Dynamic network loading based on link travel times

Omar Mansour*

Tomer Toledo

Technion – Israel institute of Technology

{omar, toledo}@technion.ac.il

* Corresponding author

Introduction

Dynamic network loading (DNL) models have been widely used in transportation planning and operations to evaluate long-term and short-term traffic states in urban areas. More recently, DNLs are also used in real-time to predict traffic conditions in support of making control actions. Most DNL models formulate and solve the problem using densities and flows as the variables of interest.

In some applications, other variables, such as speeds and travel times, may be of required. These may be indirectly calculated from the obtained solutions. In particular, travel times are often of interest, either from the users' perspective, or with a social objective to minimize total travel times in the system. Furthermore, sensing technology advancements provide the capability to directly measure travel times, for example, using Bluetooth sensors, navigation applications and number plate recognition systems. As a result, there is a need to explore the travel time dynamics in transportation networks.

In this paper, we report on the development of a new DNL model that is defined in terms of travel time dynamics, rather than densities. First, the basic model is developed and an algorithm to solve it are presented. This model is appropriate for uncongested networks, in which queues do not propagate to affect upstream links on the network.

Travel time dynamics

Single link

We begin the derivation of the model by defining the travel time dynamics for a single link. Let $N(x,t)$ be the cumulative flow at point x and time t. Figure 1 shows the cumulative flow curves at two points: the entrance and exit of a link, which are denoted with – and + signs, respectively.

In the figure, travel times on the link are the time difference between points of equal cumulative flow on the two curves. These travel times are time dependent and may be defined based on the entry time to the link (forward travel time, $\tau^{+}(t)$) or the exit time (backward travel times, $\tau^{-}(t)$).

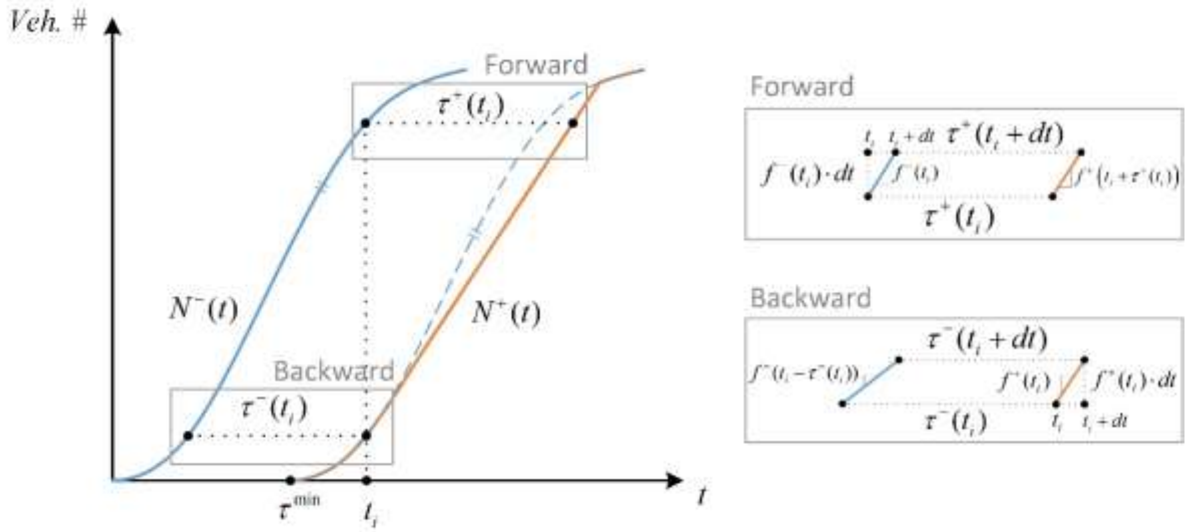

Figure 1 Cumulative flow curves and travel times on a single link

The link inflow and outflow are expressed by the time derivatives of the cumulative curves. We denote these as $f^{-}(t)$ and $f^{+}(t)$, respectively. Finally, we note that link travel times are bounded from below by $\tau^{\min}$, which is the travel time achieved with the free flow speed. Using these variables, the forward travel times dynamics is given by:

$$\tau^{+}(t+dt)=\max\left\{\tau^{+}(t)+\left(\frac{f^{-}(t)}{f^{+}\left(t+\tau^{+}(t)\right)}-1\right)dt,\quad \tau^{\min}(t)\right\} \tag{1}$$

This forward travel time dynamic was used in delay-function models. However, its use within a DNL is difficult because it depends on the future unknown outflow $f^{+}\left(t+\tau^{+}(t)\right)$.

We now focus on the backward travel times. In this case, the travel times dynamics are given by:

$$\tau^{-}(t+dt)=\max\left\{\tau^{-}(t)+\left(1-\frac{f^{+}(t)}{f^{-}\left(t-\tau^{-}(t)\right)}\right)dt,\quad \tau^{\min}(t)\right\} \tag{2}$$

Equation (2) above holds as long as $f^{-}\left(t-\tau^{-}(t)\right)>0$. If the entry flow $f^{-}\left(t-\tau^{-}(t)\right)=0$, then, the outflow $f^{+}(t)$ entered the link later, at the moment that again $f^{-}\left(t-\tau^{-}(t)\right)>0$ is again positive. In the case that downstream traffic conditions do not affect inflows, Algorithm 1, shown in Figure 2, is a time-based calculation of the travel time dynamics for a single link. We next show how this algorithm can be combined with a node processing algorithm to model networks, if they do not experience spillback effect.

Algorithm 1 pseudo-code for travel-time dynamics in a single link

input: $f^-(t), f^{\max}(t)$

for $t = T_0 : dt : T$

$$f^+(t) = \begin{cases} \min\left(f^-\left(t - \tau^-(t)\right), f^{\max}(t) \right) & \tau^-(t) = \tau^{\min} \\ f^{\max}(t) & \tau^-(t) > \tau^{\min} \end{cases}$$

$temp = \tau^-(t)$

while $f^-(t - temp) = 0$ and $temp > \tau^{\min}$

$temp = \max\left(temp - dt, \quad \tau^{\min}\right)$

end

if $f^-(t - temp) > 0$

$$\tau^-(t + dt) = \max\left(\tau^-(t) + \left(1 - \frac{f^+(t)}{f^-(t - temp)}\right) dt, \quad \tau^{\min} \right)$$

else

$\tau^-(t + dt) = \tau^{\min}$

end

end

Figure 2 Algorithm 1 - travel time dynamics on a single link

Node processing

Nodes in the network represent intersections, points of geometrical changes or network boundaries. As such, they represent any bottleneck and are the source to any delay in the network. Some nodes are origins and/or destinations. Origin nodes contain the demand profile for the routes originating there.

Each node maintains sets of its incoming and outgoing links, and similar sets of routes. These are defined by:

$$\mathbf{f}_{\mathbf{n}}^{-} \triangleq \begin{bmatrix} f_1^{+} \\ \vdots \\ f_i^{+} \\ \vdots \\ f_I^{+} \end{bmatrix}, \quad \mathbf{f}_{\mathbf{n}}^{+} \triangleq \begin{bmatrix} f_1^{-} \\ \vdots \\ f_j^{-} \\ \vdots \\ f_J^{-} \end{bmatrix}, \quad \mathbf{f}_{\mathbf{n}}^{\mathbf{r-}} \triangleq \begin{bmatrix} \mathbf{f}_{\mathbf{1}}^{\mathbf{r+}} \\ \vdots \\ \mathbf{f}_{\mathbf{i}}^{\mathbf{r+}} \\ \vdots \\ \mathbf{f}_{\mathbf{I}}^{\mathbf{r+}} \end{bmatrix}, \quad \mathbf{f}_{\mathbf{n}}^{\mathbf{r+}} \triangleq \begin{bmatrix} \mathbf{f}_{\mathbf{1}}^{\mathbf{r-}} \\ \vdots \\ \mathbf{f}_{\mathbf{j}}^{\mathbf{r-}} \\ \vdots \\ \mathbf{f}_{\mathbf{J}}^{\mathbf{r-}} \end{bmatrix}, \quad i \in L_n^{-}, \quad j \in L_n^{+}, \quad n \in N$$

Where, L_n^{-}/L_n^{+} are the sets of upstream and downstream links for node n, respectively. $\mathbf{f}_{\mathbf{i}}^{+}$ and $\mathbf{f}_{\mathbf{n}}^{-}$ are vectors of the flows entering and exiting the node, respectively. That is they are the outflows on the links entering the node and the inflows on the links that exit the node. f_i^{+} is exit flow on link i that enters the node. f_j^{-} is the entry flow to link j that exits from the node.

$\mathbf{f}_{\mathbf{n}}^{\mathbf{r-}}$ and $\mathbf{f}_{\mathbf{n}}^{\mathbf{r+}}$ are the corresponding vectors or flows on routes entering and exiting the node. Without loss of generality, we assume that they are ordered by the entry or exit links that they use through the node. Thus, $\mathbf{f}_{\mathbf{i}}^{\mathbf{r+}}$ and $\mathbf{f}_{\mathbf{j}}^{\mathbf{r+}}$ are vectors of the routes that enter the node through link i, and those that exit the node through node j, respectively.

The mapping of flows on links entering the node to flows on links exiting the node is given by:

$$\boldsymbol{\beta} = \begin{bmatrix} \mathbf{1}_{1\times \mathrm{K1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{1}_{1\times \mathrm{KJ}} \end{bmatrix}^{T} \begin{bmatrix} \mathbf{T}_{\mathbf{11}} & \cdots & \mathbf{T}_{\mathbf{1I}} \\ \vdots & \ddots & \vdots \\ \mathbf{T}_{\mathbf{J1}} & \cdots & \mathbf{T}_{\mathbf{JI}} \end{bmatrix} \begin{bmatrix} diag(\mathbf{p}_{\mathbf{1}})_{\mathrm{M1}\times \mathrm{M1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & diag(\mathbf{p}_{\mathbf{I}})_{\mathrm{MI}\times \mathrm{MI}} \end{bmatrix} \begin{bmatrix} \mathbf{1}_{1\times \mathrm{M1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{1}_{1\times \mathrm{MI}} \end{bmatrix} \quad (3)$$

Where, the first matrix in the multiplication is the link-route incidence matrix for the links exiting from the node. Similarly, the last matrix is a link-route incidence matrix for the links entering the node. The dimensions $K1$ through KJ are the numbers of routes that use the exiting links. Similarly, $M1$ through MI are the numbers of routes that use the entering links. The matrices T are mapping from the routes on links that enter the nodes to those that exit the nodes. P are the fractions of the flows on links that stem from the relevant routes, such that for every entering link, the values of p for all routes that pass through it sum to 1.

The relationship between the links' inflows and outflows in a node is given by:

$$\mathbf{f}_{\mathbf{n}}^{+} = \boldsymbol{\beta} \cdot \mathbf{f}_{\mathbf{n}}^{-} \quad (4)$$

The corresponding relation among route flows is given by:

$$\mathbf{f}_{\mathbf{n}}^{\mathbf{r+}} = \mathbf{T}_{\mathbf{n}} \cdot \mathbf{f}_{\mathbf{n}}^{\mathbf{r-}} \quad (5)$$

We assume that entry capacities are associated with links in the network. In some cases, the flows calculated for these links may exceed their entry capacities. Then, the exit flows on the links entering the ode need to be adjusted so that the capacity constraints are respected. This may be done by multiplying them by a factors α that captures the priority policy for these links. For example, the factor may reflect a policy where the proportion of vehicles on the various routes are kept. With the priority policy, the node update as follow:

$$\begin{aligned} f_j^- &= \min\left(f_j^{\max}(t), \quad \beta \cdot f_i^+ \right) && \forall i \in L_n^-, \\ f_i^+ &= \alpha_i \cdot f_i^+ && \forall j \in L_n^+ \end{aligned} \tag{6}$$

If the priority factors do not maintain the proportions of flows on the various routes, these proportions P are adjust accordingly for the next time step. These proportions are also shared with downstream nodes so that they can also update their routes proportions accordingly.

Network

In this section we combine the travel time dynamics for a single link with the node update to simulate the traffic in a network level. Figure 3 shows an algorithm to simulate traffic flow on the entire network using a pre-defined set of routes.

Algorithm 2 pseudo-code for travel-time dynamics for a network

input: Nodes, Links, Routes, Demands

for $t = T_0 : dt : T$

for $\forall l \in Links$

$$f_l^+(t) = \begin{cases} \min\left(f_l^-\left(t - \tau_l^-(t)\right), f_l^{\max} \right) & \tau_l^-(t) = \tau_l^{\min} \\ f_l^{\max} & \tau_l^-(t) > \tau_l^{\min} \end{cases}$$

end

for $\forall n \in Nodes$

if $n \in Origins$

$$f_n^+(t) = OriginUpdate\left(Nodes_n, Routes_n, Demand_n(t)\right)$$

else

$$\left[f_n^+(t), f_n^-(t) \right] = NodeUpdate\left(f_n^-(t), Nodes_n, t - \boldsymbol{\tau}_{\mathbf{n}}^{\mathbf{-}}(\mathbf{t}) \right)$$

end

end

for $\forall l \in Links$

$temp = \tau_l^-(t)$

while $f_l^-(t - temp) = 0$ and $temp > \tau_l^{\min}$

$temp = \max\left(temp - dt, \quad \tau_l^{\min}\right)$

end

if $f_l^-(t - temp) > 0$

$$\tau_l^-(t + dt) = \max\left(\tau_l^-(t) + \left(1 - \frac{f_l^+(t)}{f_l^-(t - temp)} \right) dt, \quad \tau_l^{\min} \right)$$

else

$\tau_l^-(t + dt) = \tau_l^{\min}$

end

end

end

Figure 3 Algorithm 2 - travel time dynamics for a network

Results

To test the model, we compare the results of this model with a microscopic traffic simulation model. The test network is a section of the Ayalon Highway passing through the Tel Aviv metropolitan area in Israel. The model is first calibrated to the micro-simulation results through minimization of the difference between the routes travel times from the two models during the peak period. Calibration variables are the link entry capacities described above. We then present results of running the model with various incidents and interruptions to traffic flow.