Dynamic User-Equilibrium Properties of Fixed Points in Iterative Routing/Assignment Methods

David E. Kaufman
Robert L. Smith
Karl E. Wunderlich

University of Michigan
Ann Arbor

The University of Michigan IVHS

Intelligent Vehicle-Highway Systems
College of Engineering • (313) 764-4332
Professor Kan Chen, 4112 EECS, Ann Arbor, MI 48109-2122, and
University of Michigan Transportation Research Institute • (313) 936-1066
Robert D. Ervin, 2901 Baxter Road, Ann Arbor, MI 48109-2150

IVHS Technical Report: #92-12

Dynamic User-Equilibrium Properties of Fixed Points in Iterative Routing/Assignment Methods

David E. Kaufman
Robert L. Smith
Karl E. Wunderlich

University of Michigan
Ann Arbor

IVHS Technical Report: #92-12

Dynamic user-equilibrium properties of fixed points in iterative routing/assignment methods

D.E. Kaufman
R.L. Smith
K.E. Wunderlich
Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, Michigan 48109

July 17, 1992

1 Introduction

An assignment of flow in a traffic network in which no vehicle can alter its route choice so that its trip duration is reduced is described as satisfying user equilibrium. This condition models drivers behaving noncooperatively with perfect information about the available route choices. There are well known models and methods for determining static user equilibrium flows, under the assumption that traffic conditions are constant over time (see, e.g., [8]). The problem of dynamic user equilibrium, where traffic conditions change over time, has not been solved so satisfactorily.

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 given constraints or an objective function intended to force the trip durations to satisfy user equilibrium. These models have difficulties guaranteeing the existence of a user-equilibrium solution and computing a solution when one exists.

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. 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 might not be reproduced by the calculated routings. If the calculated routings reproduce the input link travel times through the assignment map, however, then the routings do satisfy user equilibrium. We describe such routings as fixed points of the combined routing/assignment mapping. We use this characterization to investigate existence of user-equilibrium solutions through fixed-point theory. We consider the effects of modelling in discrete or continuous time and with all-or-nothing or multipath routing.

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 the set of links. Let T be the set of time indices at which decisions can be made. We call a function $\pi : N \times D \times T \rightarrow L$ an *all-or-nothing* routing policy, so called because 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. (We call routings that split the traffic at (n, d, t) onto two or more next links *multipath* policies.) Denote by Π the set of all-or-nothing policies.

We call $c : L \times T \rightarrow \Re^+$ a *travel time function* ($\Re^+$ denotes the nonnegative reals); $c(\ell, t)$ will be interpreted as the link travel time experienced by a vehicle that enters link ℓ at time t. Denote by C the set of link travel time functions.

We require a function $A : \Pi \rightarrow C$, which we call 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 π. We also introduce $R : C \rightarrow \Pi$, which computes as $\pi = R(c)$ the time-dependent fastest paths in the network under link travel times c; we call R the *routing mapping*. We call a routing policy satisfying $R(A(\pi)) = \pi$ a *fixed point*.

The problem of calculating time-dependent fastest paths, i.e., evaluating R, has been explored in [4], where it was shown that 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 encompasses the range from mathematical models based on link impedance functions (e.g. [1,3,5,6]) to traffic simulations (e.g. [10]). We now make formal the characterization of user-equilibrium solutions as fixed points.

Definition 1 A policy $\pi \in \Pi$ satisfies *user equilibrium* if, when all vehicles follow π, the ensuing link travel times are such that all paths that are used by some vehicle from any node $n \in N$ starting at any time $t \in T$ to any destination $d \in D$ have equal duration, and the paths that are not used have equal or greater duration.

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

Proof: By contradiction. Suppose a policy $\pi \in \Pi$ is a fixed point, i.e., $\pi = R(A(\pi))$, but π does not satisfy user equilibrium. Then under the link travel times $A(\pi)$, which prevail in the network under π, 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 the trip duration choosing a different link ℓ' leaving n. But since R determines π as the fastest-path routing under link travel times $A(\pi)$, $\pi(n, d, t) = \ell'$, a contradiction. ∎

The theorem completely characterizes user-equilibrium solutions, but does not specify how to compute them. The natural approach is to choose an arbitrary initial routing policy π_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 by Kaufman, Wunderlich, and Smith [2] under the acronym SAVaNT (Simulation of Anticipatory Vehicle Network Traffic), using the INTEGRATION traffic simulation [10] as the assignment mapping. In a small test network, the iteration frequently terminated with $j = k - 1$, i.e., with a fixed point π_k, where k was typically about 10 iterations. 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.

In general, we cannot guarantee that a fixed point exists under all-or-nothing routing. In the discrete-time case 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. Only two routing policies exist, one choosing the first link and the other choosing the second. Under the usual 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 nonexistence dilemma is twofold. First is the essential discreteness of all-or-nothing routing, which does not permit us to seek an equilibrium solution by splitting the platoon until the flow on the two paths is balanced. This motivates our investigation of multipath routing, which follows. Second, our definition of user equilibrium is somewhat restrictive. In the example, neither policy is a fixed point because the unused path always looks faster; but when the route is changed the travel time is the same as before. Both policies would satisfy a weaker definition of user equilibrium, encompassing policies that allow alternate paths that are faster when unloaded but which would be slower 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. This would be a considerable computational burden for traffic simulation models, and would also be difficult in mathematical assignment models.

It might appear 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 $[0, \hat{t}]$, 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. Even this generalization is not sufficient to guarantee existence of fixed points. To show this, we 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 $[0, t_1]$. Then no matter how small we make t_1, the other link looks better 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 previously we specified a route choice as a next link $\ell \in L$, we now give route choice as a vector of fractional splits. Strictly for simplicity of notation, we assume that G has exactly two links departing each node. Let

$$\bar{L} = \left\{(v_1, v_2) \in \Re^2 : v_1, v_2 \geq 0, v_1 + v_2 = 1\right\}$$

be the space of routing split vectors; if $v = (v_1, v_2)$ is the routing split vector to be applied to a flow of x vehicles at node n, time t with destination d, then for $i = 1, 2$ the destination-d flow onto the i^{th} link away from n at time t is $v_i x$. We do not require that the resulting vehicle flow assignments be integer.

We define the set of multipath policies by $\bar{\Pi} = \{\bar{\pi} : N \times D \times T \to \bar{L}\}$; 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. In particular, we wish to apply the following (see e.g. [9]):

Theorem 2 *(Schauder's theorem) Let X be a complete metric space, and let K be a compact convex subset of X. Then any continuous function $f : K \to K$ has a fixed point $x \in K$.*

In our case, we consider X as a space of routing/assignment functions similar to $R(A(\cdot))$ and K as imposing restrictions that the elements of the routing functions, i.e., the routing split vectors, have nonnegative components summing to one.

In this section we discuss how to formulate routing mappings so that the combined routing/assignment function is continuous. We then discuss approaches to casting the whole of the problem in a form that meets the topological requirements of metrizability, completeness, and compactness. If we meet these requirements, we can conclude by the theorem that a fixed point exists. We also prove that for the routing map we present, a fixed point satisfies user equilibrium. Note that our prior definition of user equilibrium requires no generalization.

3.1 An iterative routing mapping

To apply Schauder's theorem, we must define the routing/assignment function so that it is continuous. It is not our purpose here to concentrate on the assignment half of the loop; we simply assume a continuous assignment mapping $\bar{A} : \bar{\Pi} \to C$ giving the link travel times that follow from multipath routings. It then suffices to construct a continuous routing function, since the composition of continuous functions is continuous. In the all-or-nothing case, we used a routing function R, which simply produced a single fastest path for each decision point. If we proceed in the multipath case in the same way, by calculating fastest paths based on no other input than time-dependent link travel times $c \in C$, the resulting routing split vectors will always be either $(1,0)$ or $(0,1)$, i.e., an all-or-nothing choice. Therefore, we must construct a routing map that uses more information.

We will present an *iterative* routing mapping which can be seen as part of a SAVaNT-type loop, taking a previous routing policy as its additional input. This mapping has the form $\bar{R} : \bar{\Pi} \times C \to \bar{\Pi}$. In the current paper, we will restrict our attention to the case of networks with only one destination; we expect the generalization to multiple destinations to present more difficulty in exposition than in technical content. Henceforth, we suppress notation for the destination set D and its representative element d.

Consider a "current" policy $\bar{\pi}_0 \in \bar{\Pi}$ and let $c = \bar{A}(\bar{\pi}_0)$ be the resulting link travel times. Consider a particular decision point (n,t) and let ℓ_1 and ℓ_2 be the links leaving n. We begin by calculating the time-dependent fastest path from (n,t) to the destination exactly as we did in the all-or-nothing case; but we also store a byproduct of that computation, namely the *dynamic programming slacks* s_1 and s_2 associated with ℓ_1 and ℓ_2. The DP slacks are defined as follows: if p is the duration of the fastest path from (n,t) to the destination, then $p + s_i$ is the duration of the fastest path from (n,t) to the destination *choosing l_i as the next link of the trip*. Put another way, s_i is the unnecessary delay incurred by choosing l_i as the next link, and l_i is an optimal next link if and only if $s_i = 0$. The following properties hold:

- the slacks are continuous as functions of c
- $s_1, s_2 \geq 0$
- $\min\{s_1, s_2\} = 0$
- If $s_1 = s_2 = 0$ then the paths starting with l_1 and l_2 have the same duration and are both optimal.

We use the slacks to adjust the current policy; the greater the potential delay of a suboptimal route choice, the more we will adjust the current split vector to favor the optimal choice. Formally, write the current split vector $\pi_0(n,t)$ as $v^0 = (v_1^0, v_2^0)$ and the new split vector $\pi_1(n,t)$ (the (n,t) component of $\pi_1 = \bar{R}(\pi_0, \bar{A}(\pi_0))$) as $v^1 = (v_1^1, v_2^1)$. We calculate the new split vector by

$$\begin{aligned} v_1^1 &= \frac{1}{e^{\alpha s_1} + e^{\alpha s_2} - 1} v_1^0 + (e^{\alpha s_2} - 1) \\ v_2^1 &= \frac{1}{e^{\alpha s_1} + e^{\alpha s_2} - 1} v_2^0 + (e^{\alpha s_1} - 1) \end{aligned} \tag{1}$$

for some choice of $\alpha > 0$. The denominator is present only to preserve the normalization of the vector, i.e.; to make the components of the split sum to one. Technically, this may result in split vectors with elements outside $[0,1]$; when this happens, write zero in place of anything smaller and one in place of anything larger before normalizing. Note that v^1 is continuous as a function of the previous split v^0 and the slacks s. We noted previously that the slacks are continuous in c. Therefore, the routing mapping for each decision point is continuous over the domain $\bar{\Pi} \times C$, which was our goal in adding the iterative element to the routing mapping. We can now prove that a fixed point of $\bar{R}(\cdot, \bar{A}(\cdot))$ satisfies user equilibrium.

Theorem 3 *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,t), $\bar{\pi}(n,t)_i \equiv v_i > 0$ if and only if the i^{th} link away from n is optimal at time t according the link travel times c, or equivalently if $s_i = 0$. We denote $v' = \bar{\pi}'(n,t)$. We consider three cases. First, $v = (v_1, v_2) = (1,0)$. Then from equation (1), $v = v'$ if and only if $s_1 = 0$. In the second case, when $v = (0,1)$, $v = v'$ if and only $s_2 = 0$ by symmetry. In the third case, when $v = (\beta, 1-\beta)$ for some $0 < \beta < 1$, then $v = v'$ if and only if $s_1 = s_2 = 0$. This last is easy to see because at most one of s_1, s_2 is nonzero.

Thus at each decision point (n,t), $v = v'$ if and only if π directs flow only onto links that are optimal according to c, i.e., if and only if π is user equilibrium. ■

The iterative routing mapping suggests an ad hoc computational procedure just like SAVaNT, beginning with an arbitrary routing policy and applying the assignment and iterative routing mappings until some convergence criterion is met. However, even if we succeed in proving the existence of fixed-point multipath policies, we cannot guarantee that this version of SAVaNT will terminate. To prove convergence of this *successive approximation* method, we would have to prove that $\bar{R}(\cdot, \bar{A}(\cdot))$ is a *contraction mapping* (cf. [9]), which would depend on the specific form of the assignment mapping.

3.2 Existence of fixed points

We close by discussing the topological issues involved in applying Schauder's theorem to conclude existence of fixed points and hence by theorem 3 user-equilibrium routing policies.

Schauder's theorem requires a definition of the policy space that admits of a *metric*, satisfying the usual axioms of nonnegativity, symmetry, and the triangle inequality. The space must be a Banach space, i.e., *complete* under this metric. Completeness, the property that any Cauchy sequence has a unique limit, is a relatively common property of metric spaces; we do not consider this issue to be one of the major obstacles. The set of admissible policies must be compact, which in a metric space is equivalent to the condition that any infinite sequence contains a convergent subsequence. Finally the routing/assignment mapping must be continuous over the set of admissible policies. We have noted above that the individual component of the routing/assignment mapping corresponding to a single routing split vector is continuous, but any proof that the product mapping across all routing split vectors (i.e., all decision points) is continuous depends on the particular choice of metric. The last requirement for the theorem, convexity, is not a topological property; the policy space is convex if for any two policies π_1, π_2, $\lambda\pi_1 + (1-\lambda)\pi_2$ is also a routing policy for all $0 < \lambda < 1$. This is trivial independent of the metric.

In a discrete-time framework, the topology seems deceptively simple. In that case, we can express the policy space as a finite-dimensional Euclidean space, with the usual Euclidean metric making the space complete. In Euclidean space, for the set of admissible policies $\bar{\Pi}$ to be compact, we only need it to be closed and bounded, which follows from the restrictions of nonnegativity and split vectors summing to one. Furthermore, a function on Euclidean space is continuous if and only if its components are continuous, hence the routing mapping is continuous. The difficulty is in the assignment mapping. If vehicles only have decision opportunities at discrete time epochs $0, \Delta t, 2\Delta t, \ldots$, then the link travel times must all occur in exactly the same discrete units. Thus the assignment mapping cannot yield a continuum of travel times, i.e. it is not continuous. Unless we find a new way of looking at this situation, the discrete-time case will remain a potentially valuable computational approach that has theoretical validity only as an approximation to a more general mathematical framework.

The continuous-time case is more complicated. Here the routing split vectors appear as functions into Euclidean space over a continuous time interval $[0, \hat{t}]$. We have considerable freedom in choosing a topology within these function spaces, but the well-known choices all have drawbacks. One way to ensure compactness is to restrict our attention to routing split vector functions that are continuous over time and try to apply Ascoli's theorem. However, it is not apparent that a class of continuous functions over time is rich enough to model the vehicle routing controls we wish to study. Among other well-known topologies, the L_∞ space (induced by the *supremum* norm) can be shown to fail compactness, and the *product* topology $\Re^{[0,\bar{t}]}$ is compact by the Tychonoff theorem but fails to be metrizable. (For more information on the topological concepts we have invoked, see [7].)

References

[1] Janson, B.N. 1989. Dynamic Traffic Assignment for Urban Road Networks. *Transportation Research B* **25B** 2/3 143-161.

[2] Kaufman, D.E., K.E. Wunderlich, and R.L. Smith. 1991. *An Iterative Routing/ Assignment Method for Anticipatory Real-Time Route Guidance.* IVHS Technical Report 91-01, University of Michigan.

[3] Kaufman, D.E., J. Nonis, and R.L. Smith. 1992. *A Mixed Integer-Linear Programming Model for Dynamic Traffic Assignment.* IVHS Technical Report, University of Michigan.

[4] Kaufman, D.E. and R.L. Smith. 1990. *Minimum Travel Time Paths in Dynamic Networks with Application to Intelligent Vehicle/Highway Systems.* IVHS Technical Report 90-11, University of Michigan.

[5] Lafortune, S., R. Sengupta, D.E. Kaufman, and R.L. Smith. 1991. *A Dynamical System Model for Traffic Assignment in Networks.* IVHS Technical Report 91-01, University of Michigan.

[6] Lafortune, S., R. Sengupta, D.E. Kaufman, and R.L. Smith. 1992. *Optimal Dynamic Traffic Assignment Using a State Space Model.* IVHS Technical Report 92-09, University of Michigan.

[7] Munkres, J.R. 1975. *Topology: A First Course.* Prentice-Hall, Inc., Englewood Cliffs, N.J.

[8] Potts, R.B. and R.M. Oliver. 1972. *Flows in Transportation Networks.* Academic Press, Inc., New York.

[9] Saaty, T.L. 1964. *Nonlinear Mathematics.* McGraw-Hill, Inc., New York.

[10] Van Aerde, M., J. Voss, A. Ugge, and E.R. Case. 1989. An Integrated Approach to Managing Traffic Congestion in Combined Freeway and Traffic Signal Networks. *ITE Journal* **59** 2.