The Traffic Assignment Problem: Models and Methods*. Koninklijke Wothrmann, Zutphen, The Netherlands.
- [15] Potts, R.B. and Oliver, R.M. (1972) *Flows in Transportation Networks*. Academic Press, Inc., New York.



- [16] Ran, B., Boyce, D.E. and LeBlanc, L.J. (1993) A New class of instantaneous dynamic user-optimal traffic assignment models. *Operations Research* **41**, 192–202.
- [17] Saaty, T.L. (1964) *Nonlinear Mathematics*. McGraw-Hill, Inc., New York.
- [18] Santiago, A.J. and Rathi, A.K. (1990) Urban network traffic simulation: TRAF-NETSIM Program. *Journal of Transportation Engineering* **116**, 734–43.
- [19] Sheffi, Y. (1985) *Urban Transportation Networks : Equilibrium Analysis with Mathematical Programming Methods*. Prentice-Hall, Englewood Cliffs, N.J.
- [20] Smith, M. and Wisten, M. (1995) A continuous day-to-day traffic assignment model and the existence of a continuous dynamic user equilibrium. *Annals of Operations Research* **60**, 59-79.
- [21] Van Aerde, M., Voss, J. and Ugge, A. (1989) Managing traffic congestion in combined freeway and traffic signal networks. *ITE Journal* **59**, 36–42.
- [22] Wie, B.W., Tobin, R., Bernstein, D. and Friesz, T. (1995) A Comparison of system optimal and user optimal equilibrium dynamic traffic assignments with scheduled delays. *Transportation Research* **3C**, 389-411.
- [23] Wong, S.. (1990) TRAF-NETSIM: How it works, what it does. *ITE Journal* **60**, 22–27.
- [24] Wunderlich, K., Kaufman, D. and Smith, R.L. (1997) Link travel time prediction techniques for convergent iterative anticipatory route guidance methods. Submitted to *Transportation Research C*.

## Appendix: Existence proofs

Existence of fixed points is obtained by putting the routing/assignment model into a topological structure to which the following standard result (cf. Dunford and Schwartz (1957, V.10.3)) applies:

**Theorem** (Schauder-Tychonoff) *Let  $E$  be a compact convex subset of a locally convex linear topological space. If  $f : E \rightarrow E$  is continuous, then there exists a fixed point of  $f$  in  $E$ .*

Reviewing notation for the main proof, the multipath routing policy set is  $\bar{\Pi} = \{\bar{\pi} : N \times D \times T \rightarrow \bar{L}_k\}$  where  $T = [0, \bar{T}]$  and  $\bar{L}_k = \{v = (v_1, \dots, v_k) \in \mathfrak{R}_+^k : v \geq 0, v_1 + \dots + v_k = 1\}$ . **Theorem 3** *If  $\bar{R} : \bar{\Pi} \times C \rightarrow \bar{\Pi}$  is a balancing routing mapping and  $\bar{A} : \bar{\Pi} \rightarrow C$  is continuous and yields time-continuous link functions, then there exists in  $\bar{\Pi}$  a fixed point of the combined routing/assignment map  $\bar{R}(\cdot, \bar{A}(\cdot))$ .*

**Proof:** We can safely reinterpret the set of routing policies as

$$\bar{\Pi} = \prod_{n \in N} \prod_{d \in D} \prod_{t \in T} \bar{L}_k$$

viewed as a subset of  $\Omega = \prod_{n \in N} \prod_{d \in D} \prod_{t \in T} \mathfrak{R}_+^k$ . To establish properties of continuity and compactness, we must give  $\Omega$  a topology, i.e., a definition of openness of sets (and by implication of continuity of functions) in  $\Omega$ . We choose the Cartesian product topology (cf. Munkres (1975, sec. 2.8) applied to  $\bar{L}_k$ , which is given the usual (metric) topology of  $\mathfrak{R}^2$ .

That  $\Omega$  is a locally convex linear topological space can be verified from definitions II.1.1 and V.2.9 of Dunford and Schwartz (1957). Convexity of  $\bar{\Pi}$  is trivial.  $\bar{L}_k$  is closed and bounded, hence compact in  $\mathfrak{R}^2$ . By the Tychonoff theorem (cf. Munkres (1975, 5.1.1)) the arbitrary product of compact spaces is compact in the Cartesian product topology.

By the Schauder-Tychonoff theorem, it remains only to show that  $\bar{R}(\cdot, \bar{A}(\cdot))$  is continuous. Equivalently in the product topology (c.f. Munkres (1975, 2.8.5)), we need to show that each component  $\bar{R}_{ndt}(\cdot, \bar{A}(\cdot)) : \bar{\Pi} \rightarrow \bar{L}$  is continuous. Since  $\bar{R}$  is a balancing routing mapping,  $\bar{R}_{ndt}$  is obtained by applying a balancing function  $f(v, s)$ , where  $v = \bar{\pi}(n, d, t)$  and  $s$  is the vector of DP slacks at  $(n, d, t)$ . The slacks  $s$  are determined solely by  $c = \bar{A}(\bar{\pi})$ , and it was observed in section 3.1 that  $s = s(c)$  is continuous in  $c$  if  $c(\ell, t)$  is continuous in  $t$  for all  $\ell \in L$ , i.e., when  $\bar{A}$  yields time-continuity.

Since  $\bar{A}$  itself is also continuous,  $s = s(\bar{A}(\bar{\pi}))$  is continuous in  $\bar{\pi}$ . By definition, the balancing function  $f(v, s)$  is continuous with respect to  $v$  and  $s$ , hence with respect to  $\bar{\pi}$ . Thus each component function  $\bar{R}_{ndt}$ , and hence  $\bar{R}$  itself, is continuous on  $\bar{\Pi}$ . ■

To prove existence of fixed points under time-continuity, we will apply the following standard result (cf. Saaty (1964)):

**Theorem** (Schauder) *Let  $E$  be a compact convex subset of a complete metric space. If  $f : E \rightarrow E$  is continuous, then there exists a fixed point of  $f$  in  $E$ .*

**Theorem 4** *Under the restrictions of Theorem 3, and the additional requirement that  $\bar{\Pi}_\epsilon$  is closed under the routing/assignment mapping  $\bar{R}(\cdot, \bar{A}(\cdot)) : \bar{\Pi}_\epsilon \rightarrow \bar{\Pi}_\epsilon$ , then there exists in  $\bar{\Pi}_\epsilon$  a fixed point of  $\bar{R}(\cdot, \bar{A}(\cdot))$ .*

**Proof:** Let  $\Omega_c$  be the set of time-continuous members of  $\Omega$  defined in previous proof. Under the metric determined by equation (1),  $\Omega_c$  is a complete metric space Dunford and Schwartz (1957, IV.6). Further, since  $\bar{\Pi}_\epsilon$  is closed and bounded in  $\Omega_c$ , the equicontinuity of  $\bar{\Pi}_\epsilon$  is necessary and sufficient for  $\bar{\Pi}_\epsilon$  to be compact in  $\Omega_c$ . Also,  $\bar{\Pi}_\epsilon$  is evidently convex. Denote  $\phi(\cdot) = \bar{R}(\cdot, \bar{A}(\cdot))$ . The conditions carried over from Theorem 3 ensure that each component  $\phi_{ndt}$  is continuous. Thus by (1),  $\|\phi(\bar{\pi}) - \phi(\bar{\pi}^0)\| \rightarrow 0$  as  $\bar{\pi} \rightarrow \bar{\pi}^0$ , i.e.,  $\phi$  is continuous.

Thus a fixed point exists by Schauder's theorem. ■