Stochastic Eco-routing in a Signalized Traffic Network

Jie Sun, Henry X. Liu

*Department of Civil, Environmental, and Geo-Engineering, University of Minnesota, Twin Cities, 500 Pillsbury Drive SE, Minneapolis, MN 55455, USA

b Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, 2320 G.G. Brown, 2350 Hayward Street, Ann Arbor, MI 48109-2125, USA

c University of Michigan Transportation Research Institute (UMTRI), 2901 Baxter Road, Ann Arbor, MI 48109, USA

Abstract

In this paper, an eco-routing algorithm is developed for vehicles in a signalized traffic network. The proposed method incorporates a microscopic vehicle emission model into a Markov decision process (MDP). Instead of using GPS-based vehicle trajectory data, which are used by many existing eco-routing algorithms, high resolution traffic data including vehicle arrival and signal status information are used as primary inputs. The proposed method can work with any microscopic vehicle model that uses vehicle trajectories as inputs and gives related emission rates as outputs. Furthermore, a constrained eco-routing problem is proposed to deal with the situation where multiple costs present. This is done by transferring the original MDP based formulation to a linear programming formulation. Besides the primary cost, additional costs are considered as constraints. Two numerical examples are given using the field data obtained from City of Pasadena, California, USA. The eco-routing algorithm for single objective is compared against the traditional shortest path algorithm, Dijkstra's algorithm. Average reductions of CO emission around 20% are observed.

Keywords: eco-routing; shortest path; traffic signal; vehicle emission

1. Introduction

In the U.S., approximately 30% of the nation’s total petroleum consumption is made by vehicles on road (EPA, 2008b). Transportation sector is also a significant contributor of total greenhouse gas emission in U.S. (EPA, 2008a). Therefore, environmental problems related to transportation system have increasingly attracted people’s attention. Recently, finding an optimal route that is most environmentally friendly is formulated as “eco-routing” problems and different solution methods have been proposed (Ericsson et al., 2006; Barth et al., 2007; Boriboonsomsin et al., 2012; Nie and Li, 2013).

By following the environmentally friendly paths, vehicles are expected to use less gas or make less emissions. Although there have been many methods to find the optimal paths in terms of travel distance or travel time, it has been shown that a time or distance minimizing route does not always minimize fuel consumption or emissions (Ahn and...
The problem to consider environmentally related costs is much more complicated than those to use time or distance as costs, as vehicle fuel consumption and emissions depend on many factors. Because of this, various microscopic vehicle emission models have been developed to estimate vehicle fuel consumption and emissions (Barth et al., 2000; EPA, 2012; Frey et al., 2010; Onayama et al., 2001; Rakha et al., 2004). These models usually use vehicle trajectories as one of the most important inputs to calculate vehicle fuel consumption rate and emissions rates. They also require other inputs such as road grades and vehicle characteristics. But such information is usually static and relatively easy to obtain.

The key to an eco-routing problem becomes how to estimate the vehicle trajectories. It is obvious that vehicle trajectories on a link depend on many factors. When dealing with eco-routing problem, people usually estimate a trajectory on a link based on historical vehicle trajectories collected by GPS devices or a set of explanatory variables for a link. Then, this information is used as the input to vehicle emission models to calculate the vehicles fuel consumption and emissions for that link. After calculating the environmental cost for each link, a standard shortest path algorithm is used to calculate the optimal path that minimizes the environmental impacts (Ericsson et al., 2006; Barth et al., 2007; Boriboonsomsin et al., 2012).

The above approaches usually involve the collection of a large amount of GPS-based vehicle trajectories, but they ignore detailed traffic signal and queue information that is obtainable from the traffic controller (Liu and Ma, 2009; Liu et al., 2009). Such information is essential in deciding vehicle trajectories in a signalized traffic network. The environmental consequences of vehicle activities in signalized traffic network can be significant due to frequent stops caused by traffic signals. Thus, it is crucial to incorporate such information into an eco-routing problem.

The presence of traffic signals brings much complexity into the problem. In the case of vehicle actuated traffic signals, the durations of red lights are not deterministic. This brings randomness into the problem. In addition, costs on adjacent links may be correlated because of traffic signal coordination on major corridors in urban areas. To deal with these issues, we employ a Markov decision process (MDP) based formulation of vehicle routing problem (Sun and Liu, 2014).

By using the MDP framework, we are able to estimate vehicle trajectories link by link given signal status at intersections. The estimation process assumes a vehicle only stops because of red lights or queued vehicles in front of it. The estimated trajectories are used as inputs to microscopic vehicle emission models, such as CMEM and VT-Micro (Barth et al., 2000; Rakha et al., 2004). Then, microscopic vehicle emission models provide vehicle fuel consumption and emission rates, which are used to calculate step costs needed in the MDP. Different from traditional shortest path algorithm, the MDP provides an optimal policy that gives a vehicle en-route guidance based on the newest available information, so that the expected total cost to the destination is minimized.

For an eco-routing problem, it might be insufficient to consider only one cost. Many microscopic vehicle emission models, such as CMEM and VT-Micro, can generate estimation of fuel consumption, HC, CO, NOx and CO2. Together with travel distance and time, there are multiple costs of interest for a single trip. As some of the objectives conflict with each other, it might be useful to consider several of them at the same time.

When there are multiple concerns in one problem, people usually formulate the problem as a multi-objective problem. One common approach to a multi-objective problem is to find an optimal solution to a problem with an objective of weighted average of different costs. Applying this approach to the problem described above is straightforward, once the weights are known.

But in some cases, the weighted average approach may not be most appropriate. For example, one may want to minimize the fuel consumption while keeping the travel time less than a given threshold. In this situation, it is more appropriate to use a constrained method, where there is a primary objective and some other objectives are considered as constraints of the problem. This is also consistent with the international agreement such as Kyoto Protocol that sets emission targets for given pollutants. Although minimization of some emissions, e.g. CO2, may not be achievable, it is acceptable to maintain the emissions below a given level.

A constrained eco-routing problem can still be formulated as a MDP. It is known that an MDP problem can be transformed into a linear program and solved using standard linear programming techniques (Altman, 1999). For a given constrained eco-routing problem, we first transformed the original MDP formulation into a linear programming formulation and set a primary objective such as travel time or fuel consumption. Then, other costs of interest are considered as constraints. For each addition cost, there is a corresponding constraint added to the transformed linear
The resultant linear program with additional constraints can be solved by standard solution techniques, e.g. simplex method.

This paper is organized as follows. In Section 2, we will give a brief review of related work on eco-routing problems. In Section 3, the formulation of eco-routing problem based the Markov decision process is introduced. The estimation of environmentally related costs is also discussed in Section 3. Following this, the treatment to the constrained eco-routing problem is given in Section 4. Section 5 includes two numerical examples based on real world traffic data and Section 6 concludes this paper.

2. Literature review

Many eco-routing methods relied on GPS-based vehicle trajectory data. In a study by Ericsson et al. (2006), streets were classified as different types according 6 features: street function, type of environment, speed limit, density of traffic signals, traffic-calming measures, and traffic flow. The first 5 features were static and the last one was dynamic. The traffic flow conditions on each road segment were classified as peak or off-peak hours according to traffic counts. For each street type, many driving patterns from GPS data were extracted and then put into microscopic engine map models to calculate corresponding fuel consumption. Finally, a weighted fuel consumption factor (FCF) for each link was calculated and used as inputs to a shortest path search problem based on Dijkstra algorithm.

Similar to this, an eco-routing navigation system was developed based on multiple sources of traffic information (Boriboonsomsin et al., 2012). In addition to historical GPS-based vehicle trajectory data, real-time data from wireless vehicle sensors, probe vehicles, and loop detectors are all incorporated with a data fusion algorithm for estimating energy/emissions operational parameter set (EOPs), which is similar to FCF mentioned above.

Rakha et al. (2012) proposed a simulation-based approach for eco-routing. In this framework, traffic assignment model was used to estimate vehicle speed on links, and then different fuel consumption rates for corresponding speed level were used to calculate link fuel consumption. This approach may not be suitable for real-time vehicle navigation as microscopic traffic simulation is required, which is computationally expensive. In addition, this approach only used cruise speed to calculate the emissions and fuel consumption. Since fuel consumption rate is greatly affected by a vehicles acceleration and deceleration process (Rakha et al., 2000), ignoring vehicle acceleration and deceleration process will lead to large estimation errors.

The eco-routing problem can also be formulated as a mathematical program (Nie and Li, 2013). In their study, the objective was to minimize the total travel costs, which were the monetary value of both fuel and time consumed from origin to destination. In addition, a constraint on CO2 emission was imposed to the problem based on CO2 emission standard. Different from previous research, more than one objective was considered in their model. Besides primary objectives such as fuel consumption and travel time, emissions (e.g. CO2) were added as constraints and the problem was formulated as a constrained shortest path problem. Delays at intersections and emission associated with them are explicitly considered and many microscopic vehicle behaviors are preserved in the model.

Environmental related route choice problems with multiple objectives have also been studied in the context of traffic assignment problems. Tzeng and Chen (1993) proposed a traffic assignment model that simultaneously considered travel distance, travel time and CO emission in the objective function. CO emission on a link was modeled as a linear function of link traffic volume. Chen et al. (2011) formulated a traffic assignment model in such a way that CO emission was considered as a side constraint. Link CO emission was modeled as nonlinear function of link length and link travel time.

In the past, various microscopic vehicle emission models have been developed to estimate vehicle fuel consumption and emissions (Barth et al., 2000; Frey et al., 2010; Oneyama et al., 2001; Rakha et al., 2004; EPA, 2012; Pelkmans et al., 2004)(see a summary in Table 1). These models usually use second-by-second vehicle speed profiles as one of the most important inputs and generate vehicle fuel consumption and emission rates as outputs. As second-by-second vehicle speed profiles are highly dependent on traffic states, and the stochastic nature of traffic states makes it difficult to obtain vehicle trajectories before actual trips.

In the literature, almost no attention has been paid to effects of traffic signal control on eco-routing problem. As described in the introduction, this piece of information is vital in deciding vehicle trajectory on links within a signalized traffic network, which in turn plays an important role in the calculation of fuel consumption and vehicle emission.
Traffic signal information has already been considered in some shortest path problems. Chen and Yang (2000) studied a shortest path problem with the presence of fixed timing traffic signals, which were modeled as multiple time windows. Ahuja et al. (2002) extended the shortest path problem considering fixed timing traffic signals to allow the costs to be time-dependent. They have showed that the minimum time path problem could be solved in polynomial time, but the minimum cost path problems were generally NP-hard. Yang and Miller-Hooks (2004) studied the shortest path problem with adaptive traffic signals. The available and unavailable times for a movement at an intersection were assumed to be exponentially distributed and modeled as a two-state continuous time Markov chain (CTMC). Although their paper provided some explanations for the choice of the CTMC modeling approach, it is difficult to connect their model to real world traffic signal control parameters. Sun and Liu (2014) studied the shortest path problem where vehicle actuated traffic signal is used. They incorporated the signal and queue information in the shortest path problem based on Markov decision process (MDP). The resultant problem can be solved by value iteration method, which is widely used for solving MDP.

3. The eco-routing problem as a Markov decision process

Traditional path search algorithms are mainly concerned with costs on links. This makes it difficult to model vehicle behaviors at intersections, which are vital to determine vehicle trajectories and thus vehicle emissions and fuel consumption. In this research, we solve the eco-routing problem based on Markov decision process (MDP). This is similar to the travel time minimizing path search algorithm that has been developed by Sun and Liu (2014). As this method has explicitly modeled vehicle behaviors at intersections, it is very suitable for optimal path search problems with the objective to minimize environmental impacts. For eco-routing problem, the estimation of environmentally related costs is so complicated that a microscopic vehicle emission model is needed. How to incorporate a vehicle emission model into an MDP based path search algorithm is the major question needs to be answered in this research.

The Markov decision process (MDP) has been extensively studied in the literature. Interested readers can find a good introduction to MDP in the book by Puterman (1994). For the sake of completeness, a brief introduction to MDP is given below.

3.1. Markov decision process (MDP)

A Markov decision process (MDP) is a discrete time stochastic control process. The system is described by a set of states \( S \). A stage is the process between two consecutive states. At a given time step or stage \( k \geq 0 \), the system is in a state \( s \in S \). There is an initial state \( s_0 \) at stage \( k = 0 \). At each stage \( k \), the controller or decision maker chooses an action \( a \in A_s \subseteq A \), where \( A_s \) is the set of available actions in state \( s \) of stage \( k \) and \( A \) is the action set. The system then randomly transits to a new state \( s' \) at next stage with probability \( p(s'|s, a) \). The cost corresponding to this transition is a function of state \( s \) and action \( a \) and written as \( c(s, a) \). So an MDP can be defined as a quadruplet,

\[
\mathcal{M} = (S, \{A_s\}, c(s, a), p(s'|s, a)).
\]  

Please note the transition probability only depends on the current state \( s \) and action \( a \), but not previous states and actions. The state and action at stage \( k \geq 0 \) are denoted by \( s_k \) and \( a_k \) respectively. The system’s behavior, when \( k \) goes to infinity, is then described by a stochastic process \( \{(s_k, a_k)\}_{k=0}^{\infty} \).

The mathematical model introduced above is very general. We usually impose some assumptions on the model. First, it is assumed that \( S \) and \( A_s \) do not vary with \( k \). In this research, we also assume both \( S \) and \( A_s \) are finite.

Table 1. Summary of major vehicle emission models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Developed by</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMEM</td>
<td>University of California, Riverside</td>
<td>HC, CO, NOx, CO2 and fuel</td>
</tr>
<tr>
<td>VSP</td>
<td>North Carolina State University</td>
<td>HC, CO, NO, and fuel</td>
</tr>
<tr>
<td>VT-Micro</td>
<td>Virginia Tech</td>
<td>HC, CO, NOx, CO2 and fuel</td>
</tr>
<tr>
<td>MOVES 2010b</td>
<td>US EPA</td>
<td>CO, NO, NOx, PM10, PM2.5, fuel, etc</td>
</tr>
</tbody>
</table>
This assumption eliminates many subtle mathematical issues when they are not discrete, but is sufficient for our application. In addition, we assume that \( \sum_{s' \in S} p(s' | s, a) = 1 \), which basically says there is no leakage out of the system during the process. Not requiring this assumption allows wider application of the model, but that is not the focus of this research. In the formulation, we consider the stage cost as a function of current state and action, \( c(s, a) \). When the cost will also depend on the state at the next stage, we denote the corresponding cost by \( c(s, a, s') \) and calculate the expected cost at current stage by

\[
c(s, a) = \sum_{s' \in S} c(s, a, s') p(s' | s, a).
\] (2)

A decision rule, denoted by \( d(s) \), is a procedure for action selection at a given state \( s \). If the action is chosen with certainty, then we say it is a deterministic or pure. A randomized decision rule specifies a probability distribution on a set of downstream intersections of intersection \( v \), and a vehicle’s arrival times at both intersections \( t_w \). And the stops, in turn, depend on a vehicle’s arrival times and arrival directions at both intersections. So a good choice of the state space should include the information at current intersection \( v \), traffic stream intersection \( u \), and \( t_w \). The vehicle trajectory on a link largely depends on whether the vehicle stops at the upstream intersection as well as the downstream intersection. And the stops, in turn, depend on a vehicle’s arrival times and arrival directions at both intersections. So a good choice of the state space should include the information at current intersection \( v \), down stream intersection \( w \), and a vehicle’s arrival times at both intersections \( t_v, t_w \).

Let \( u \) be the upstream intersection of \( v \). The arrival direction at intersection \( v \) can be determined given \( u \). And the arrival direction at intersection \( w \) is known given \( v \). Mathematically speaking, the state space of a stochastic eco-routing problem can be written as

\[
S = \{(u, v, w, t_v, t_w) : u \in U_v, v \in V, w \in W_v, t_v \in T_v, t_w \in T_w, T_v \subset T, T_w \subset T\},
\] (4)

where \( U_v \) is the set of upstream intersections of intersection \( v \); \( V \) is the set of intersections in the network; \( W_v \) is the set of downstream intersections of intersection \( v \), and \( T \) is the set of vehicle arrival times at intersections.

As will be shown in Section 3.4, it is not necessary to include the information from the downstream intersection with mild assumptions. By excluding the information at downstream intersection, we reduced the size of our state space by the order of \( O(|N| \times |T|) \), which can be significant when \( N \) is large. The simplified state space model becomes

\[
S = \{(u, v, t_v) : u \in U_v, v \in V, t_v \in T_v, T_v \subset T\}.
\] (5)
If the signal status is available to the traveler at real-time, we should have a better estimation of vehicle status at intersections. In this case, we include the signal status information into the state space, and it becomes

\[ S = \{(u, v, t_v, r_v) : u \in U, v \in N, t_v \in T_v, T_v \subset T, r_v \in [0, 1]\} , \]

where \( r_v = 0 \) if the signal status at intersection \( v \) is red at time \( t_v \), and \( r_v = 1 \) otherwise.

An action is the control to the system given current state. For an eco-routing problem, it is to choose which downstream intersection to visit. More precisely, an action is a function that maps the current state to a downstream intersection of current intersection, \( a(s) \in N \), where \( N \) is the set of intersections.

### 3.3. Transition probability

During a trip, a vehicle transits from state to state until it reaches to a set within a destination set, which is a set of states whose current intersection is destination intersection. The destination set \( D \) is defined as,

\[ D = \{(u, v, t_v) : v \text{ is destination intersection}\} . \]

A stage in an eco-routing problem starts from the arrival of an intersection and ends before the arrival of the downstream intersection. At the start of each stage, an action is chosen based on current state. Depending on the chosen action and current state, the system transits from current state to the next state with a known probability. For a state-action pair, there are usually more than one possible next state. The probability to reach a given next state is a function of current state and chosen action, which is a parameter of the model. Let’s denote by \( p(s'|s, a) \) the transition probability from state \( s \) to state \( s' \), given action \( a \) is chosen at \( s \). For the situation where intersections \( v \) and \( w \) are coordinated with offset \( \xi \) and background cycle length \( \gamma \), the transition probability can be calculated as follows (Sun and Liu, 2014),

\[ p(s'|s, a) = p(t' = (t + \delta_r + \delta_q + \tau - \xi) \mod \gamma|s, a) , \]

where \( \mod \) is the modulo operation; \( \delta_r \) is the red light delay at current intersection; \( \delta_q \) is the queuing delay at current intersection; \( \tau \) is current link travel time; \( \xi \) is the relative signal offset between current intersection and downstream intersection; and \( \gamma \) is the cycle length of the two intersections.

In Equation 8, only \( \delta_r, \delta_q, \) and \( \tau \) are random variables in this formula. So the essential information needed to construct the transition probability is \( p(\delta_r + \delta_q + \tau|s, a) \), i.e. the probability distribution of the sum of the intersection delay \( (\delta_r + \delta_q) \) and travel time \( (\tau) \) along the link. Sun and Liu (2014) has developed a method to estimate \( p(\delta_r + \delta_q + \tau|s, a) \) using high-resolution traffic data that are obtainable from inductive loop detectors and traffic controllers.

For cases where adjacent intersections are not coordinated, it is assumed that the possibility of arrival times at downstream intersections is uniformly distributed, i.e.

\[ p(s'|s, a) = \frac{1}{\gamma(w)} , \]

where \( \gamma(w) \) is the cycle length at downstream intersection \( w \).

### 3.4. Estimation of environmentally related costs

Associated with each transition from state to state, there is a cost. We want to find an optimal policy that minimizes the expected total cost to the destination. To achieve this goal, the first step is to calculate the cost for each step. In an eco-routing problem, the costs of interest are mostly environmentally related. They depend on many factors including vehicle characteristics, road characteristics, and traffic conditions, etc. All the information is used as input to a microscopic vehicle emission model.

Some of the information, such as vehicle characteristics (e.g. vehicle make, year) and road characteristics (e.g. grade), is static. So it is relatively easy to prepare these information for a vehicle emission model. Some of the other information, such as relative humidity and temperature, is dynamic, but they are mostly independent of traffic conditions. We assume all the information is available as inputs to a vehicle emission model.
One of the most important input to a microscopic vehicle emission model is second by second speed, i.e. vehicle trajectory. This is the most traffic related input to a vehicle emission model. Because of uncertainty in traffic condition, it is challenging to obtain such information for path search problem. This is especially true for signalized network because of the disruption to traffic from traffic signal controls. In this section, we will introduce a method to estimate vehicle trajectories based on the information of traffic signal status and vehicle arrival counts at intersections. It has been shown that these data are readily obtainable from the field and can be used for queue length estimation (Liu and Ma, 2009; Liu et al., 2009).

To estimate the vehicle trajectories in a signalized traffic network, we first make some assumptions about vehicle’s behaviors at intersections and on links. We first assume a vehicle either stops at an intersection or pass by the intersection with free flow speed. This is also the approach suggested by EPA for analyzing carbon monoxide of intersection project (EPA, 2010). When a vehicle starts to move from a stop, we assume it always accelerates from zero speed to free flow speed with constant acceleration rate. And when a vehicle decelerates, we assume it always make a full stop with constant deceleration rate. With these assumptions, a vehicle at an intersection can be in one of the two status: stop or travel at free flow speed; and a vehicle on a link can be in one of three statuses: deceleration, travel at free flow speed, or acceleration. Figure 1 gives an overview of vehicle status transitions at intersections and on links.

One more assumption is needed before we are able to calculate vehicle trajectories on links. When a link length is small and a vehicle stops at both ends of the link, it is possible that the vehicle needs to decelerate before it accelerates to its desire speed. But this situation may not happen so often, as short links usually appear in urban street network where speed limit is quite low. Consequently, it is reasonable to assume that links are long enough for vehicles to accelerate from zero speed to desire speed and then decelerate to zero speed. If this assumption is violated, we may over estimate the environmental costs on short links, which will be discussed in details at the end of Section 3.4.

Depending on a vehicle status at upstream and downstream intersections of a link, the trajectory of the vehicle on the link can be one of the following type (Figure 2):

- Type I: vehicle stops at both intersections;
- Type II: vehicle stops at upstream intersection;
- Type III: vehicle stops at downstream intersection;
- Type IV: vehicle stops at neither intersection.
Once trajectory type is determined, vehicle trajectory can be estimated given link length, acceleration rate, deceleration rate, and desire speed. From Figure 2, it can be seen that the vehicle trajectory on a link depend on the vehicle status at both ends of the intersections. This is the reason we have state space model specified by Equation 4 at the first place. But given the assumptions we have made above, it is possible to describe vehicle trajectories by only using the information at current intersection. The following paragraphs elaborate how.

To better understand a trajectory on a link, we divide a link into three parts according to trajectory type $I$, assuming acceleration rate, deceleration rate, and free flow speed are fixed. As shown in Figure 3, the first part of the link ($x_1$) corresponds to the acceleration process. It starts from the upstream intersection of the link and the length of the first part equals to the distance needed by a vehicle to accelerate from a full stop to free flow speed. The third part of the link ($x_3$) corresponds to the deceleration process. Its length equals to the distance needed by a vehicle to decelerate from free flow speed to a full stop. And it ends at the downstream intersection of the link. The remaining part of the link is the second part ($x_2$), on which the vehicle travels at free flow speed.

Let’s denote the length of a given link by $x$, free flow speed on this link by $\dot{x}$, acceleration rate on this link by $\ddot{x}_a$, and deceleration rate on this link by $\ddot{x}_d$. Simple physics allows us to determine the values of $x_1$, $x_2$, $x_3$ by the following equations:

$$
\begin{align*}
    x_1 &= \frac{\dot{x}^2}{2\ddot{x}_a} \\
    x_2 &= x - x_1 - x_3 \\
    x_3 &= \frac{\dot{x}^2}{2\ddot{x}_d}
\end{align*}
$$

As the values of $x$, $\dot{x}$, $\ddot{x}_a$, and $\ddot{x}_d$ are link specific, so do the values of $x_1$, $x_2$, $x_3$. 

---

Fig. 2. Vehicle trajectory types on links
The benefit of dividing a link trajectory into three parts is that we can disentangle the vehicle status at upstream intersection of a link from that at the downstream intersection of the same link. This simplification together with the previous assumptions allow us to calculate the step cost only with the information of current intersection. And we will use the state space model specified by Equation 5 from now on.

One technique to integrate emission model with transportation model is to use velocity/acceleration-indexed lookup tables. But this eliminates the time dependence in vehicle emissions on vehicle operation history, which can be significant to instantaneous emission values (Frey et al., 2001). To be as accurate as possible, the vehicle trajectories are used as inputs to the microscopic emission models in our method. But at the same time, some compromises are made in consideration to the feasibility and efficiency. Specifically, the complete trajectory along a route is divided into pieces link by link. Although correlations between link trajectories are modeled through vehicle status at intersections, they are put into microscopic model one by one and the time dependence in vehicle emissions between link trajectories is ignored. Furthermore, one link trajectory is divided into three parts, between which the time dependence in vehicle emissions is also ignored.

In our formulation, one step cost includes the cost at current intersection and the cost on immediate downstream link. The environmental costs at intersections are generated when a vehicle is idling. When traveling on a link, a vehicle can be in one of the three statuses on a link: acceleration, free flow speed travel, and deceleration. Considering this together with the link division mentioned earlier, we define the following variables,

- Link cost vector $C_{gh}^k \in \mathbb{R}^b$ represents vehicle emission and fuel consumption costs for a vehicle traveling on link part $g$ when it is in status $h$ at stage $k$. The values of $h \in \{1, 2, 3\}$ correspond to acceleration, free flow speed travel, and deceleration. Superscript $b$ denotes the number of environmental costs of interest.
- Intersection cost vector $I_k$ is a vector denoting the expected vehicle emissions and fuel consumption when idling at stage $k$. It is a function of idling emission/fuel consumption rate and idling duration.
Because we have divided a link into three parts, the shapes of the vehicle trajectory on different parts of the link do not affect each others. So, the environmental costs related to different parts are independent from each others. Since we don’t have information at downstream intersection at current state when using the state space model in Equation 5, we assume a vehicle always travels on the third part of the link at free flow speed. When we calculate the costs at the immediate downstream intersection, corrections to the costs are made accordingly, which will be described next.

When calculating the stage costs, the costs corresponding to the last part of a link is always first assumed to be $C^{32}$, as we assume a vehicle always travels at free flow speed during that time. In the case where the vehicle stops at current intersection, the difference in costs of last step will be corrected by adding back the difference ($\Delta C$) for the upstream link. More precisely, the costs at stage $k$ can be written as

\[
\begin{align*}
C_k &= C_k^{12} + C_k^{22} + C_k^{32} \\
C_k &= C_k^{11} + C_k^{22} - C_{k-1}^{32} + C_{k-1}^{33} + I_k & \text{if not stop}, \\
C_k &= C_k^{11} + C_k^{22} - C_{k-1}^{32} + C_{k-1}^{33} + I_k & \text{if stop}.
\end{align*}
\]

And one step cost $c(s, a)$, i.e. the step cost when action $a$ is taken in state $s$, becomes

\[
c(s, a) = c_0 \times p(\delta = 0|s, a) + \sum_{\delta \neq 0} c_1(\delta) \times p(\delta|s, a),
\]

where $\delta = \delta_r + \delta_q$ is the possible delays at current intersection.

As we discussed earlier, when a link is short and a vehicle stops at both ends of the link, the vehicle may have to decelerate before it accelerates to its desire speed. But the speed to which the vehicle needs to accelerate cannot be determined with the information only from one intersection, so we still use the full acceleration and deceleration process to calculate the cost, which is an overestimation of the actual cost. In a word, when links are so short that there is not enough space for a vehicle to accelerate to desire speed and decelerate to zero speed, the approximation method introduced here over estimates the actual costs.

3.5. Solving the problem by value iteration method

Value iteration method (Bellman, 1957) is a generic method for solving an MDP problem. As the eco-routing problem is formulated based MDP, it can also be solved by value iteration method. To do this, we first generate the state space and action set based on the network topology and signal control information such as cycle length and phase configuration. Then, we estimate the transition probability using historical high-resolution traffic data. The possible trajectories for each step is generated from link information, traffic signal information, and vehicle characteristics such as acceleration and deceleration rates. By supplying vehicle trajectories to the microscopic vehicle emission model, step cost for each state-action pair is obtained. Finally, we can solve for the optimal policy that minimizes the expected total cost to the destination. The value iteration method used here is similar to the one that is used by Sun and Liu (2014). For the sake of completeness, we describe it again in the following.

For stochastic shortest path problem with any initial conditions $u_0$, it has been shown that the sequence $u_m(s)$ generated by the following dynamic programming iteration (Bertsekas (1995), chapter 7.2 of volume I),

\[
u_{m+1}(s) = \min_{a \in A_s} E_{s'} [c(s, a) + u_m(s')],
\]

where $m$ is the index of the dynamic program, converges to $u^*(s)$ for each $s \in S$ and $u^*(s)$ satisfy the Bellman optimality equation

\[
u^*(s) = \min_{a \in A_s} E_{s'} [c(s, a) + u^*(s')].
\]

Two assumptions are required for the existence and uniqueness of the solution to Equation 14. In the context of the shortest path problem, it simply means 1), there exists at least one policy with which the target states will be reached with positive probability after finite number of stages (existence of proper policy); 2), the cost $c(s, a)$ for non-destination states are strictly positive. These assumptions are satisfied in formulated eco-routing problem. So the
MDP formulated earlier in this chapter can be solved by the value iteration method. One possible version of pseudo code for solving the MDP algorithm is given in Algorithm 1.

**Algorithm 1: Value iteration algorithm for infinite horizon MDP (Sun and Liu, 2014)**

**Data:** traffic network, signal settings, intersection delay distributions, link travel time distributions, step costs, destination (D)

**Result:** expected time cost at each state \((u(s))\), the navigation policy to the destination with minimal expected time cost \((\pi^*(s))\)

```plaintext
for each state \(s \notin D\) do
    \(u_0(s) = \infty\);
end

for each state \(s \in D\) do
    \(u_0(s) = 0\);
end

while \(\| u_m - u_{m-1} \|_\infty > \epsilon, \epsilon > 0\) do
    \(m = m + 1;\)
    for each state \(s\) do
        for each action \(a\) do
            compute \(Q_m(s, a) = c(s, a) + \sum_{s'} p(s'|s, a) u_{m-1}(s')\); 
        end
        compute and store \(\pi^*_m(s) = \arg\min_a Q_m(s, a)\);
        compute and store \(u^*_m(s) = Q_m(s, \pi^*_m(s))\);
    end
    return \(\pi^*_m(s), u^*_m(s)\)
```

In Algorithm 1, there is a state value \(u(s)\) corresponding to each state. At initialization, all the state values that are corresponding to destination states are set to zero; otherwise, the state values are set to be infinite. Then, the algorithm updates all state values according to Equation 13, until the difference between the consecutive iterations \(\| u_m - u_{m-1} \|_\infty\) is smaller than a threshold \(\epsilon\). The final state value for each state is the minimum expected total cost from the particular state to the destination state set. Optimal policy \(\pi^*_m(s)\) is obtained at the same time.

4. The constrained eco-routing problem

The eco-routing problem we formulated above only consider one cost at a time. But it is obvious that multiple costs can present in an eco-routing problem. For example, it is possible that people want to consider travel time and environmental costs at the same time. This brings us to the idea of a routing problem with multiple costs.

When there are multiple concerns in one problem, people usually formulate the problem as a multiple objective problem. One common approach to the multi-objective problem is to find an optimal solution to a problem with an objective of weighted average of different costs. Applying this approach to the model we developed in previous sections is straightforward, once the weights are known. Nonetheless, there is another way to solve the problem. One can specify a primary goal for the problem, and then consider other costs as side constraints. This is more appropriate in some cases. For example, a driver may want to minimized the fuel consumption while keeping the travel time under a given limit.

To solve this type of problem, we need to introduce the linear programming formulation of MDP. Based on the linear programming formulation, we can have a constrained method for the problem with multiple costs of interest, by converting objectives concerning specific pollutants into constraints. Linear programming is particularly suitable for this case. In a linear program, various costs can be considered as constraints, in addition to the primary objective, which is the case of the eco-routing problem with multiple costs.
To describe the problem, we still use the MDP developed earlier. Then, we convert the problem to a linear program. This allows us to add constraints related to costs. Finally, we find the solution to the constrained problem using standard solution methods for linear programs.

We first introduce the linear programming formulation of MDP for unconstrained problems. This formulation is similar to “extended TMD-model” described by Kallenberg (1983) in chapter 3 of his book. The term “TMD-model” is used to describe a Markov decision model who uses total reward criterion. The adjective “extended” specifies a model with an extra absorbing state. In our case, we should have a set of absorbing states, whose current intersection is the destination in our problem.

For each path search problem, there is at least a destination, which is denoted by \( D \subset S \). Take the model in Equation 5 as an example, each state in the model is described by \((u, v, t)\), where \( v \) denotes the current intersection. Then, we should change our model in the following way,

\[
\tilde{p}(s'|s, a) = \begin{cases} 
  p(s'|s, a), & \text{if } s = (u, v, t), v \notin D, \\
  0, & \text{otherwise}.
\end{cases}
\]

(15)

\[
\tilde{c}(s, a) = \begin{cases} 
  c(s, a), & \text{if } s = (u, v, t), v \notin D. \\
  0, & \text{otherwise}.
\end{cases}
\]

(16)

Note here \( c(s, a) \) is one component of vector \( c(s, a) \) that corresponds to the primary cost of interest. And we still use the same state space \( S \) and action space \( A \) as we do for the unconstrained problem.

To solve the problem, we use “ALGORITHM VI” introduced by Kallenberg (1983). We rewrite the algorithm using our notations, and call it Algorithm 2.

- step 1: Take any vector \( \beta \) such that \( \beta_{s'} > 0, s' \in S \).
- step 2: Calculate the optimal solution \( x^* \) of the following linear programming problem

\[
\text{Minimize } \sum_s \sum_a \tilde{c}(s, a)x_{sa} 
\]

subject to \[
\sum_s \sum_a (\delta_{s,s'} - \tilde{p}(s'|s, a))x_{sa} = \beta_{s'}, \quad s' \in S \\
x_{sa} \geq 0, \quad a \in A_s, \quad s \in S,
\]

where \[
\delta_{s,s'} = \begin{cases} 
  1 & \text{if } s = s' \\
  0 & \text{otherwise}
\end{cases}
\]

- step 3: the probability of chosen action \( a \) in state \( s \) is calculated as

\[
d(s, a) = \frac{x_{sa}}{\sum_a x_{sa}}
\]

(18)

It has been shown that any feasible solution to the problem has \( x_{sa} > 0 \) for exactly one \( a \in A_s \) for every \( s \in S \). So what we get is a pure and stationary policy (Kallenberg, 1983). In the linear programming formulation, the decision variables \( x_{sa} \) can be interpreted as the expected number of times action \( a \) is chosen in state \( s \). And the vector \( \beta \) is the initial distribution of the system.

This algorithm gives the optimal actions for all the states when the destination is specified. But it does not give the expected cost starting from a given state. The value of the objective function is the weighted expected cost from all states to the destination whose weights are given by vector \( \beta \). When the optimal policy is obtained, it is easy to recover the expected total cost for a given start state by assigning zero value at the destination states and backward propagating to the start state.
When we impose constraints on expected total cost on one or more costs of interest, a policy that is optimal for all initial states does not exist in general (Kallenberg, 1983). For our problem, we are interested in the constrained optimal policy with respect to a given initial state.

As we have constraints on the expected total cost, we can write the constraints in the following way.

\[ \sum_s \sum_a c_{s,a} x_{s,a} \leq \bar{b}_q, \]  

(19)

where \( c_{s,a} \) is the cost coefficient of \( q \)th cost when action \( a \) is taken in state \( s \) and \( \bar{b}_q \) is the given threshold of \( q \)th cost.

Because we only consider a given initial state for the constrained problem, we set the corresponding element of \( \beta \) to be 1, and all the other elements of \( \beta \) to be 0. That is

\[ \beta_{s'} = \begin{cases} 1 & \text{if } s' = s_0, \\ 0 & \text{otherwise}. \end{cases} \]  

(20)

where \( s_0 \) is the given initial state.

As we add additional constraints to the problem and allow components of \( \beta \) to be zero, it is no longer true that \( x_{s,a} > 0 \) for exactly one \( a \in A_s \) for every \( s \in S \). Consequently, the solution to the problem may not be pure.

To get the optimal stationary policy for a constrained MDP with total expected cost, we use the Algorithm 3, which is described below.

• step 1: initialize vector \( \beta \) according to Equation 20.
• step 2: Calculate the optimal solution \( x^* \) of the following linear programming problem

\[ \begin{align*}
\text{Minimize} & \quad \sum_s \sum_a \tilde{c}(s,a)x_{s,a} \\
\text{subject to} & \quad \sum_s \sum_a (\delta(s,s') - \tilde{p}(s'|s,a))x_{s,a} = \beta_{s'}, \quad s' \in S \\
& \quad \sum_s \sum_a c_{s,a} x_{s,a} \leq \bar{b}_q, \quad q = 1, 2, \ldots, b \\
& \quad x_{s,a} \geq 0, \quad a \in A_s, s \in S,
\end{align*} \]  

(21)

where \( b \) is the number of additional cost of interest.

• step 3: the probability of chosen action \( a \) in state \( s \) is calculated as

\[ d(s,a) = \frac{x_{s,a}}{\sum_a x_{s,a}} \]  

(22)

Following the solution obtained by Algorithm 3 will minimize the cost in the objective function on average if a vehicle starts from the given initial state. At the same time, other costs considered as side constraints will be less than or equal to the given threshold on average.

The linear programs in Algorithm 2 and Algorithm 3 can be solved using standard linear programming solution techniques, e.g. simplex method or inter points method. It has been shown that inter points method has worst case polynomial time complexity, while the worst case complexity of simplex method is exponential (see a overview of complexity on linear programming by Megiddo (1987)). In practice, however, both methods, especially the simplex method, have better performance. For our problem, the matrix is sparse as there are usually 2 or 3 non-zero transition probabilities for each state. This makes it relatively easy to solve the linear program.

5. Numerical examples

In this section, numerical examples will be presented using the signal data obtained from the City of Pasadena, California, USA. We will first give the field configurations of the examples, including data format, signal configuration,
and network layout. Then, we will solve an eco-routing problem with CO as objective, using our proposed method. The resulting optimal policy will be compared against the optimal path calculated from traditional shortest path algorithm, using link CO emission as cost. The comparison is carried out using a virtual probe approach, which will be discussed in details later. We choose to compare our method against traditional shortest path algorithm because: 1) there is no existing methods that are readily applicable to vehicle-actuated traffic signal controls, which is the case for our examples; 2) although traditional shortest path algorithm is incapable of dealing with traffic signals, it is familiar to a lot of people. Using it as a base line make it easier for people to appreciate our method. After this, an example of multiple costs in one problem will be given.

5.1. Experiment site configurations

For the purpose of numerical studies, a signalized traffic network consisting of 20 intersections is chosen from City of Pasadena, California, USA. 19 of these 20 intersections are controlled by traffic signals. The other one intersection at Cordova St and Oak Knoll Ave has stop signs installed for the north-south approaches. In Figure 4, the chosen network is represented by blue lines. All the signalized intersections are represented by green balloons, and the intersection controlled by stops is given by a red balloon.

Traffic signal configurations are given in Figure 5. Intersections are represented by cycles numbered from 1 to 20. Intersection 4 is the non-signalized intersection. The speed limits for all the east-west direction links are 35 mph and all the north-south direction links are 25 mph. Hudson Ave and Mentor Ave are one way streets.

For each intersection has 2 or 3 arriving directions. For each arriving direction, there are 60 or 80 possible arrival times, depending on the cycle length at the intersections. So the size of the state space should be less than $20 \times 3 \times 80 = 4800$.

Arrows in the circles indicate the coordination directions of signalized intersections. Apparently, Del Mar Blvd is well coordinated while it is not the case for Cordova St. Signal event data from four working days are used to generate red light delay distributions. These four days are Nov. 21, Nov. 22, Nov. 23 and Nov. 26, 2012. Data during morning peak hours from 7:00 to 9:00 on these days are used. The signal plans are the same during these times.

An example of generated red light delay distribution is shown in Figure 6. It is for the movement from intersection 16 to intersection 18 via intersection 17, which is the coordinated movement at intersection 17. There are two peaks in Figure 6. The peak around 16 seconds corresponds to the cycles where there is no pedestrian calls during the cycle, while the other peak has a higher value of delays due to the pedestrian calls.
Transition probabilities between different states can be calculated by Equation 8, given red light delay distributions and queue delay distributions. The red light delay distribution can be calculated using high-resolution signal data. The calculation of queue delay distribution also requires high-resolution vehicle arrival data, which we don’t have in this example. So queue delays are ignored. One advantage of the proposed method is that it still works even if some of the information is not available. The impacts of the missing information, however, may need further examination. We leave this topic to future research.

5.2. CO emission as objective

As an example, we solve an eco-routing problem with CO emission as objective. We will first find an optimal path using a traditional shortest path algorithm, Dijkstra’s algorithm. When applying Dijkstra’s algorithm, link costs are CO emissions calculated by assuming free flow speed traveling on links. CO emission at intersections is not considered, as the algorithm is not designed to incorporate costs at intersections. This optimal path is used as a
baseline, against which the optimal policy from the proposed method will be compared later. In the comparison, CO emission at intersections will be considered in both cases, as we are able to calculate the actual cost from field data.

We assume the free flow speed on links is the same as posted speed limit on the links, which are 25 \text{ mph} for north-south directions and 35 \text{ mph} for east-west directions. We also assume the acceleration rate is 4 \text{ ft/s}^2 and deceleration rate is 10 \text{ ft/s}^2. Based on these assumptions, the CO emission parameters are calculated using CMEM model and given in Table 2.

![Fig. 7. Optimal policy to minimize expected total CO emission](image)

Table 2. CO emission parameters

<table>
<thead>
<tr>
<th>Speed</th>
<th>$c_{11}(g)$</th>
<th>$c_{12}(g)$</th>
<th>$c_{22}(g)$</th>
<th>$c_{33}(g)$</th>
<th>cruise rate (g/s)</th>
<th>idle rate (g/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>35 mph</td>
<td>1.0326</td>
<td>0.0950</td>
<td>0.0380</td>
<td>0.0394</td>
<td>0.0148</td>
<td>0.0041</td>
</tr>
<tr>
<td>25 mph</td>
<td>0.5254</td>
<td>0.0300</td>
<td>0.0120</td>
<td>0.0229</td>
<td>0.0065</td>
<td>0.0041</td>
</tr>
</tbody>
</table>

We will use a virtual probe approach for the comparison. In this approach, an imaginary vehicle travels in the network during the time when data are available, and corresponding intersection delays and environmentally related costs are calculated using available signal information. This approach works in the following way. In the case of the optimal policy, let a vehicle starts to travel from a given intersection at a given time. The time for the vehicle to arrive at the next intersection can be calculated by assuming the vehicle travels at free flow speed. Then, the optimal action is chosen based on the arrival time at the next intersection. Given the signal data is known at that time, the corresponding intersection delays can be calculated, and consequently, the following start time for the next link is known. At the same time, corresponding environmentally related costs are calculated using the method described in Section 3. This process can be repeated until the vehicle reaches to the destination. The total cost is the sum of the cost at steps during the whole process. For the optimal path from Dijkstra’s algorithm, the process is similar except there is no action choice at each intersection based on the arrival time. The vehicle just follows the optimal path given by the Dijkstra’s algorithm.

If the destination is set to be intersection 20 (Del Mar Blvd at Hill Ave), the optimal path from intersection 1 calculated by Dijkstra’s algorithm is shown by purple dash line in Figure 7; the optimal policy for CO emission to start from the intersection 1 is given by arrows in Figure 7. Red arrows show the optimal policy when going east from intersection 1 and green arrows give the optimal policy when going south. At some of the intersections, there are more than one optimal actions, which are dependent on arrival time at that intersection. The optimal actions depending on arrival times at intersection 2 are given in Figure 8. Apparently, the optimal policy prefers the well coordinated path (Del Mar Blvd), comparing to the optimal path calculated by Dijkstra’s algorithm.

According to the optimal path or optimal policy, we carry out 6 experiments using virtual probe approach introduced above. Data during morning peak hours of Nov/21/2012, Nov/22/2012, and Nov/23/2012 are used. For each experiment, we imagine that vehicles start to travel from intersection 1 at every minute during a 30 minute time period. So there are 30 runs for each experiment. For each day, we have conducted two such experiments. One starts from
7:10 am and the other starts from 8:10 am. The average travel time reductions from tradition shortest path algorithm by using our proposed method are shown in Table 3.

Table 3. Average CO emission reduction by using proposed method

<table>
<thead>
<tr>
<th>Time</th>
<th>Nov/21/2012</th>
<th>Nov/22/2012</th>
<th>Nov/23/2012</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:10 am</td>
<td>-19.93%</td>
<td>-30.63%</td>
<td>-20.28%</td>
</tr>
<tr>
<td>8:10 am</td>
<td>-23.67%</td>
<td>-25.85%</td>
<td>-18.31%</td>
</tr>
</tbody>
</table>

From Table 3, it can be seen that by following the optimal policy from the proposed method gives significant improvement on average CO emission over traditional shortest path algorithm. The improvement mainly comes from the additional information from traffic signals. The traditional shortest path algorithm ignores such information when searching for the optimal path, while the proposed method explicitly accounted for this factor. This leads less expected stops at intersections, and thus CO emission is reduced on averaged. It should also be pointed out that the proposed method is only better on average, which means the CO emission may be higher by following the optimal policy for some runs.

5.3. The constrained eco-routing problem

In this section, we will solve a constrained eco-routing problem using the method in Section 4. We still use the same network from City of Pasadena with the same settings. One prerequisite of solving the constrained problem is to determine the constraint constants ($\tilde{b}_q$). In this example, we first calculate the optimal value for each cost of interest and then add some buffer to each one of the costs to get the constraint constants.

We still use intersection 20 as destination. The optimal costs for starting from intersection 1, arriving at intersection 2 at cycle time 1, to get to the destination are given in row 1 of Table 4. These optimal values are obtained by solving unconstrained problems for each cost of interest.

Next, we solve a problem with minimizing travel time as the primary objective. The costs of other pollutants are used as constraints. The constraint constants are given in row 2 of Table 4. The first column of the second row gives the optimal expected travel time, given the constraints on other costs. It can be seen that we are no longer able to achieve the optimal travel time when constrained by other costs. But it is also obvious that the difference between the constrained optimal value and the true optimal value is quite small. This means the costs considered here do not contradict to each others lot in our problem settings.

Another interesting thing is that the optimal solution for the constrained problem is no longer pure. We get randomized action at intersection 2, as shown in Figure 9. It is best to go straight about 90% of the time and turn right about 10% of the time when arriving at that intersection at cycle time 1.
Table 4. Optimal values and constraints constant of different costs

<table>
<thead>
<tr>
<th></th>
<th>time (s)</th>
<th>fuel (g)</th>
<th>CO (g)</th>
<th>CO₂ (g)</th>
<th>HC (g)</th>
<th>NOₓ (g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>optimum</td>
<td>211.8404</td>
<td>234.0996</td>
<td>6.5032</td>
<td>731.203</td>
<td>0.337</td>
<td>0.5695</td>
</tr>
<tr>
<td>constraint</td>
<td>224.9417*</td>
<td>260</td>
<td>6.7</td>
<td>735</td>
<td>0.35</td>
<td>0.59</td>
</tr>
</tbody>
</table>

*optimal expected travel time given the constraints on other costs.

6. Conclusion

In this paper, an eco-routing algorithm is developed for a signalized traffic network. Because vehicle behaviors are highly dependent on traffic situation at intersections, vehicle trajectories are estimated using traffic information at intersections, including traffic signal and vehicle arrival information. Then, vehicle trajectories are used as inputs to well developed microscopic vehicle emission models, like CMEM and VT-Micro, to calculate corresponding vehicle emissions. A method to incorporate the environmentally related costs into a Markov decision process (MDP) based vehicle routing algorithm is proposed to find the optimal policy that guides vehicles to the destination with minimum expected environmentally related cost.

This paper also developed a method to solve the eco-routing problem where multiple costs of interest are considered. This is done by converting the original MDP formulation to a linear programming formulation. And then consider additional costs as constraints. This approach is useful in the cases when there is a primary objective, but additional costs also need to be considered.

At the end of the paper, two numerical examples are given. In the first one, the proposed method for single objective eco-routing problem is compared against the traditional shortest path algorithm, using a virtual probe approach with the field data from City of Pasadena, CA, USA. The results have shown that the proposed method significantly reduces CO emission on average. The second example has demonstrated the constrained eco-routing problem and how additional constraints affect the optimal value of the primary objective.

This paper tries to solve an eco-routing problem using traffic information that is readily available from the field. Although the initial results are promising, there are certainly ways for improvement. For example, we have used a virtual probe approach to show the effectiveness of the proposed method in this study. But the performance of the proposed method needs further scrutiny by a real field study. From methodology point of view, we don’t use GPS-based data in the proposed method. But GPS-based vehicle trajectories are invaluable to eco-routing problems, it would be great to develop way to incorporate GPS-based data in our method when they are available. One possible way is to associate these data with signal status and traffic state information. Then, instead of using predetermined
vehicle parameters, such as free flow speed, constant acceleration and deceleration rates, we can use real world vehicle trajectories for cost estimation at each step.

References


