Reduction

2011-2014

Deliverable D3.2

Report on Eco-Routing Computation Techniques

10-12-2012

Public Document

Project acronym: REDUCTION

Project full title: Reducing Environmental Footprint based on Multi-Modal Fleet management Systems for Eco-Routing and Driver Behavior Adaptation

Work Package: **D3.2**

Document title: Report on Eco-Routing Computation Techniques

Version: **1.1**

Official delivery date: **2012.08.31**

Actual publication date: 2012.08.31

Type of document: Report

Nature: Public

Authors: Chenjuan Guo (AU), Bin Yang (AU), Christian S. Jensen (AU), Manohar Kaul (AU), Yu Ma (AU), Kristian Torp (AAU), Ove Andersen(AAU)

Approved by: _____

Version	Date	Sections Affected
1.0	2012.08.15	Initial version
1.1	2012.12.10	Review comments processed
1.2	_____	Updated to reflect changes in the implementation
1.3	_____	Updated to reflect changes in the implementation

Executive Summary

The purpose of this report is to cover key techniques invented for D3.2 of WP3, whose primary objectives are summarized as follows.

- Convert the data available (primarily GPS data) into existing metrics for the capture of environmental impact, e.g., greenhouse gas emissions.
- Evaluate and validate the results provided by the eco-routing techniques proposed. Here the use of fuel consumption and other vehicle data is expected to be used. This evaluation and validation must support the field trials carried out in WP5. Extensive experimental studies are expected where alternative routing and evaluation techniques are assessed and improved.
- Propose basic eco-routing techniques, and ensure that the eco-routing techniques invented are conceptually sound, correct, and offer adequately and predictably accurate environmental impact estimates. Provide output map formats that are easy to integrate with existing mapping solutions so that the eco-routes can be easily included in reports to transport authorities or offered as feedback to the individual drivers.

In order to make fuel estimations, we used some of the existing well-known and widely accepted vehicular environmental impact models available in the transportation science community. To better understand the existing and related work available, we conducted a study that compared the most popular environmental impact models and classified them (highlighted in II below). During this study, we realized the significance of ascertaining the slope/grades of road segments over which the vehicles traverse. As most of the existing road maps were only 2D, for e.g. OpenStreet Maps, we decided to fill this gap by generating our own 3D road network which would be highly accurate and would allow us to make better fuel and emission estimations (highlighted in I below). Finally, techniques for basic eco-routing based on appropriate vehicular environmental impact models and shortest routes algorithms are developed (highlighted in III below)..

Techniques invented and implemented for D3.2 of WP3 are reported as follows.

I. A 3D Spatial Network. The report describes a 3D spatial network that provides grade information for all road segments. This network is constructed by using a laser scan point cloud for lifting a 2D spatial network. The use of laser point data yields a model with high accuracy.

- In the report, we propose a novel filtering and lifting framework that augments a standard 2D spatial network model with elevation information extracted from massive aerial laser scan data and thus yields an accurate 3D model.
- We investigate a filtering technique that is capable of pruning irrelevant laser scan points in a single pass, but assumes that the 2D network fits in internal memory. We also provide an external-memory filtering technique that makes no such assumption and is scalable.

- During lifting, a triangulated irregular network surface is constructed from the remaining points and is projected onto the 2D network. By means of interpolation, the 2D network is lifted into 3D.

- We report on a large-scale empirical study that offers insight into the accuracy, efficiency, and scalability properties of the framework.

II. An Evaluation of Vehicular Environmental Impact Models. The report investigates a number of models that have been proposed with the aim of quantifying the emissions of a vehicle based on GPS data from the vehicle and a 3D model of the spatial network the vehicle travels in.

- We develop an evaluation framework, called EcoMark, for such environmental impact models.

- We survey all eleven state-of-the-art environmental impact models known to us, which do not require special characteristics of the vehicles.

- To gain insight into the capabilities of the models and to understand the effectiveness of the EcoMark, we apply the framework to all models.

- Intensive experiments are conducted to show the advantages and disadvantages of each environmental impact model. Scenarios where specific models are suitable are also discussed and presented.

III. Techniques for Basic Eco-routing Tasks. The report describes techniques used for the basic eco-routing techniques. The algorithm is developed based on the classical routing algorithm, i.e., Dijkstra's algorithm.

- A graph for modeling spatial network is proposed that records information about road intersections and road segments, and measures the length of each road segment, the travel time required to traverse the road segment, and fuel consumption or GHG emission on the road segment.

- The implementation of the graph is based on OpenStreetMap, and thus formats of the proposed network graph and OpenStreetMap can be transformed between each other.

- Based on the proposed network graph, the classical Dijkstra's algorithm is adapted to develop the eco-routing techniques.

- Intensive experiments are conducted. The classical Dijkstra's algorithm and the eco-routing Dijkstra's algorithm are utilized to compute the shortest routes, the fastest routes and the eco-routes. Three types or routes are compared with each other. Different types of routes can be easily visualized.

Table of Contents

1. Introduction

This section identifies the high-level objectives and targets of the project in Section 1.1, followed by the objectives and contributions of work package WP3 in Section 1.2. Section 1.3 shows the objectives and contributions of this deliverable.

1.1 Project Overview

The reduction of greenhouse gas (GHG) emissions from transportation is essential for achieving politically agreed upon emissions reduction targets that aim to combat global climate change. A major portion of these emissions is identified to be from vehicular transportation.

The REDUCTION project aims at using advanced ICT solutions to manage multi-modal fleets and reducing their overall environmental impact. The objectives can be summarized as:

1) Optimizing driving behavior through effective decision making for the enhancement of drivers education and the formation of effective policies about optimal traffic operations (speeding, braking, etc.), based on the analytical results over the data that associate driving-behavior patterns with CO2 emissions.

2) Eco-routing to recommend the most environmental-friendly routes that reduce overall fuel usage and CO2 emissions.

3) Multi-modality that allows fleets operating based on various transport modes and take decisions about how to: a) balance capacity within and throughout modes of transport, b) reduce their environmental footprint by comparing consumption data and alternative routing information that involve mode changing, c) develop efficient driver policies for their entire set of vehicles and not only for isolated modes.

The project aims to provide accurate and highly scalable eco-driving/routing solutions that aim at substantially reducing the overall CO2 emissions. Scientific papers are also expected to be produced to disseminate knowledge collected during the course of this project.

1.2 Work Package Objectives and Tasks

The objective of WP3 is to design and develop a software prototype that can convert vehicle-related data, primarily GPS data, to metrics that capture environmental impact. The prototype must handle very large volumes of data from different types of vehicles and must efficiently compute the multi-modal eco-routes in both real-time and offline modes. In addition, the prototype must be able to report on the temporal evolution of eco-routes, e.g., due to a variety of changes in the transportation infrastructure and its use. The work package will (a) define the interfaces for how vehicles communicate with the server side and with each other; (b) invent and prototype techniques for computing eco-routes; (c) invent and prototype techniques for the validation of eco-routes; and (d) design and prototype high-performance data structures and algorithms for the handling of very large volumes of streaming data from the vehicles.

1.3 Objective of this Deliverable

This report describes basic eco-routing techniques invented and implemented in D3.2 of WP3.

Particular effort is contributed to constructing infrastructure necessary to the eco-routing techniques. Specifically, the availability of accurate 3D models of spatial networks can enable substantial improvements in vehicle routing. Notably, such models enable eco-routing in a way that information of uphill and downhill roads can be provided, and thus eco-routing can be carried out more accurately than in 2D spatial networks.

It is also necessary to be able to reliably quantify the emissions of vehicles as they travel in a spatial network. This information allows weights that reflect emissions of road segments to be assigned to a spatial network, and, in turn, enables the application of the classical routing algorithms, e.g., Dijkstra and A* algorithms, to eco-routing.

A graph for modeling a spatial network is proposed that records information about road intersections and road segments, and measures the length of each road segment, the travel time required to traverse the road segment, and fuel consumption or GHG emission to traverse the road segment. The classical Dijkstra's algorithm, which is originally used for searching the shortest routes, is adapted to conduct the basic eco-routing tasks.

Objective of Task	Proposed Methodology
Lifting a 2D spatial network to 3D	Lifting 2D OpenStreetMap road network by using 3D laser scan data, using novel algorithms to perform this task in fast, accurate and efficient manner.
Evaluating Models of Vehicular Environmental Impact	A thorough classification and analysis of 11 state-of-art vehicular environmental impact models.
Basic Eco Routing	Eco-weights are computed based on aggregated vehicular environmental impact models. The classical Dijkstra algorithm is applied to conduct basic eco routing.

Table 0: Summary Table

2. Related Work and Reduction

2.1 3D Road Networks

As to provide eco-routing service, we have to precisely model the real transportation network. A typical transportation network can be viewed as a graph, the road segments are the edges in the graph and the intersection points of the road segments are the vertex in the graph. For now we have various map service website which provide 2D modeling of the real transportation network. Intuitively, if a vehicle goes uphill, it consumes more fuel and emits more GHG, in contrast, if a vehicle goes downhill, it produces less GHG. So if we want to get more accurate GHG emissions estimation from the spatial network, we have to attach the slope information of the road segments to the current spatial network. With the help of land surface laser scan data, it is possible to add the 3D information of the road segments to the existing 2D spatial network. For now, only little work has been conducted on the creation of 3D spatial networks. To the best of our knowledge, this section presents the first work that obtains a 3D spatial network by lifting a 2D spatial network using laser scan data.

Tavares et al. [3] study eco-routing based on a 3D network and find that eco-routes are 1.8% longer than the corresponding shortest routes and that fuel cost savings are in the range 8–12%. Their 3D spatial network is developed based on contour lines, which are of much lower resolution than is the laser scan point cloud that we use.

The most widely used 3D models that represent spatial networks or cities are generated from Shuttle Radar Topography Mission (SRTM) data [8], which are latitude-longitude raster data where the finest available resolution is 30 m2 and where a single elevation value is stored per cell. Over et al. [9] generate a 3D city model using OpenStreetMap data and SRTM data for the purpose of urban planning. The main focus is on augmenting buildings and landmarks in a city with elevation and shape information, rather than on identifying the grades and elevations of the spatial network of a city. We note that the relatively low resolution of SRTM data renders it difficult to augment spatial networks with proper grades and elevations. In contrast, the laser scan data we use has one 3D point per square meter and is of a much higher resolution.

Other recent works that employ TINs to obtain accurate height information include studies of watershed on terrains for hydrological analyses and flood prediction [10, 11]. The TIN surfaces are generated from SRTM data and contain nearly 100K triangles.

Several methods have been proposed that use statistical or machine learning techniques to classify aerial laser scan points into different categories [13, 14]. We only consider a particular category of points, namely ground points, which capture the actual land surface, and exclude other points representing, e.g., vegetation, water, buildings, and noise. In the remainder of the paper, when mentioning laser scan points, we refer only to ground points.

2.2 Vehicular Environmental Impact Models

As to provide eco-routing service, we have to know how to estimate GHG emissions from the historical data and the 3D spatial network. Although CANBus data provides detailed information of the vehicle speed, location, fuel usage and so on, it's impossible to get abundant CANbus data and it's also very hard to interpret CANbus data, so we choose to estimate the vehicular environmental impact with GPS tracking data which is easy to get and use. For now we have a massive GPS tracking data set and there are a bunch of models could be utilized to estimate the environmental impact of the vehicles. Although there are many existing environmental impact estimation models, there's no comprehensive and analysis of them. As to choose the most suitable models for our own purpose, we have to study and analyze the existing models to see their behaviors, thus we can get the consistency and difference between all the models. In this section, we will show a brief summary of the existing models, including required input of the models, classification of the models, etc.

2.2.1 Variables that influence vehicular environmental impact

A few literatures [33, 37, 39] summarize variables and parameters that influence vehicular environmental impact as follows.

- Vehicle technology factor includes general vehicle design characteristics (e.g., weight, aerodynamic efficiency, model and size), type of fuel, engine power, etc.
- Vehicle status factor includes mileage, age and engine status, etc.
- Vehicle operating factor includes vehicle speed and acceleration, power demand, engine speed, gear selection, etc.
- Driver-related factor includes driving habit and behaviours, etc., and is reflected via vehicle operating factor.
- Air condition factor includes ambient temperature, atmospheric pressure, air humidity and wind effects, etc.
- Road condition factor includes the road grade (i.e., slope), curve, surface roughness, and pavement quality, etc.
- Traffic condition factor includes vehicle-to-vehicle and vehicle-to-control interaction, etc.

2.2.2 A classification of models for estimating vehicular environmental impact

The methods for estimating fuel consumption have been developed for several years. In the context of urban transportation network, the methods are classified into macroscopic scale and microscopic scale [48].

Macroscopic Scale Estimation Models

Macroscopic models account for the total emission for a given time period (e.g., a day, a week or a year) and a given area (e.g., a region, a city or a country) [39]. These models differ from microscopic models mainly in that they utilize the aggregated characteristics (e.g., the average speed, the total length of a trip) to estimate the emission rate, while the microscopic models use instantaneous characteristics. In addition, the macroscopic models also consider other factors, e.g., air conditions, engine power, described in Section 1.2.1. As described in a review [37], "*the emission factors are calculated as mean values of measurements on a number of vehicles over given driving cycles, and are usually stated in terms of the mass of pollutant emitted per vehicle and per unit distance (g vehicle^{-1} km^{-1}) or per unit of fuel consumed (g litre^{-1})*".

Examples of the state-of-the-art macroscopic models are: MOBILE6 [43] and its update MOVES [44] developed by U.S. Environmental Protection Agency (EPA), COPERT 4 [42] developed by European Environment Agency (EEA), EMFAC [5] developed by the California Air Resources Board (ARB), EMEP/CORINAIR Emission Inventory Guidebook [68] conducted by EEA (European Environment Agency), and SIDRA Intersection [69] developed by Sidra Solutions Australia, GreenGPS [71] developed by Ganti et al..

Most macroscopic models, except SIDRA Intersection, require comprehensive input data for estimating fuel consumption, such as vehicle technology and operating factors and air condition factor described in Section 1.2.1. On the other hand, SIDRA Intersection provides a list of values for vehicle-related parameters, which can be used as default settings of vehicles, and thus it only takes as input vehicle speed and acceleration and road grade to estimate the fuel consumption.

Over the years, the macroscopic models are used for large-scale transportation analyses and cases, where the average speed is adequate to characterize traffic conditions. These models are suitable to be used in applications where emission prediction at a coarse granularity is expected. However, due to kinds of limitations exhibited in the macroscopic models [37, 48], they are considered not sufficient to accurately estimate the fuel emission of modern vehicles.

- Various vehicle operations (e.g., vehicle speed, vehicle acceleration, gear change pattern) may lead to the same average speed. Using the single average speed is not sufficient to characterize the precise driving behaviour of a vehicle, and therefore, not accurate to estimate the emission level of a vehicle.

- Average speed becomes a less reliable factor for estimating the emission of modern vehicles, because emission of the modern vehicles often occur during gear changes or rapid acceleration and deceleration.

- Average speed cannot capture the detailed spatial information (e.g., intersection) of a road network, and thus do not allow accurate emission estimations of vehicles at small segments of a road network.

Microscopic Scale Estimation Models

Microscopic models measure instantaneous fuel consumption and emission rates of an individual vehicle at a given time using, e.g., instantaneous speed and acceleration/deceleration rate. The time interval for estimation is usually one sec. More comprehensive estimations also involve vehicle factors (e.g., gear ratios, air drag coefficient, mass), road conditions (e.g., road grade) and driving modes (acceleration, cruise, deceleration, idle), etc. Microscopic models can be classified into three general categories, specifically, emission maps, regression-based models and load-based models [39].

Emission maps look up velocity-acceleration matrices for querying fuel consumption. An emission function can be built as a two-dimensional matrix, where one dimension represents the velocity range and the other represents the acceleration range. Each cell of the matrix stores instantaneous fuel consumption given particular velocity and acceleration at that time. Although emission maps are simple and easy to use, they can be sparse and too sensitive to the driving cycles [72].

Regression-based models usually adopt mathematic functions of second-by-second speed and acceleration of a vehicle to predict its instantaneous fuel consumption. Physical parameters of a vehicle are not required in the input of the models [40, 47, 60].

Load-based models are the most comprehensive instantaneous models for fuel consumption estimation, as they consider a wide range of parameters related to the fuel consumption, such as vehicle second-by-second speed and acceleration, grade, air condition, engine max power, gear ratio, engine power demand, etc. Example models are: CMEM [70, 61] developed by the University of California-Riverside, VeTESS [55] delivered from European Commission 5th Framework DECADE project, PERE [53] developed by U.S. EPA, PHEM [45] developed during the European Cooperation in the field of Scientific and Technical Research (COST) 346, VT-CPFM-1 and VT-CPFM-2 [58] developed on the basis of collaboration between Virginia Tech Institute, NAVTEQ and K.U. Leuven, Belgium.

Microscopic models are often employed to evaluate the environmental impact of individual road segments on a spatial network and particular driving operations. The comprehensive load-based models generally offer the best estimates of fuel consumption. However, the required parameters are difficult to obtain for individual vehicles in a scalable manner. In contrast, regression-based models are fairly easy to apply because their input, e.g., instantaneous velocities and accelerations, can be obtained directly from GPS trajectories. EcoMark aims to evaluate the environmental impact of travel using models that do not require vehicle-specific factors. The details of the qualifying models are covered in Section 1.4.

3. Lifting a Spatial Network Using Laser Scan Data

3.1 Introduction

While current vehicle routing systems rely on 2D spatial networks, e.g., Google Maps, Bing Maps, and OpenStreetMap, future generations of vehicular routing systems are likely to use much more accurate models of transportation infrastructures. The benefits of using higher-fidelity models are many.

Notably, by capturing not just the latitude and longitude (i.e., the (x, y) coordinates) of the embedding into geographical space of a transportation network, but capturing also altitude or elevation (i.e., (x, y, z)), thus obtaining a 3D model, a foundation for eco-routing is obtained.

A transportation study finds that eco-routing that uses a 3D spatial network model, can yield fuel cost savings of 8–12% when compared to standard routing based on a 2D model [3]. Another study reports that the use of a 3D model built from aerial laser scan data for vehicle routing will yield annual fuel savings of approximately USD 6 billion in USA [4]. This figure stems from TomTom, a worldwide leading manufacturer of navigation systems.

Next, seemingly futuristic technologies like autonomous vehicles are becoming reality. In June 2011, the Nevada legislature passed AB 511 [5] that authorizes the use of autonomous vehicles. Google's autonomous cars [6, 7], equipped with LiDAR (Light Detection And Ranging) sensors, are able to scan nearly 200 feet in all directions to obtain a 3D view of the cars' surroundings that is accurate to within 2 cm. When combined with an accurate 3D network model, accurate positioning results.

Over the past years, aerial laser scanning has become the preferred method for constructing Digital Elevation Models (DEMs). In laser scanning, a point cloud consisting of (x, y, z) samples (see Figure 1) of the surface of geographical space are obtained from a low-flying aircraft. Samples are typically collected at a 10–100 KHz frequency, and the spacing between samples may be below one meter.

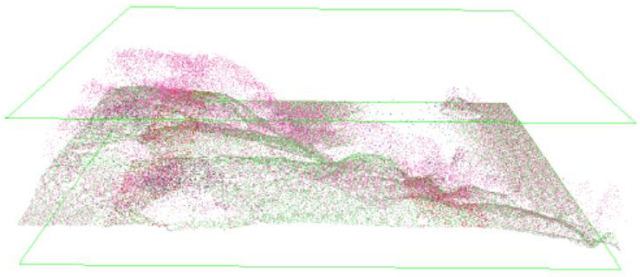


Figure 1: Example Laser Scan Point Cloud

This section studies the use of aerial laser scan data for obtaining a 3D spatial network model from a conventional 2D spatial network model. In particular, the spatial network from OpenStreetMap is used as the underlying 2D model. This open-source network is freely available and is both quite

accurate and complete for many countries.

Different representations of geographical space exist, including point clouds, triangulated irregular networks (TINs), and planar structures. A key challenge is to obtain a suitable balance between the accuracy of a model, as dictated by the intended applications, and the storage space required by the model. This paper addresses that challenge by proposing techniques that are capable of using commodity computing hardware for generating accurate and compact 3D spatial network models from massive laser scan point clouds and a 2D spatial network.

To the best knowledge of the authors, this is the first work that investigates how to apply a 3D aerial laser scan point cloud to the lifting of a 2D spatial network, thus obtaining a 3D spatial network. Specifically, our contributions are four-fold.

1 We propose a novel filtering and lifting framework that uses an aerial laser scan point cloud for lifting a spatial network.

2 Two alternative filtering techniques are proposed for obtaining the particular laser points from a laser scan point cloud that are needed for the lifting. A one pass filter makes the assumption that the 2D spatial network fits entirely into main memory, while an external memory based filter makes no assumption about the size of the 2D spatial network and hence is scalable.

3 Techniques for spatial lifting, consisting of triangulation and interpolation, are proposed to augment the 2D spatial network with elevation information using the remaining laser points.

4 We present a comprehensive and large-scale empirical study that offers insight into the accuracy, efficiency and scalability properties of the framework. Compared with previous studies using 3D data for other purposes, our study involves data sets that are orders of magnitude larger [10, 11, 12].

The remainder of this section is organized as follows. In Section 1.2, we survey related work. Section 1.3 presents a formal problem definition, including the proposed filtering and lifting framework. Section 1.4 details the filtering and lifting phases. Section 1.5 covers the empirical analyses of the proposed techniques. Finally, Section 1.6 concludes the section.

3.2 Preliminaries

We introduce definitions that underlie the proposed problem, formalize the spatial network lifting problem, and provide an overview of the filtering and lifting framework. An overview of the notation used in the paper is provided in Table 1.

Notation	Description
G_{2D}	A 2D spatial network.
G_{3D}	A 3D spatial network.
P_c	A 3D laser scan point cloud.
p_i	A 3D laser scan point in P_c.
$prj^{\top}(p_i)$	The projection of a 3D point p_i onto the 2D plane.
$prj^{\perp}(g, \triangle)$	The projection of a 2D model element g onto a TIN $\triangle$.
$\epsilon N(g)$	The ϵ-neighborhood of a 2D model element g.

Table 1: Notation for 3D Spatial Network Model

3.2.1 Data modeling

A spatial network captures both the topology and the embedding into geographical space of a transportation network. We define 2D and 3D spatial networks next.

Definition 1. A **2D spatial network** is modeled as an undirected graph G_{2D} = (V, E, F_{2D} , H_{2D}), where V and E is the vertex set and edge set, respectively, and F_{2D} and H_{2D} are the functions recording the embedding of vertices and edges into the 2D plane, respectively.

A vertex $v_i \in V$ indicates either a road intersection or the end of a road. An edge $e_k \in E \subset 2^V$ is defined as a set of two vertices, and represents a road segment connecting the two vertices. For example, edge $e_k = \{v_i , v_j\}$ represents a road segment that connects vertex v_i and vertex v_j. Function F_{2D}: $V \rightarrow R^2$ takes as input a vertex and returns its coordinates in the 2D plane. Function H_{2D}: $E \rightarrow R^2 \times \ldots \times R^2$ takes as input an edge e, and outputs a 2D polyline represented by a sequence of 2D points. For example, edge $H_{2D}(\{v_i, v_j\}) = (v_i , a_1 , a_2 , a_3 , a_4 , v_j)$, as shown in Figure 2.

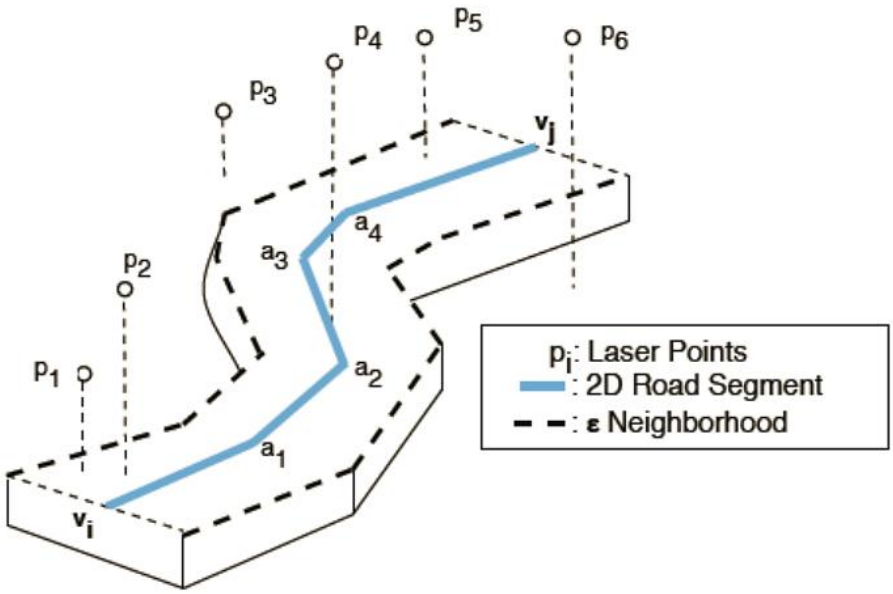


Figure 2: A 2D Road Segment With its ε-Neighborhood

Definition 2. A **3D spatial network** is also modeled as an undirected graph $G_{3D} = (V, E, F_{3D}, H_{3D})$. The definitions of V and E in a 3D spatial network are identical to the counterparts in a 2D spatial network. However, functions F_{3D} and H_{3D} record the geometric information of vertices and edges in 3D space. Function $F_{3D}: V \rightarrow R^3$ takes as input a vertex and returns its coordinates in 3D space; and function $H_{3D}: E \rightarrow R^3 \times \ldots \times R^3$ takes as input an edge and outputs a 3D polyline, represented as a sequence of 3D points. Figure 3 shows the 3D polyline of a road segment.

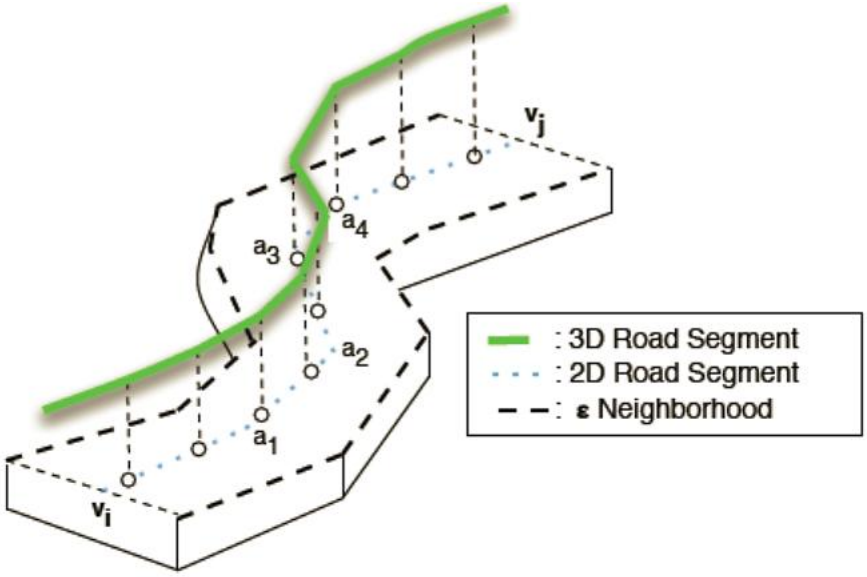


Figure 3: A 3D Road Segment

Definition 3. A Laser Scan Point Cloud (P_c) is a set of laser points p_i, which is formalized as

$$P_c = \{p_i = (x_i, y_i, z_i) \in R^3 \mid 1 \leq i \leq N\},$$

where N is the total number of laser points in the point cloud.

3.2.2 Problem formulation

Spatial network lifting augments a 2D spatial network G_{2D} with elevation information extracted from a laser scan point cloud P_c and returns G_{2D}'s corresponding 3D representation G_{3D} as the result.

Intuitively, to lift a 2D model element (e.g., either a 2D polyline H_{2D} (e_k) indicating a road segment or a 2D point F_{2D} (v_i) indicating a road intersection), the 3D laser points locating nearby the 2D model element are more useful than the laser points that are further away. We take into account the laser points belonging to the ε-Neighborhoods of the 2D elements while lifting them.

Definition 4. Given a point cloud P_c, the ε-**Neighborhood** of a 2D model element g, denoted as εN(g), is a set of laser points in P_c that satisfy a given spatial predicate ε.

$$\varepsilon N(g) = \{p_i \in P_c \mid \varepsilon(p_i, g)\},$$

where $\varepsilon(p_i, g)$ denotes a spatial predicate defined on a laser point p_i and a 2D model element g.

3.2.3 Framework overview

Figure 4 depicts the overview of spatial network lifting, which consists of two major phases: filtering and lifting.

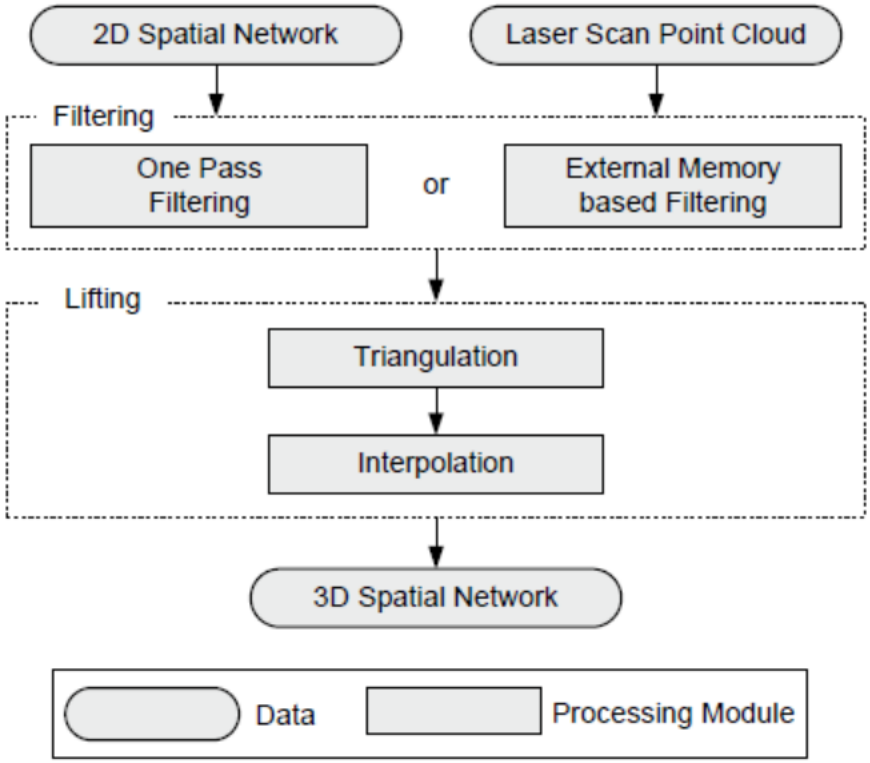


Figure 4: Lifting 3D Spatial Network: Framework Overview

The filtering phase takes as input a 2D spatial network and a massive laser scan point cloud, and prunes irrelevant laser points in the point cloud in order to obtain an appropriate ε-Neighborhood of every road segment and road intersection in the 2D spatial network. Two alternative filtering methods, one pass filtering and external memory based filtering, are provided in order to exploit situations where the 2D networks fits in main memory and to also provide a general solution that does not make this assumption.

The lifting phase consists of two steps, where interpolation follows triangulation. In the triangulation step, laser points in ε-Neighborhoods are transformed into TINs. After projecting the 2D spatial network onto the TINs, the interpolation step computes the corresponding elevation information, thus providing a 3D spatial network as the final output.

3.3 Spatial Network Lifting

Spatial network lifting focuses on generating an accurate and compact 3D spatial network in an efficient and scalable manner.

3.3.1 Filtering

Overview

The main task of the filtering phase is to obtain ε-Neighborhoods for the 2D model elements in G_{2D}. Instead of proposing yet another index to filter the laser points, we explore the opportunities of applying existing indexing techniques. A priori knowledge of the laser points being almost uniformly distributed with a guaranteed minimum resolution (e.g., at least one laser point per square meter), significantly influences our decision to choose a space-driven indexing technique, in particular, the grid index, instead of a data-driven indexing technique, e.g., a tree-based index.

As an aside, we chose not to use an approach where we first build spatial indexes on the point cloud P_c and on the 2D spatial network (e.g., using R-trees) and then join the two by synchronized traversing the two indices [15, 16, 17]. This is because P_c is massive in size (for Denmark, on the order of terabytes) and is collected rarely. Thus, the join operation is not carried out repeatedly, and the filtering is merely an intermediate step in solving the lifting problem.

By utilizing grid based indices, the filtering becomes parallelizable and can be processed easily on powerful machines with large main memories. However, we also consider the setting where the available main memory is limited, as this renders the paper's proposal applicable to commodity hardware.

We provide two filtering approaches that differ primarily in how they manage the smaller data set, i.e., the 2D spatial network G_{2D}.

1 **One pass filtering** assumes that there is enough internal memory to accommodate a grid index on G_{2D} and filters the point cloud P_c in a single pass by checking whether a laser point belongs to the ε-Neighborhoods of all road segments in G_{2D};

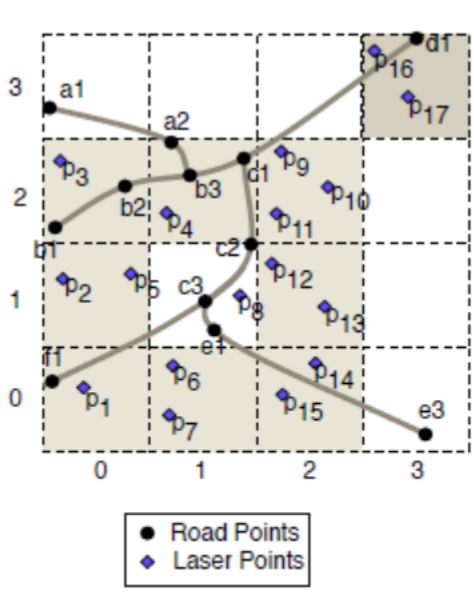
2 **External memory based filtering** works without any assumption on main memory size, so neither G_{2D} nor P_c are assumed to fit in internal memory. Two different traversal strategies, row-major and z-curve order, are used for loading disk blocks into memory for filtering.

One Pass Filtering

One pass filtering (or OPF for simplicity) utilizes a uniform grid to index both the 2D spatial network and the laser points. Cells in the grid are squares, and the width of a cell is governed by a user specified parameter δ that needs to be given before generating the grid. Figure 5 shows an example of how grid cells map to both road points (points contained in 2D polylines) and laser points. For ease of illustration, only a small portion of laser scan points mapped to cells are shown, while in reality, the cells have much more laser scan points due to the high density of the laser scan point cloud.

The spatial predicate used in OPF, denoted as $\varepsilon_{OPF}(g, p_i)$, returns true if a 2D model element g and the 2D projection of a laser point p_i, i.e., $prj^T(p_i)$, fall into the same grid cell. Given the same 2D model element g, the bigger the cell width δ is, the more laser points are contained in its ε-

Neighborhood N(g).



OPF

Cell	H_g	H_p
(3,3)	{d1,c1}	p_{16}, p_{17}

EMF

Cell	Road Blk	Laser Blk
(0,0)	{c3,f1}	p_1
(0,1)		p_6, p_7
(0,2)		p_{14}, p_{15}
(1,0)		p_2, p_5
(1,1)	{c2, c3}, {c3,f1}, {c3, e1}, {e1,e3}	p_8
(1,2)		p_{12}, p_{13}
(2,0)	{b1,b2},...	p_3
(2,1)	{a2,b3},...	p_4
(2,2)		p_9, p_{10},p_{11}

Figure 5: Grid Partitioning Index

OPF assumes that the grid index on the 2D spatial network, i.e., the mapping H_g given below, can fit fully into internal memory.

$$H_g : C \rightarrow \bigcup_{e \in E} H_{2D}(e).$$

For each cell c in grid cells set C, mapping Hg maintains a set, denoted as $H_g(c)$, containing all the 2D model elements that intersect with the cell.

Upon creation, H_g acts as a seed for generating the grid index on the point cloud P_c. OPF sequentially scans P_c only once to generate another mapping H_p from seed cells (those cells having 2D model elements in H_g) to laser points whose 2D projections are within the cells. H_p is formally defined as follows.

$$H_p : SC \rightarrow P_c; \text{ where } H_g(c) \neq \emptyset \text{ if } c \in SC \subseteq C.$$

Note that $H_p(c)$ records the ε-Neighborhoods of the 2D model elements in $H_g(c)$.

Isenburg and Lindstrom [18] observe that laser points are inherently topologically-coherent, which implies that the laser points are stored in an order that is an artifact of how they were collected by the planes flying over the covered land surfaces. In other words, the laser points in the point cloud

are not stored randomly. Rather, laser points that are geographically close are also stored close to each other and hence the point cloud is stored in a manner that to some extent is locality preserving. OPF exploits this property and thus avoids performing several passes to sort the point cloud P_c.

Based on the above observation, the mapping H_p needs not be maintained for every cell in memory at all times. Once $H_p(c)$ contains sufficient laser points, the 2D model elements in $H_g(c)$ are lifted. Recall that the laser point cloud we use guarantees one laser point per square meter, meaning that a cell with width δ is expected to contain δ • δ laser points. As OPF scans the point cloud, when $H_p(c)$ contains more than α • δ • δ laser points, $H_p(c)$ can be passed to the lifting phase immediately. Here, the fill factor α ∈ (0, 1]) is a parameter that represents a tradeoff between efficiency and accuracy: the higher the fill factor is, the more laser points must be contained in the ε - Neighborhoods, thus making the final 3D spatial network more accurate.

Algorithm 1 describes OPF. Function *GetIntersectedCells*(ls, δ) (in line 3) returns the cells that intersect with line segment is according to cell width δ, and function *GetContainedCells*(p_i , δ) (in line 6) returns the cell that contains the 2D projection of laser point p_i according to cell width δ.

```
   Input  : 2D spatial network G_2D, grid cell width δ,
            point cloud P_c, fill factor α.
   /* Initialize mapping H_g                          */
1  for each edge e ∈ G_2D.𝔼 do
2      for each line segment ls ∈ H_2D(e) do
3          for each cell c ∈ GetIntersectedCells(ls, δ) do
4              H_g(c) ← H_g(c) ∪ ls;
   /* One pass scan on the point cloud P_c            */
5  for each laser point p_i ∈ P_c do
6      Cell c ← GetContainedCell(p_i, δ)
       /* Use H_g as a seed                           */
7      if H_g(c) ≠ ∅ then
8          H_p(c) ← H_p(c) ∪ p_i;
9          if |H_p(c)| ⩾ α · δ · δ then
               /* Lift the 2D model elements in
                  cell c                               */
10             Lifting(H_g(c), H_p(c));
11             Release H_g(c) and H_p(c);
```

Algorithm 1: OnePassFilter

The advantages of one pass filtering are: (i) OPF does not incur any pre-processing cost of sorting or indexing the massive point cloud P_c; and it only scans the point cloud once; (ii) OPF is able to output parts of the resulting 3D spatial network with different accuracy requirements (by configuring α) as the laser points stream in. (iii) OPF can easily be parallelized to take advantage of either new hardware architectures like GPUs or cloud infrastructures like MapReduce.

External Memory Based Filtering

In contrast to OPF, external memory based filtering (abbreviated as EMF) works even if neither G_{2D} nor P_c is able to fit in internal memory. The basic goal of EMF is to efficiently filter large data sets of arbitrary sizes given a limited and fixed main memory budget.

In order to achieve an accurate TIN, EMF employs a spatial predicate ε_{EMF} (g, p_i) that returns true if the cell containing prj (p_i) is the cell that contains g or is one of the eight neighboring cells of the cell containing g (also called Moore neighbor cells of g). In the following discussion, we use 9-cell to indicate a cell and its Moore neighbor cells. For example, the 9-cell of cell (1, 1) is shown in the bottom left of Figure 5.

EMF scans both G_{2D} and P_c, organizing them into road blocks (lines 1–4 in Algorithm 2) and laser blocks (lines 5–7 in Algorithm 2), where each road (laser) block contains the road segments that intersect with (laser points that are in) a cell. The size of a laser block is typically decided by the cell width δ. Assuming each laser point takes 20 bytes (two doubles and one float), a laser block needs $20 \bullet \delta \bullet \delta$ bytes.

```
   Input : 2D spatial network G_2D, grid cell size δ,
           point cloud P_c, Curve Tag TAG;
           Memory Budget B.
   /* Scan and sort G_2D into road blocks   */
 1 for each edge e ∈ G_2D.𝔼 do
 2     for each line segment ls ∈ H_2D(e) do
 3         for each cell c ∈ GetIntersectedCells(ls, δ) do
 4             writeBlock(c, ls, FILE_G);
   /* Scan and sort P_c into laser blocks   */
 5 for each point p_i ∈ P_c do
 6     Cell c ← GetContainedCell(p_i, δ);
 7     writeBlock(c, p_i, FILE_P);
 8 Buffer A ← ∅;
 9 while Decide next cell c according to TAG curve do
10     A ← readBlock(c, FILE_G);
11     for each cell c' ∈ MooreNeighbour(c) ∪ c do
12         B ← LRU(B, readBlock(c', FILE_P));
           /* Lift the 2D model elements in
              cell c                          */
13         Lifting(A, B);
```

Algorithm 2: ExternalMemoryFilter

A road block typically takes up much less space than a laser block because it is uncommon to have very dense road segments (e.g., with a point for every one meter). We therefore assume that a road block has at most the same size as a laser block. After block reorganization, EMF reads road blocks into main memory according to a locality preserving space filling curve. After reading a new road block, EMF reads in its corresponding 9-cell laser blocks and overwrites laser blocks using the least recently used (LRU) [19] policy. A road block along with its 9-cell laser blocks are fed into the lifting phase, as described in lines 9–13 in Algorithm 2.

Given a limited memory budget, the order in which road blocks are read has a significant impact on the performance of EMF. In order to avoid frequent re-reading of the same laser blocks, locality preserving space filling curves are considered when EMF loads the road blocks. In particular, we consider two space-filling curves, namely the row-major curve and the z-order curve. Figure 6 shows the orders in which road blocks are loaded in memory starting from the bottom left cell, according to the two row-major and z-order curves, respectively.

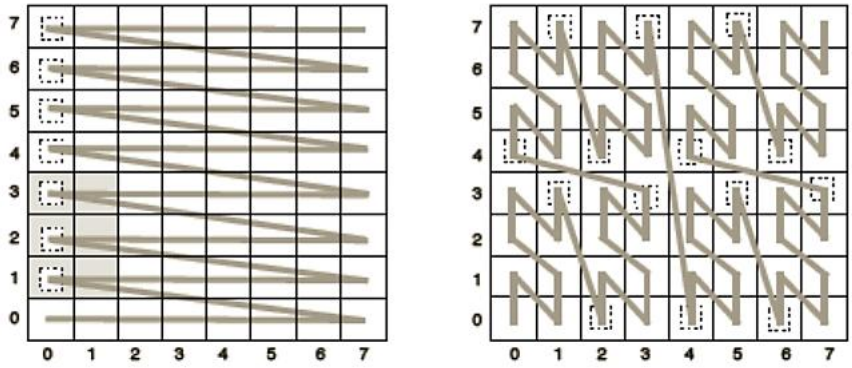


Figure 6: Locality Preserving Space Filling Curves

When a road block is being processed, its corresponding 9 laser blocks must be available in memory for processing. As EMF reads road blocks and moves along a curve, there are moments where memory is full and old laser blocks are overwritten with new ones. The dotted boxes in Figure 6 show the points where there is a high likelihood that new laser blocks must be read from disk.

Given a grid with n • n cells and a fixed-size memory (a multiple of n) that employs the LRU policy [19], we investigate the effects of using Z-order and row-major curves. Two grids with sizes 16•16 and 64•64 are considered, and we vary the memory size in multiples of n with 0.5 • n being the finest granularity. We then report the number of laser block replacements in Figure 7. The results show that for a memory budget ranging from 0.5•n to 2.5•n, the Z-order yields the fewest reads. The benefit of using the Z-order increases as the grid size n increases. Starting at 3 • n and onwards, the row-major ordering begins to outperform Z-order and performs stably. This is because the 9 laser blocks that must be considered for a road block belong to 3 grid rows.

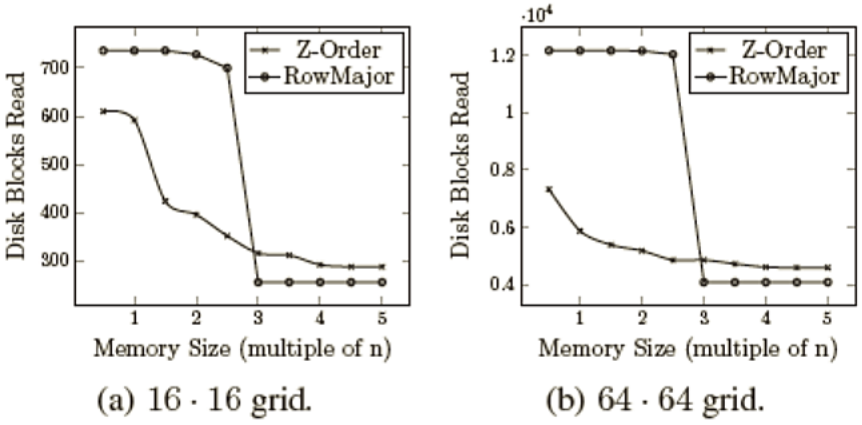


Figure 7: Performance of Z-order and Row-major Curves on Varied Memory Budget Sizes

To illustrate, consider again Figure 6. Assuming that the memory budget is 3 • n, when the first two rows of road blocks have been processed according to the row-major curve, the memory budget is fully occupied by the first three rows of laser blocks. In order to process the next road block, i.e., (2, 0) (surrounded with a dashed box), laser blocks (3, 0) and (3, 1) need to be read, and laser blocks (0, 0) and (0, 1) are over-written. The over-written blocks will never again be needed and hence with memory budget size 3 • n, no laser block needs to be read more than once. Therefore, with memory budget size no less than 3 • n, row-major behaves in a stable fashion, while with the Z-order some blocks that are over-written may be needed at a later time. Thus, if the memory budget is less than 3 • n, Z-order is preferable, while row-major wins and performs stably if the memory budget is no less than 3 • n.

3.3.2 Lifting

Upon successful filtering, the laser points in the ε-Neighborhoods are triangulated into a TIN. Then the elevation information in the TIN is assigned to the 2D road segments by projecting these onto the TINs and performing interpolation.

Triangulation

The elevation values in a given region are only available for the points where measurements were taken (e.g., the laser points in the region). To get the elevation for other points in the region, some form of approximation must be applied. A naive approach assigns an elevation to a point that is equal to the elevation of the point's nearest neighbor in the laser point cloud or that is equal to the average elevation of its k nearest laser points. However, these approaches are unable to produce accurate results.

We adopt a different tack: given a region, all the pertinent laser points, e.g., those in ε-Neighborhoods, are triangulated into a TIN to approximate the surface of the region. The elevation

of any point in the region can then be interpolated from the TIN.

Triangulation transforms a set of laser points, which represent discrete measurements on a surface, into a set of non-overlapping triangles where the vertices of the triangles are the laser points. We use Delaunay Triangulation [20] for triangulation. This is a specialized triangulation method where, in the resulting triangles, no triangle vertex is inside the circumscribed circles of any other triangle.

Recall that OPF utilizes a fill factor α to act as a parameter when determining how many laser points must be present when moving to the triangulation step. The higher the fill factor is, the more accurate the TIN approximates the real surface, thus yielding a more accurate 3D spatial network. For example, with a low fill factor of α = 0.1, some road segments (road points a_1, b_1, and c_1) are not covered by the resulting TIN (shown in Figure 8(a)). Increasing α to 0.4 improves the coverage (Figure 8(b)).

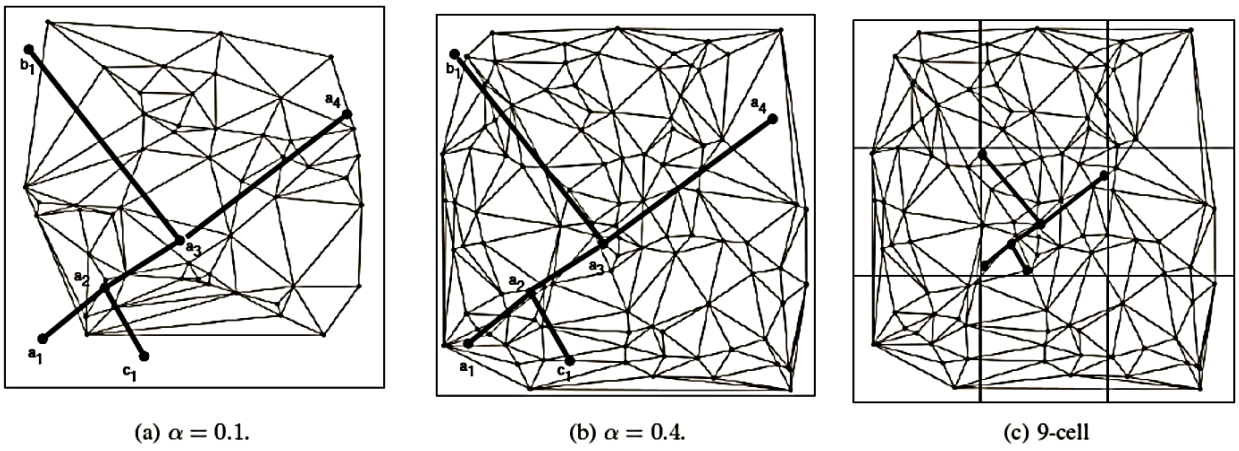


Figure 8: (a)(b) Delaunay Triangulation With Varying Fill Factor α in OPF.

(c) Delaunay Triangulation With 9-Cell in EMF.

EMF passes all the laser points in the relevant 9-cell to the triangulation step, as shown in Figure 8(c), which typically guarantees that all the road segments in the center cell are fully covered by the resulting TIN, thus achieving higher accuracy.

Interpolation

Figure 9 shows a 2D polyline representing a road segment and its TIN. Intuitively, projecting the 2D polyline to the TIN, the road segment's 3D representation becomes available. However, directly projecting a polyline to a TIN is computationally expensive. Instead, we sample a set of 2D road points on the 2D polyline and then project them onto the TIN in order to obtain their 3D counterparts. By connecting these 3D road points, the 3D polyline representation of the road segment is obtained, e.g., the 3D polyline (a_1, . . ., a_5, . . ., a_{11}) shown in Figure 9. Specifically, we

consider two sampling strategies: exact sampling (ES) and approximate sampling (AS).

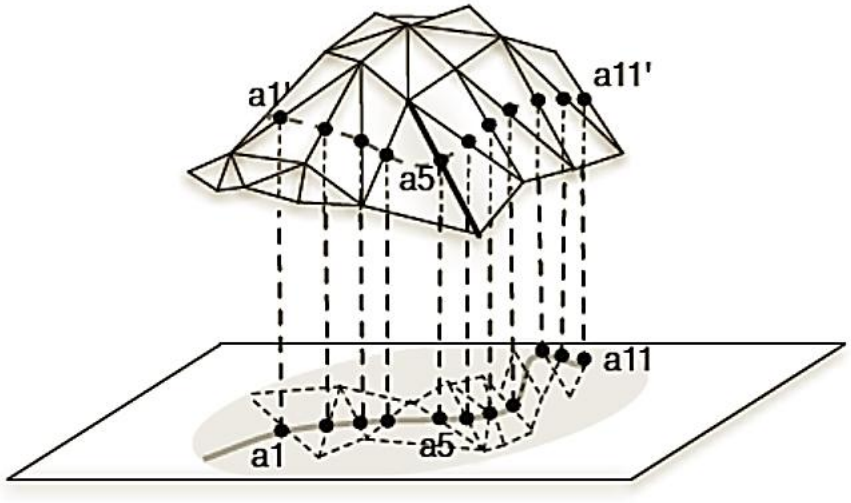


Figure 9: ES With Intersection Road Points

Exact sampling: ES selects the points where either the direction or the grade of the road segment change. ES starts by projecting the TIN to the 2D plane by ignoring the z coordinates of all vertices. To illustrate, this yields the dashed triangles on the 2D plane shown in Figure 9. Next, points on the road polyline are sampled by picking the points in the original 2D polyline representation (i.e., the points where the direction of the road changes) and all the intersections of the edges of the 2D triangles and the original 2D polyline (i.e., the points where the grade of the road segment may change). The grade of a road segment may change only when the road segment crosses from one triangle to another.

Since the intersections obtained by ES are always on TIN triangle edges, the elevations at the vertices of the triangle edges are used in linear interpolation to compute the intersections' elevations. Figure 9 shows an example of a triangle edge in bold; its two corresponding vertices are used for computing the elevation of road point a_5.

The remaining points in ES may fall in a triangle, and the elevations of these points are interpolated by taking into account the elevations of all three vertices of the triangle, i.e., using plane-based interpolation. Specifically, suppose Equation 1 represents the plane of a triangle.

$$A \cdot x + B \cdot y + C \cdot z + D = 0 \qquad (1)$$

Given the three vertices of a triangle: (x_1, y_1, z_1), (x_2, y_2, z_2), and (x_3, y_3, z_3), the parameters in

Equation 1 can be determined as follows.

$$A = \begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix} \quad B = \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix}$$

$$C = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \quad D = -\begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}$$

With the plane equation, the elevation (the z coordinate) of any 2D road point can be interpolated by plugging its x and y coordinates into the equation. Figure 10 shows an example of a triangle in bold lines where all the three vertices are used for computing the elevation of road point a_3'.

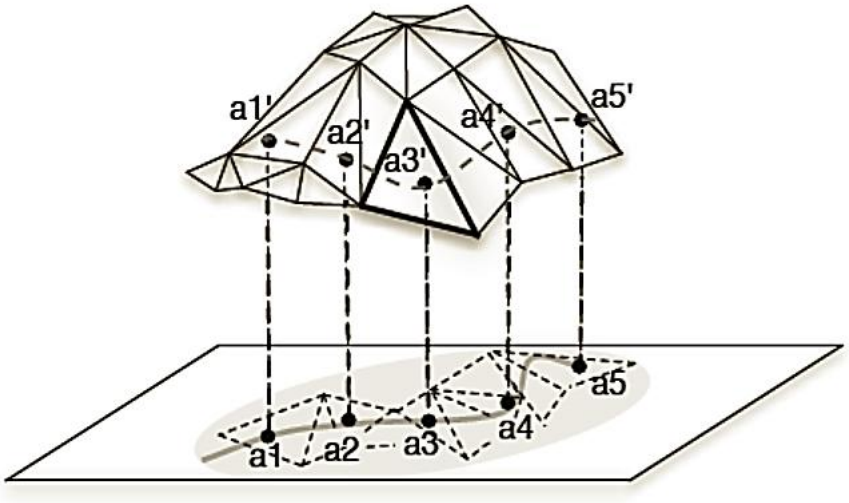


Figure 10: AS With Equidistant Road Points

Although the resulting 3D polylines using ES are very accurate and capture all the locations where the directions or the grades of road segments change, they may contain large numbers of road points, whose total size is typically orders of magnitude larger than those of the original 2D polyline representation.

Since we aim to provide a compact 3D spatial network with size comparable to that of the original 2D spatial network, we employ Douglas-Peucker (DP) simplification [21] to reduce the number of road points in the resulting 3D polylines. Given a user-specified distance threshold $\tau > 0$, DP simplification is able to simplify the original 3D polyline into a new 3D polyline with fewer road

points, while satisfying the property that the maximum distance between the original polyline and the new one does not exceed τ. By varying the distance threshold τ, we obtain different 3D spatial networks with different accuracies and sizes.

Approximate sampling (AS)**:** AS simply chooses equidistant points along road segments. Given an equidistance parameter *ed* and a 2D polyline, AS starts by choosing one end point of the polyline. It then chooses the point located distance *ed* from the previous point along the polyline, and it keeps choosing points in this manner until the polyline is exhausted.

After obtaining the equidistant points, plane based interpolation is applied to get their elevation values. By varying the equidistance parameter *ed*, the accuracy and the size of the resulting 3D polylines can be configured to cater to different application requirements. Figure 10 shows an example of AS with equidistant points along a road segment and interpolation of road point a_3'.

3.4 EXPERIMENTS

We detail the data sets, parameter settings, and implementation. Then we cover the empirical study of the efficiency, accuracy, and scalability properties of the proposed filtering and lifting framework.

3.4.1 Experimental setup

Data Sets: Two spatial networks in Denmark are lifted in the experiments.

Aalborg (AA) covers the Aalborg region (North Jutland, Denmark) which is of size approximately 7km • 5km. The laser point cloud of AA occupies 2.84 GB. The spatial network of Aalborg, obtained from OpenStreetMap, has a total length of approximately $4 \bullet 10^5$ m, and its 2D polyline representation is 1 MB.

North Jutland (NJ), the northern part of Jutland, Denmark, covers a region of 185km • 130km. The laser point cloud of NJ occupies 342 GB. NJ contains a spatial network with a total length of $1.17 \bullet 10^7$ m, whose 2D polyline representation is 28 MB.

We report on experiments were carried out on AA, unless stated explicitly otherwise.

Parameter Settings: The experiments are conducted by varying several parameters to study the effect of the trade-offs between accuracy, efficiency, and memory usage. Table 2 shows the parameters with default values shown in bold. The equidistant parameter *ed* used in approximate sampling is set to 10m. Experiments are conducted using default parameter values unless explicitly stated otherwise.

Grid Cell Width δ (m)	16, **64**, 128
Filling Factor α	0.4, **0.6**, 1.0
Memory Size	c, $2 \cdot c$, **$3 \cdot c$**, $4 \cdot c$

Table 2: Parameter Settings

Table 3 shows details on the grid indices for different cell widths. In the table, r • c indicate the size of the grid, where r and c denotes the number of rows and columns, respectively.

δ (m)	AA $(r \cdot c)$	NJ $(r \cdot c)$	BS (KB)
16	$316 \cdot 465$	$8082 \cdot 11526$	5
64	$79 \cdot 117$	$2021 \cdot 2882$	80
128	$40 \cdot 59$	$1011 \cdot 1441$	320

Table 3: Detail of Grid Indices

We compare variants of EMF that use row-major ordering and Z-curve ordering in terms of disk block reads by varying the available memory budget. The available memory is set as a multiple of c blocks (where c is the width of the grid). Note that as δ varies, the block size, which relates to how many laser points or 2D polylines fall in a cell in the grid, also varies accordingly. The size of a (laser or road) block is shown in the last column of Table 3.

Implementation: The filtering and lifting framework is implemented in C and C++, and Perl is used to perform auxiliary tasks. SCALGO Terrastream[1] is used to generate the TIN used in our ground-truth method. Triangle[2] is employed for Delaunay triangulation in the lifting phase. An R-tree library[3] is applied to facilitate the interpolation step in the lifting phase. All the experiments are carried out on an Ubuntu 11.04 LINUX machine with an Intel Xeon W3565 @3.2 GHz CPU (8MB cache, hyper-threading, 4 cores), 8 GB internal memory and 16.5 TB hard disk.

3.4.2 Ground truth generation

The ground truth is generated by triangulating all the laser points in the NJ point cloud into a huge TIN and then interpolate the road points obtained by exact sampling using this TIN. The statistics of the huge TIN are listed in Table 4, where ▽ indicates triangles in a TIN.

[1] http://madalgo.au.dk/Trac-TerraSTREAM
[2] http://www.cs.cmu.edu/ quake/triangle.html
[3] http://superliminal.com/sources/RTree.zip

Data	Laser	TIN	TIN-to-▽	Number of ▽
AA	2.84 GB	3.1 GB	6.3 GB	$3.7 \cdot 10^7$
NJ	342 GB	372 GB	742 GB	$9 \cdot 10^9$

Table 4: Ground Truth Storage

We use sum of squared errors (SSE), which is defined in Equation 2, for quantifying the differences between the 3D spatial networks generated by the filtering and lifting framework and the ground truth.

$$SSE = \sum_{i=1}^{|Rp|} (z_i - gt_i)^2 \qquad (2)$$

In the equation, |Rp| is the total number of road points based on a sampling; z_i is the elevation reported by the proposed method, and gt_i is the elevation obtained from the ground truth. The absolute error, $A_i = |z_i - gt_i|$, is also introduced as another accuracy measure in the subsequent discussion.

3.4.3 Sampling road points

In order to gain insight into the merits of the road points sampling, we consider the most hilly region in Aalborg to conduct a sensitivity experiment. The aim of this study was to compare the accuracy of exact sampling with DP simplification versus approximate sampling. We report the statistics for a 2 km long road with both positive and negative grades in Table 5.

Method	τ	# points	SSE
ES	-	2854	0
ES	0.1	638	3.89
with	0.2	333	16.32
DP	0.244	**282**	**21.23**
Simplification	0.3	232	30.90
AS	-	**286**	**23.73**

Table 5: EDP versus EXIP and DP selections

The study indicates that with approximately the same number of sampled road points, AS where *ed* = 10m is slightly worse than ES with DP simplification where $\tau = 0.244$. Note that there is an order of magnitude difference in the number of road points when compared to ES without DP simplification. Since we plan to provide a compact 3D spatial network and most regions in North

Jutland, Denmark are quite flat, the findings reported in subsequent experiments use approximate sampling.

3.4.4 Accuracy studies

We analyze the accuracy of the proposed approaches against the ground truth.

Accuracy Analysis of OPF: Figure 11(a) illustrates the effect of varying the fill factor (α), while fixing the grid size to its default size (64m). As we increase α, more laser points are passed into Delaunay triangulation, which yields more accurate elevation values.

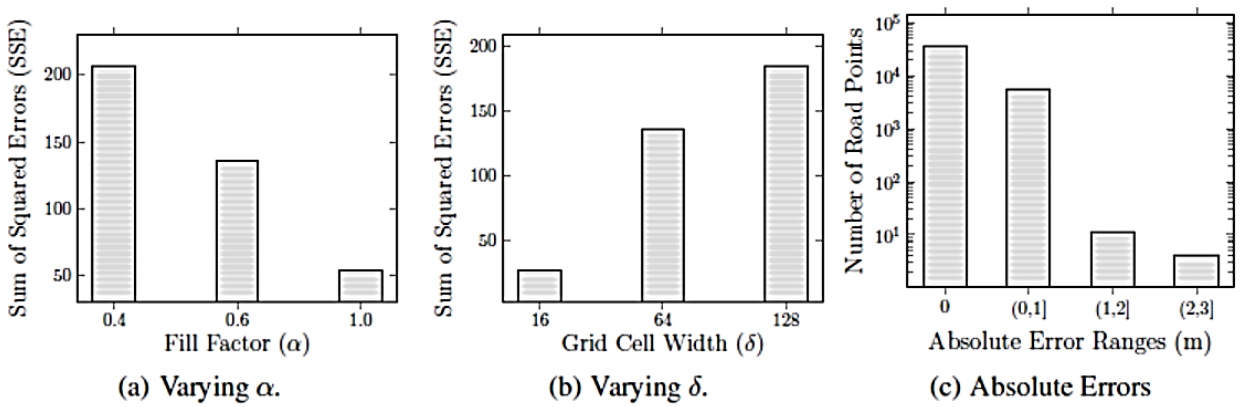


Figure 11: OPF Accuracy Study

Setting α to its default value and increasing the grid cell width, as shown in Figure 11(b), we notice that the SSE increases.

Although α acts as a threshold value for deciding how many laser points to be processed in triangulation, it is not able to control which laser points are chosen for triangulation. Since a grid cell with larger width should contain more laser points, the α fraction of laser points are more likely to be non-uniformly distributed in the cell, causing a deterioration in the triangulation and hence in the computed elevation values.

Accuracy Analysis of EMF: Unlike OPF, EMF employs triangulation on laser points in 9-cells, which results in a much better triangulation quality. As we increase the grid cell width, the SSE drops by orders of magnitude and the elevation values computed become much more accurate—see Figure 12(a). This occurs because the 9-cells contain increasingly more laser points.

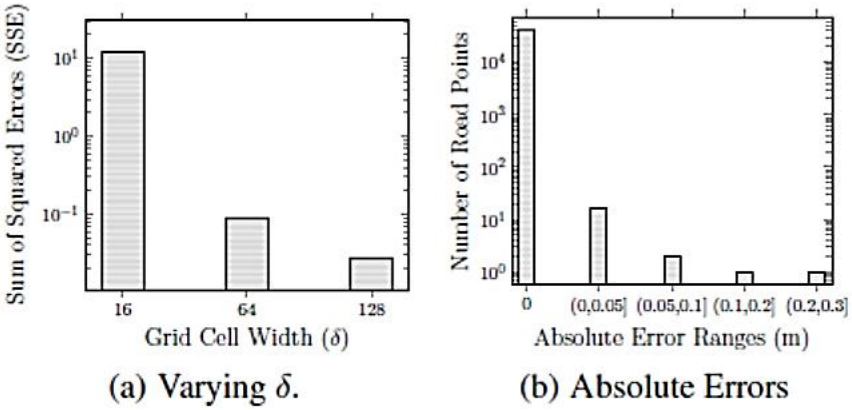


Figure 12: EMF Accuracy Study

Figure 11 (c) and Figure 12(b), for OPF and EMF, respectively, display the numbers of absolute error Ai within different error ranges, thus showing the distribution of the errors. The leftmost bar in each figure indicates the road points whose elevation is computed with no error. Note that the points in the subsequent buckets drop significantly in both methods, especially in EMF.

3.4.5 Storage studies

We quantify the storage requirements as the maximum amount of main memory required at any moment in the case of OPF; and as the number of laser and road block reads from disk in the case of EMF.

Memory Usage of OPF: Figure 13(a) shows that the maximum memory required grows as α increases. The larger α gets, the longer the wait is until there are enough laser points to proceed to the lifting phase. Hence, there is a greater demand on memory. Likewise, increasing the grid cell width yields a greater memory requirement to hold laser points as the area of the cell increases (shown in Figure 13(b)).

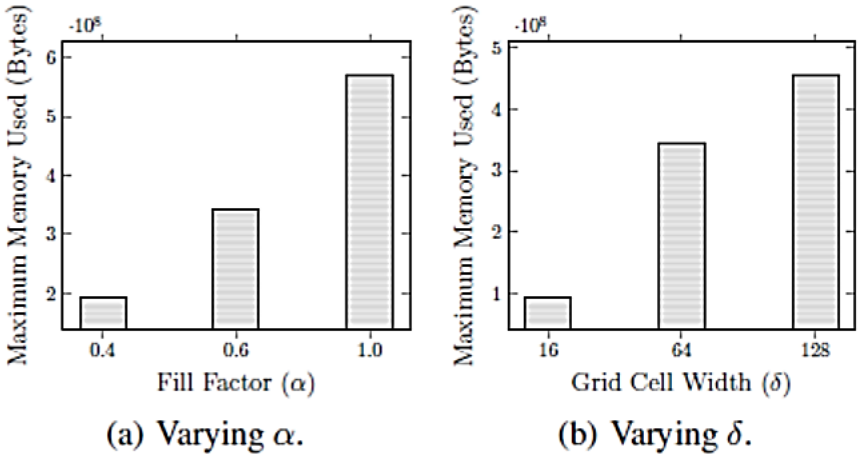


Figure 13: OPF Memory Measurements

Memory Usage of EMF: Figure 14 indicates that the disk block read performance of the Z-curve and row-major orders adhere to our earlier analytical results (cf. Figure 7). The biggest memory budget in EMF, with size of 4 • c and the largest grid cell width of 128m, does not exceed 80 MB of main memory. Additionally, given the maximum α and grid cell size, OPF has a memory upper bound of approximately 600 MB. Hence, both EMF and OPF can function with very limited memory for the AA spatial network.

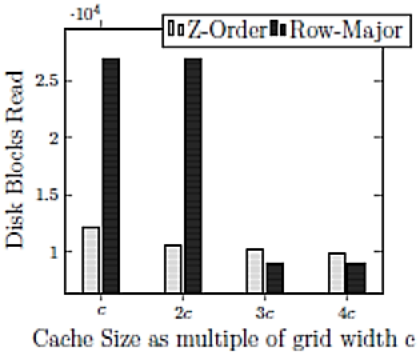


Figure 14: EMF Block Reads

3.4.6 Efficiency analysis

We analyze the performance of our proposed methods in terms of runtime. The goal is to gain insight into the practical feasibility of OPF and EMF.

For OPF, Figure 15(a) and (b) exhibits the effects of varying grid cell width and fill factor, where both end up having to pass increasing numbers of laser points to the lifting phase where triangulation and interpolation take longer.

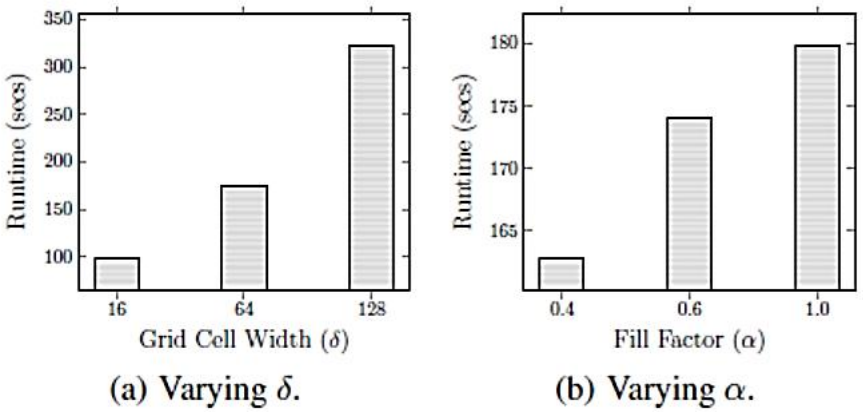


Figure 15: One-Pass Runtime Measurements

For EMF, Figure shows the effect on overall runtime when the available memory is varied. A very large proportion of time is spent on disk reads, which are much more time consuming than in-memory operations. We see a strong correlation between the disk reads and the overall runtime of EMF, which is also highlighted by the similarity to the I/O measurements shown in Figure 14.

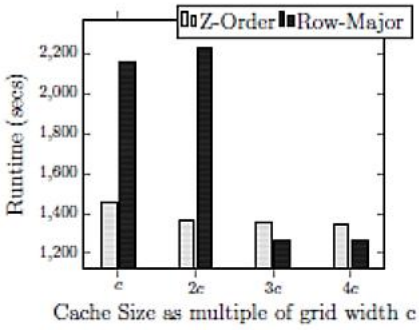


Figure 16: EMF Runtime

Although results show that OPF, which employs no pre-processing at all, is nearly an order of magnitude faster than EMF, this superior run-time performance comes at the cost of an accuracy

degradation. Note that both OPF and EMF take much less time than the ground truth computation.

3.4.7 Scalability studies

We conduct experiments on both AA and NJ to observe the scalability of the proposed methods. Recall that the point cloud of NJ is almost 120 times larger than the point cloud of AA, as shown in Table 4. We apply OPF on both data sets with default parameters. The corresponding runtime and maximum memory requirement suggest good scalability, as shown in Table 6. For EMF, we use Z-order and a memory budget with c blocks (i.e., around 230 MB); all the other parameters are set to default values.

Filtering	Data	Runtime	Max Mem
OPF (Default δ, α)	AA	2.9 m	327 MB
	NJ	282.4 m	2,457.6 MB
Filtering	**Data**	**Runtime**	**Blocks Read**
EMF (Z-Order, c)	AA	24.2 m	12,092
	NJ	6,192 m	3,092,518

Table 6: Scalability Analysis

The results in Table 6 also suggest that the runtime and number of block reads are proportional to the numbers of laser points in both data sets, which in turn suggests that EMF is scalability in the point cloud size.

3.5 Summary

We study a spatial network lifting problem that augments a standard 2D spatial network with elevation values extracted from a laser scan point cloud. We propose a novel filtering and lifting framework that aims to produce accurate 3D spatial network models that occupy limited storage in an efficient and scalable manner using commodity hardware. The results of extensive empirical studies offer insight into the design properties of the framework and suggest that the framework is practical and is indeed capable of meeting the design goals.

4. Evaluating Models of Vehicular Environmental Impact

4.1 Introduction

Reduction in the greenhouse gas (GHG) emissions is crucial for combating global warming that has increasingly adverse effects on life on Earth. The European Union (EU) aims to reduce GHG emissions by 30% by 2020 [31], and the group of Eight (G8) plans a 50% reduction by 2050 [25]. Australia aims at a more ambitious target: an 80% reduction by 2050 [24]. China targets a 17% reduction over 2010 levels by 2015 [23].

The transportation sector is the second largest in terms of GHG emissions, trailing only the energy sector. In the EU, emissions from transportation account for nearly a quarter of the total GHG emissions [27]. Further, road transport generates more than two-thirds of transport-related GHG emissions and accounts for about one-fifth of the EU total emissions of carbon dioxide (CO_2), the major greenhouse gas [28]. Therefore, reduction targets such as the above pose great challenges to the transportation sector in general and to vehicular transportation in particular.

In addition to improved design of vehicles and engines, eco-driving [35, 36] and eco-routing [51, 63] are simple yet effective approaches, which can achieve approximately 8–20% reduction in fuel consumption and GHG emissions from road transportation. Eco-driving targets eco-friendly driver behavior, e.g., accelerating appropriately, maintaining an even driving speed, and avoiding frequent starts and stops, etc. Eco-routing recommends routes that aim to minimize fuel consumption and GHG emissions.

An important first step in reducing environmental impact is to be able to measure the impact. Thus, scientists in research areas such as energy engineering, civil engineering, and environmental science have proposed a range of models that aim to measure fuel consumption and GHG emissions [40, 43, 60]. These models consider a wide range of factors, e.g., vehicle speed and acceleration, different physical features of vehicles, and geometric information on the spatial network in which the driving occurs.

While nearly a dozen impact models exist, no comprehensive comparison of these models exists. Moreover, the utility of the models for eco-driving and eco-routing is also not well understood. This report represents the first attempt at addressing these deficiencies by developing an evaluation framework, called EcoMark, for evaluating and comparing impact models using GPS trajectories and a 3D spatial network.

The use of GPS trajectories and a 3D spatial network bring the following benefits: (i) Many vehicles are equipped with GPS, and GPS data is plentiful and easy to collect. Thus, GPS data sets offer very good coverage of spatial networks, which enables Ecomark-based comparison of models to occur in a broader range of settings than in previous evaluations, where only limited selections of routes were considered [32, 59]. (ii) GPS trajectories are capable of providing the dynamic, travel-related

information required by the models, e.g., speeds and accelerations of vehicles, and of reflecting the traffic conditions in a spatial network. (iii) A 3D spatial network captures both the lengths and the grades (degree of incline or decline) of road segments, which are factors that affect fuel usage and GHG emissions. Although several models take grades into account [38, 47, 63], only Tavares et al. [63] report on empirical studies with a 3D spatial network to measure the influence of grades on fuel consumption and GHG emissions.

To the best of our knowledge, the proposal of EcoMark and its subsequent application of EcoMark represent the first comprehensive study of the environmental impact of road transportation using GPS trajectories and a 3D spatial network.

The work makes three contributions. First, a sophisticated evaluation framework is proposed that encompasses a 3D spatial network model. Second, a categorization and comparison of all eleven known models that can estimate fuel usage and GHG emissions is conducted. Third, comprehensive experimental studies are conducted on a half-year collection of GPS data from vehicles traveling in North Jutland, Denmark. Interesting findings obtained from the evaluation framework are discussed.

The remainder of this section is organized as follows. Section 1.2 gives a comprehensive review of the state-of-the-art techniques for estimating fuel consumption and GHG emissions. Section 1.3 presents the EcoMark framework and also covers the modeling of a 3D spatial network. Section 1.4 categories the 11 models into instantaneous models and aggregated models and discusses them in detail. Section 1.5 reports on the application of EcoMark to the impact models. Conclusions are drawn in Section 1.6.

4.2 EcoMark Design

Following an overview of EcoMark, we cover the modeling and construction of a 3D model of a transportation network and describe the trajectories that are used in EcoMark.

4.2.1 EcoMark overview

EcoMark is designed to evaluate state-of-the-art models of vehicular environmental impact in terms of fuel consumption and GHG emissions. It aims to provide an understanding of the utility of the impact models in relation to eco-driving and eco-routing and to offer insight into aspects such as which models can be used for identifying relationships between environmental impact and driver behavior, and which models are suitable for assigning weights to road segments that capture environmental impact, thus enabling eco-routing.

Figure 17 depicts an overview of EcoMark. Three types of raw data are used in EcoMark: a set of GPS observations, a 2D spatial network, and a laser scan point cloud [30]. A map matching module takes as input the set of GPS observations and the 2D spatial network. It outputs a set of map matched trajectories. A 3D spatial network generation module creates a 3D spatial network from the 2D spatial network and the laser scan point cloud. The trajectories and the 3D spatial network are fed into EcoMark as input data.

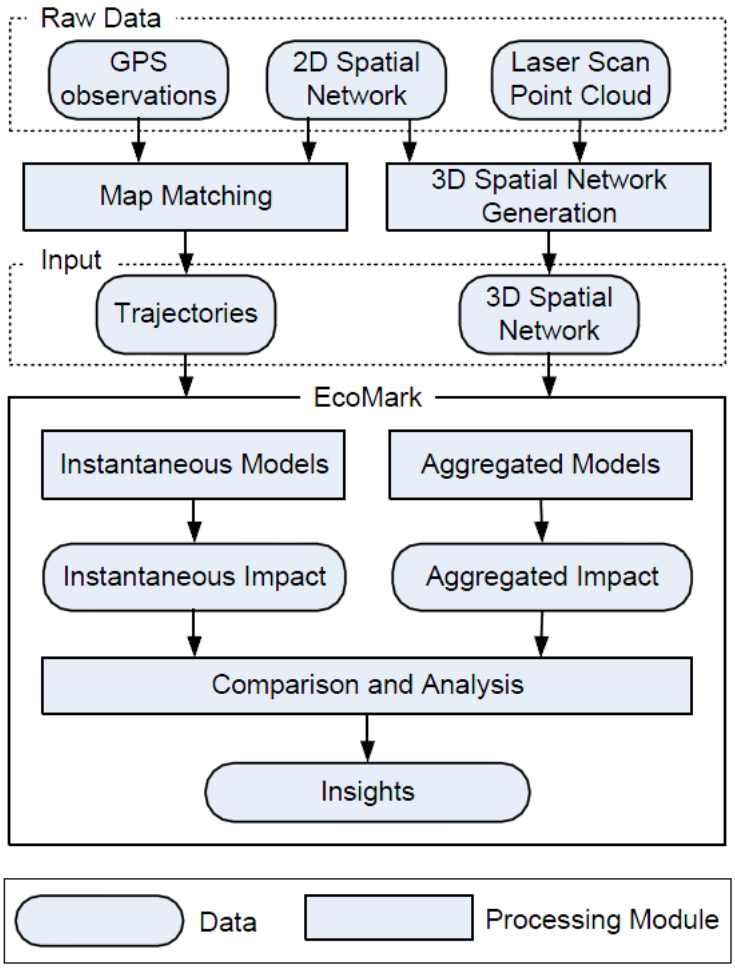


Figure 17: An Overview of EcoMark

Unless stated otherwise, we use 2D to denote the latitude-longitude plane ((x, y) plane) and 3D to denote the latitude-longitude-altitude space ((x, y, z) space).

Road grades are generally difficult to obtain because the major maps available, e.g., OpenStreetMap [1] and Bing Maps [22], are 2D and lack grades of road segments. Thus, although some models [38, 47] take grades into account, this parameter has generally been set to zero in practice due to the unavailability of segment grade information.

In EcoMark, we use a 3D spatial network that provides grade information for all road segments. This network is constructed by using a laser scan point cloud for lifting a 2D spatial network. The use of laser point data yields a model with much higher accuracy than what can be obtained when using Shuttle Radar Topography Mission (SRTM) data [29], which has been used in the past. This is

because the SRTM data contains only one altitude value for each 30m × 30m region, whereas the laser data we use contains one altitude value for each 1m × 1m region. The generation of the 3D spatial network is covered in Section 1.3.3.

In principle, the models supported in EcoMark take as input traffic and road information that can be obtained from GPS trajectories and a 3D spatial network, but do not require vehicle-specific factors. The supported models are categorized into instantaneous models and aggregated models. The instantaneous models take as input instantaneous (i.e., second-by-second) velocities and accelerations and output instantaneous fuel usage or GHG emissions. In contrast, the aggregated models take as input average velocities and output aggregated fuel usage or GHG emissions. The aggregated models can be applied at different aggregation levels, e.g., at the level of road segments or at the level of trajectories.

We use EcoMark to perform the following comparisons and analyses:

(1) comparison and analysis of instantaneous models;

(2) comparison and analysis of aggregated models;

(3) aggregation of the instantaneous results, and comparison of them with the results obtained from the aggregated models;

(4) comparison of the (instantaneous and aggregated) results with and without the use of road grades.

Finally, meaningful conclusions are drawn based on the sophisticated comparisons and analyses.

4.2.2 Modeling a 3D spatial network

A spatial network captures both topological and geometric aspects of a transportation network in a certain region. In EcoMark, a 3D spatial network is defined as a directed, weighted graph G = (V, E, F, H), where V and E is the vertex set and edge set, respectively; and F and H are functions that record the geometric information of vertices and edges, i.e., record the embedding of vertices and edges into geographical space.

A vertex $v_i \in V$ indicates a road intersection or an end of a road. E is the edge set, and an edge $ek \in E \subseteq V \times V$ is defined as a pair of vertices and represents a directed road segment connecting the two constituent vertices. For example, edge $e_k = (v_i, v_j)$ represents a road segment that enables travel from source vertex vi to target vertex v_j.

Function $F: V \rightarrow R \times R \times R$ takes as input a vertex and returns a 3D point. Function $H: E \times R \rightarrow R$ takes as input an edge and a value x, where x indicates the distance along the edge from the source vertex to a point on the edge. The function outputs the grade of the point on the edge. In EcoMark, positive grades indicate uphill directions and negative grades indicate downhill directions.

Assume that v_1 and v_2 are the vertices of a road segment in a 3D spatial network, and A is the grade

inflection point on the road segment. Figure 21 shows the altitude-longitude projection of the road segment. For edge (v_1, v_2), the grade of the sub-edge from v1 to A is θ, and the grade of the sub-edge from A to v_2 is 0. Thus, $H((v_1, v_2), x) = \theta$ if $0 \times |v_1, A|_{3D}$, where $|\bullet|_{3D}$ indicates the 3D Euclidean distance between two 3D points; $H((v_1, v_2), x) = 0$ if $|v_1, A|_{3D} \times |v_1, A|_{3D} + |A, v_2|_{3D}$. Similarly, for edge (v_2, v_1), $H((v_2\ v_1), x) = 0$ if $0 \times |v_2, A|_{3D}$; $H((v_2, v_1), x) = -\theta$ if $|v_2, A|_{3D} \times |v_2, A|_{3D} + |A, v_1|_{3D}$.

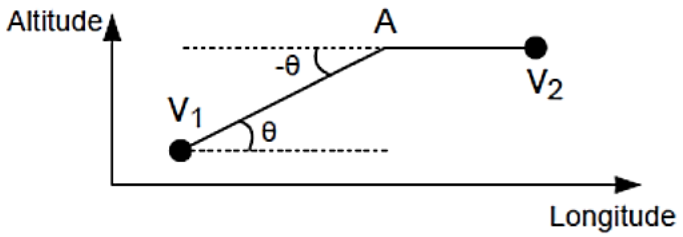


Figure 18: Altitude-longitude Projection of a Road Segment

4.2.3 Realizing a 3D Spatial Network

The 3D spatial network generation module augments a 2D spatial network with appropriate altitude information extracted from a laser scan point cloud, which contains a set of 3D points reflecting the surface of a certain region. An approximated surface of the region becomes available by transforming these 3D points into a TIN (Triangular Irregular Network) [57] surface. The altitude information of a 2D spatial network becomes available by projecting the 2D spatial network to the TIN surface, thus generating its corresponding 3D spatial network.

Specifically, we consider the region of North Jutland, Denmark in EcoMark. We obtain a 2D spatial network of North Jutland from OpenStreetMap, and we employ a laser scan point cloud that covers North Jutland to generate a 3D spatial network of North Jutland. The detail of realizing the 3D spatial network is covered in the last chapter: Lifting a Spatial Network Using Laser Scan Data.

4.2.4 Trajectories

Since vehicle tracking using GPS is widespread and growing, GPS based vehicle tracking data is increasingly available and is thus an attractive source of vehicle movement data [67]. Thus, it is used in EcoMark.

A trajectory, denoted as $T = (p_1, p_2, \ldots, p_X)$, is a sequence of GPS observations, where a GPS observation pi specifies the location (typically 2D) and velocity of a vehicle at a particular time point. It is clear that instantaneous velocities are available from GPS trajectories, and instantaneous accelerations can be derived based on consecutive GPS observations. Given a spatial network G, a map matching algorithm [56] is able to associate each observation in a trajectory with a specific

location on an edge in G.E.

4.3 Model Analysis

EcoMark covers eleven state-of-the-art models that estimate vehicular environmental impact. These models are categorized into instantaneous models and aggregated models as defined in Section 1.3.1, and are described in Sections 1.4.1 and 1.4.2, respectively. All the models share the property that their input can be derived from GPS trajectories, e.g., velocities and accelerations, or can be obtained from a 3D spatial network, e.g., road grades.

4.3.1 Instantaneous models

EMIT

EMIT (EMIssions from Traffic) [40] was proposed by Cappiello et al. at 2002, who were motivated by the idea of deriving a fuel consumption estimation model that combines advantages of both pure regression-based and load-based models. In general, the pure regression-based models simply develop functions of instantaneous vehicle speed and acceleration to estimate fuel consumption, and thus can be used widely and are relatively easy to run. However, they lack clear physical interpretations. Whereas, the load-based models estimate through a series of detailed physical parameters of vehicles, and therefore are usually complicated and require high computational effort. To overcome the disadvantages of two types of models, Cappiello et al. propose Emit to estimate fuel consumption, aiming to require simple parameters, run fast but still consider physical features of vehicles.

EMIT is a linear regression model that employs functions of instantaneous vehicle speed and acceleration to estimate fuel consumption and emissions (CO_2, CO, HC, NO_x) for light-duty vehicles. The design of EMIT aims to offer a relatively simple estimation model that requires less computational time than the load-based models. Meanwhile, it combines physical factors from the load-based models in order to increase the accuracy of fuel consumption estimation. Although EMIT provides separate models for estimating fuel consumption and emission rates, only its model for estimating fuel consumption is presented in this report.

Input: second-by-second speed v (m/s) and acceleration a (m/s^2).

Assumptions:

(1) the road grade is 0;

(2) engine power required for accessories, e.g., air condition, is 0;

(3) no history effect, e.g., cold-start, is combined in the model; and

(4) only hot-stabilized conditions is considered.

Model for estimating fuel consumption:

$$FR = \begin{cases} \alpha_{FR} + \beta_{FR} v + \gamma_{FR} v^2 + \delta_{FR} v^3 + \zeta_{FR} av & \text{if } P_{tract} > 0 \\ \alpha'_{FR} & \text{if } P_{tract} \leqslant 0 \end{cases}$$

where,

- F R denotes the fuel consumption rate (g/s).
- α_{FR} = 0.365, β_{FR} = 0.00114, $\gamma_{FR} \approx 0$ (this value is small, and therefore is dropped in the model), δ_{FR} = 9.65e-07, ζ_{FR} = 0.0943 and α_{FR} =0.299 are parameters for the fuel consumption estimation model.
- P_{tract} is total tractive power requirement at the wheels (kW). In particular,

$$P_{tract} = A \cdot v + B \cdot v^2 + C \cdot v^3 + M \cdot a \cdot v + M \cdot g \cdot sin\vartheta \cdot v$$

  where,

  - A = 0.1326: rolling resistance term (kW/m/s),
  - B = 2.7384e−03: speed-correction to rolling resistance term (kW/(m/s)2),
  - C = 1.0843e−03: air drag resistance term (kW/(m/s)3),
  - M = 132 : vehicle mass (kg),
  - g = 9.81: gravitational constant (m/s2), and
  - ϑ = 0: road grade (degrees).

Evaluation:

The data used to calibrate EMIT is the National Cooperative Highway Research Program vehicle emissions database, developed by the University of California at Riverside [73]. The emission database provides dynamometer measurements of second-by-second vehicle speed and emission rates. The results of calibration is evaluated by comparing EMIT estimates with statistics provided in the emission database. The evaluation results show that the estimated fuel consumption matches the measurements well with 0.0% error and R2 ~ 0.971. Cappiello et al. further compare EMIT with CMEM [70, 73, 61], a state-of-the-art load-based model, using data in the emission database that seem different from the data used to develop EMIT. EMIT and CMEM estimate fuel consumption with 5.3% error and R2 ~ 0.95, and -2.2% error and R2 ~ 0.81, respectively.

Cappiello et al. anticipate, but have not implemented, that EMIT can be possibly integrated with a microscopic traffic simulator to evaluate the influence of different traffic management strategies on air quality. EMIT may also be used to estimate various route planning strategies.

Discussion:

EMIT shows advantages in using a relatively simple function to estimate fuel consumption whilst considers the impact of physical features of vehicles on fuel consumption. As EMIT only requires information about second-by-second vehicle speed and acceleration during computation, it could run fairly fast. Evaluations carried by Cappiello et al. also indicate that EMIT estimates reasonable results over a wide range of operating conditions.

A major limitation of EMIT is that the model does not take into account road grade that may significantly influence the fuel consumption estimation. However, it is possible to assign road grade ϑ with values greater than 0 while calculating P_{tract}, but Cappiello et al. assume $\vartheta = 0$. Cappiello et al. emphasize that it is not trivial to combine this parameter into Emit, and thus would only be addressed in future work.

VT-Micro

VT-Micro, developed by Rakha et al. [33, 60], is a microscopic model that uses second-by-second vehicle speed and acceleration to estimate fuel consumption and emission rates. It is designed to overcome the limitations of macroscopic models that fuel consumption and emission rates are computed only based on vehicle average speed without considering transient changes in vehicle speed and acceleration, and thus are not suitable for estimating fuel consumption at operational levels, e.g., in Intelligent Transportation Systems (ITS).

The basic form of VT-Micro is a third degree polynomial and non-liner model. It takes as input instantaneous vehicle speed and acceleration to estimate fuel consumption and emission rate for light-duty vehicles. Although the fuel consumption is non-linear w.r.t. the vehicle speed and acceleration, it is generally true that as acceleration and speed increase, the fuel consumption also tends to increase.

Input: second-by-second speed s (km/h) and acceleration a (km/h/s).

Assumptions:

(1) the model is developed for light-duty vehicles and trucks;

(2) the model only estimates fuel consumption and vehicle emission for hot-stabilized conditions, and does not consider the effect of vehicle start;

(3) the speed and acceleration levels employed by the model lie in the bounds of the Oak Ridge National Laboratory (ORNL) data.

Model for estimating instantaneous HC, CO, CO2 emission rate MOE: (g/s):

$$MOE = \begin{cases} \exp(\sum_{i=0}^{3}\sum_{j=0}^{3}(K_{i,j} \times s^i \times a^j)) & \text{if } a \geqslant 0 \\ \exp(\sum_{i=0}^{3}\sum_{j=0}^{3}(L_{i,j} \times s^i \times a^j)) & \text{if } a < 0 \end{cases}$$

where,

- $K_{i,j}$: model regression coefficient at speed s power i and acceleration a power j for positive accelerations (i.e., a 0), and
- $L_{i,j}$: Model regression coefficient at speed s power i and acceleration a power j for negative accelerations (i.e., a < 0).

Evaluation:

The evaluation of VT-Micro is carried out by: I) comparing its aggregate emission with Environmental Protection Agency (EPA)'s aggregate emission measures, and II) comparing its instantaneous emission with EPA's second-by-second field HC, CO and NOx . The results show that the estimated emission rates fit well with EPA's emission values in both I and II cases. VT-Micro is also compared with MOBILE5a, an old version of a macroscopic model MOBILE6 [43] (i.e., a state-of-the-art macroscopic model), using FTP City cycle3 (a.k.a. LA4) and the Highway Economy cycle4. The results of model comparison indicate that the estimated emission rates show consistency with MOBILE5a results.

In a subsequent comparison [59], VT-Micro developers compare VT-Micro with MOBILE5a and its latest version MOBILE6 [7] and CMEM [61] (i.e., a´state-of-the-art load-based microscopic model) using ORNL and EPA data, where the ground truth emission rates are known. Although vehicle operative parameters required are different in these models, VT-Micro developers claim that the comparison is carried out in equal conditions. VT-Micro developers conclude from the comparison that the estimated emission rates of VT-Micro and MOBILE6 follow the difference of the ground truth emission rates in various driving cycles; MOBILE5a shows poor estimation results when compared to the EPA data; whereas CMEM exhibits some abnormal behaviors during estimation.

Later, VT-Micro developers assess the environmental impacts of different route choices using VT-Micro, CMEM and MOBILE6 [32]. The evaluation employs GPS data gathered during the morning commute near Washington, D.C. on weekdays between March and May of 2006. The GPS data are recorded at 1s intervals, and provide information about date, time, vehicle longitude, vehicle latitude, vehicle speed, vehicle heading, and the number of tracking satellites. There are totally 39 trips recorded, including 21 highway routes and 18 arterial routes. Using GPS data, estimated emission rates of VT-Micro and CMEM follow the same trend and indicate that choosing arterial routes generate less emission than highway routes, whereas MOBILE6 results show the trend of emission rates in the opposite way. VT-Micro developers claim that VT-Micro and CMEM are more reliable to estimate emission rates than MOBILE6, because MOBILE6 is incapable of using second-

by-second speed and acceleration to capture the instantaneous emission.

VT-Micro is also incorporated into the INTEGRATION model framework [14], which combines microscopic traffic simulation models and microscopic emission estimation models to evaluate alternative Intelligent Transportation Systems (ITS) initiatives.

Discussion: VT-Micro is a well-developed model for estimating fuel consumption and emission rates using carefully calibrated coefficients. The authors conducted intensive evaluations to demonstrate its accuracy and usefulness in real world scenarios, such as to evaluate the designs of different ITS. However, Rakha et al. overlooked the fact that physical features of vehicles and road grade are also decisive factors for estimating fuel consumption rates.

Vehicle Specific Power

Instead of designing a model for directly estimating emission rates or fuel consumption, Jose Luis Jimenez-Palacios proposes a model for computing vehicle Specific Power (SP) [9]. SP is defined as the instantaneous power per unit mass of the vehicle, and is closely related with emission rates. In particular, the model for computing SP exhibits two main benefits: I) parameters required to compute SP can be directly measured from the roadside, e.g., vehicle speed and acceleration and road grade; and II) SP captures most of the dependence of emissions on engine operating conditions. Jimenez-Palacios argues that SP is directly related to the vehicle engine load and thus to emission rates, and therefore is more reliable than solely based on vehicle speed and acceleration in estimating emission rates.

SP is the instantaneous power per unit mass of the vehicle. The design of SP considers not only parameters that are measurable from the roadside, but also vehicle engine features, e.g., the rolling resistance, aerodynamic drag and the kinetic and potential energies of the vehicle, which are combined as coefficients in the model.

Input: second-by-second vehicle speed: v (m/s), acceleration: a (m/s^2), vertical rise or slope length: grade, headwind into the vehicle: v_w (m/s).

Assumptions:

(1) SP is developed as an independent variable for studying hot-stabilized emission of passenger cars and light-duty trucks; and

(2) SP does not capture the following effects that also influence the amount of emission: engine speed, engine friction losses, transmission losses, power consumption of accessories (e.g., air conditioning), load weight.

Model for SP (kW/Metric Ton or W/kg or m^2/s^3):

$$SP = v \bullet (1.1 \bullet a + 9.81 \bullet grade(\%) + 0.132) + 3.02 \bullet 10{-}4 \bullet (v + v_w)^2 \bullet v$$

Analysis:

Analysis of the relationships between instantaneous emission rates and SP is conducted using a part of EPA emission data. The emission data include one second averages of speed, fuel economy, engine-out and tailpipe CO, HC and NO_x emissions, etc.. The analysis indicates that emission rates are closely related to SP, but are not always proportional to SP, because high emission occurs at low or high values of SP. For example, the engine-out NO_x emission increases as SP value grows from 0 to 20, but drops fast since then. However, the analysis was conducted on the relationship between fuel consumption and SP using test data obtained from a 1994 Jeep Cherokee on two driving cycles. The results indicate that the fuel consumption is proportional to SP values.

Discussion:

Although the model for computing SP does not directly estimate fuel consumption and emission rates, SP is closely related to them, and thus can be used as their relative estimates. In particular, the conducted analysis indicates that instantaneous fuel consumption is proportional to instantaneous SP. Furthermore, SP is also a fairly comprehensive estimate for relative fuel consumption, because it combines various vehicle operating conditions and includes road grade, in addition to the use of vehicle speed and acceleration. SP values are straightforward to compute, since all parameters it employs can be directly measured from the road side. As SP considers road grade in the model, it is potentially more accurate than models, e.g., VT-Micro and EMIT, that do not combine this parameter.

MEF

Inspired by the popularity of microscopic emission and fuel consumption estimation models, Lei et al. propose models for computing instantaneous HC, CO, NOx and CO2 emission and a model, called MEF, for estimating fuel consumption for light-duty vehicles [52].

MEF is based on two microscopic estimation models, i.e., VT-Micro [60] and POLY [64]. POLY employs not only current speed and acceleration but also history acceleration to estimate emission and fuel consumption. To combine the two models together, Lei et al. utilize the framework of VT-Micro and change where acceleration of current time t is used in VT-Micro into a variable that aggregates acceleration at previous nine seconds of t.

Input: speed v(t) and acceleration a(t) at current time t, acceleration a(t − i) (i=1,...,9) at previous nine seconds of current time t.

Assumptions: the models are developed based on two types of light-duty vehicles (LDV), LDVI and LDVII, which mainly differ in the accumulated mileage, i.e., the mileage of LDVI <= 50,000 miles, and the mileage of LDVII >= 50,000 miles.

MEF model for estimating fuel consumption:

$$e(t) = \begin{cases} \exp(\sum_{m=0}^{3}\sum_{n=0}^{3}(\lambda_{m,n} \times v(t)^m \times \bar{a}(t)^n)) & \text{if } \bar{a} \geqslant 0 \\ \exp(\sum_{m=0}^{3}\sum_{n=0}^{3}(\gamma_{m,n} \times v(t)^m \times \bar{a}(t)^n)) & \text{if } \bar{a} < 0 \end{cases}$$

where,

- e(t) : instantaneous fuel consumption or emission rate at time t (g/s),
- $\lambda_{m,n}$ and $\gamma_{m,n}$: model coefficients for e(t) at speed v(t) power m and acceleration a(t) power n; values of $\lambda_{m,n}$ and $\gamma_{m,n}$ for LDVI and LDVII are provided, respectively; and
- $\bar{a}(t)$: the composite acceleration at time t derived from the current acceleration a(t) and nine history accelerations a(t−i) (i=1,...,9). In particular,

$$\bar{a}(t) = \alpha \cdot a(t) + (1-\alpha) \cdot \sum_{i=1}^{9} a(t-1)/9$$

where α is calibrated as 0.5.

Evaluation:

Data utilized to develop and evaluate models proposed by Lei et al. are collected in Beijing, China, from 18th to 29th November 2007 using a Portable Emission Measurement System (PEMS) instrument. The data include CO, CO_2, HC and NO_X emission and GPS data (speed, latitude, longitude, altitude, etc.), etc. The data used for the evaluation purpose are different from those employed to develop the models. Two types of evaluation are carried out, namely the instantaneous evaluation and the aggregate evaluation. Specifically, MEF is compared with VT-Micro and POLY, and coefficients used by the three models are calibrated separately using the collected PEMS data. In the instantaneous evaluation, the second-by-second ground truth fuel consumption is compared to the fuel consumption estimated by MEF, VT-Micro and POLY, respectively. The results indicate that the three models well follow the trend of real world fuel consumption trajectory. In the aggregate evaluation, the ground truth (or estimated) fuel consumption rate is generated by averaging the real world (or estimated) fuel consumption rate over the distance of the test routes. MEF and VT-Micro have similar prediction performance, and POLY tends to overestimate the emission and fuel consumption rate.

Discussion:

MEF combines features of two models, namely VT-Micro and POLY, and replaces where acceleration of current time is used in VT-Micro with the aggregated history acceleration. However, the evaluation conduced by Lei et al. cannot show that history acceleration is an influential factor of fuel consumption, and that the combination of VT-Micro and POLY estimates fuel consumption more accurately than VT-Micro alone, because MEF always exhibits similar prediction behivour to VT-Micro, which does not consider the history acceleration during the estimation. In fact, POLY always overestimates the fuel consumption in the conducted experiments, perhaps, due to the

combination of history acceleration in its model. Moreover, MEF does not include road grade in computation, whereas the grade happens to be a significant parameter for estimating fuel consumption.

Joumard

Joumard et al. carried out a study to identify the most important vehicle parameters that influence the fuel consumption and emission rates [49]. In particular, the model is derived as a function of instantaneous speed and acceleration, and thus can be used to assess the design of traffic management systems on their impact on the traffic pollution.

The model proposed by Joumard et al. estimates the emission of passenger cars simply using instantaneous speed and acceleration.

Input: second-by-second vehicle speed v (m/s) and acceleration a (m/s^2).

Assumptions:

(1) the model is only developed for urban conditions;

(2) the model only considers hot engine emission, when the temperature is 70–80° C.

Model for estimating emission:

$$v + v \times a$$

Evaluation:

Empirical experimental study was carried out to evaluate the model. In total, more than 800 routes were explored using two test vehicles, which were equipped with measurement of emission and operational variables of vehicles. After that, the real world emission data gathered were compared to the emission rate estimated by the model proposed by Joumard et al. on the same route. However, the agreement between the gathered emission and the estimated emission is poor. Joumard et al. concluded that the reason could be problems related with the test data, such as errors in the gathered emission data or non-representability of the two test vehicles. Joumard et al. expect that the emission model should be able to predict reasonable estimation of CO, HC and NO_x based on the instantaneous speed and acceleration.

Discussion:

Developed at an early stage (i.e., 1995) of microscopic models, the model proposed by Joumard et al. inevitably encounters limitations, such as not considering vehicle engine features or road grade. Simple as the model is, it is potentially able to compute faster than other microscopic models, e.g., VT-Micro [33] and EMIT [40]. However, as Joumard et al. have not demonstrated the effectiveness of the model in their conducted validation process, the accuracy of the model is not clear.

SIDRA-Inst

SIDRA, by Bowyer et al. [38], is a framework that includes four fuel consumption estimation models. SIDRA has been developed into commercial products[4]. The four models are designed in an increasing order of aggregation and integrate vehicle energy features (e.g., vehicle mass and drag force). The four interrelated models follow the same modeling framework, where a more aggregated model is derived from a more detailed model. Specifically, the SIDRA framework includes an instantaneous model, a four-mode element (i.e., acceleration, cruise, deceleration and idle) model, a running speed model, and an average travel speed model, denoted as SIDRA-Inst, SIDRA-4Mode, SIDRA-Running, and SIDRA-Avg, respectively.

SIDRA-Inst, the least aggregated model, takes as input instantaneous speed v (m/s), percentage of grade G whose positive (or negative) value indicates a uphill (or downhill) grade, and instantaneous acceleration a (m/s^2); and it computes the instantaneous fuel consumption f_t at per unit time (mL/s) as follows.

$$f_t = \begin{cases} 0.444 + 0.09R_T v + [0.054a^2 v]_{a>0} \\ 0.444 \end{cases}$$

where,

RT is the total tractive force required to drive the vehicle:

$$R_T = 0.333 + 0.00108v^2 + 1.2a + 0.1177G$$

4.3.2 Aggregated models

Song

In order to evaluate the effects of different operational strategies in a traffic network, Song et al. develop a model to capture the effects of dynamic traffic on fuel consumption [62]. The model is designed to overcome limitations of existing fuel consumption models, which either take as input vehicle power related parameters (i.e., microscopic load-based models) or compute fuel consumption using average speed (i.e., macroscopic models). The former type of models requires parameters that can neither be obtained from field traffic nor be supported by traffic simulation models, and thus are hardly used for estimating fuel consumption in the dynamic traffic network. The latter type of models is usually criticized for its inadequate accuracy in computing fuel consumption in the dynamic traffic network. Hence, Song et al. propose the model to estimate the fuel consumption rates at link and network level for the use in the dynamic traffic network.

The model proposed by Song et al. computes a relative indicator of fuel consumption level for light duty vehicles (LDV), and is built on a variable of vehicle specific power (SP) presented in Section 3. However, the grade used in the original SP model is omitted by Song et al. Similar to the analysis conducted by Jimenez-Palacios [47], experiments carried out by Song et al. also indicate that fuel

[4] http://www.sidrasolutions.com/

consumption rates increase almost proportionally to SP when SP values are positive, and tend to be low or roughly constant when SP values are negative. In particular, the negative SP values represent decelerations.

Input: second-by-second vehicle speed v (m/s) and acceleration a (m/s^2).

Assumptions:

(1) the road grade is 0;

(2) the model is developed for light duty vehicles.

Model for computing fuel consumption indicator:

$$I = \frac{\beta \cdot \sum_t \theta + t}{\gamma \cdot \frac{\sum_t v}{\varepsilon}}$$

where,

- I: fuel consumption indicator. In particular, I is always greater than 1, and a lower value of I indicates higher traffic fuel efficiency.
- β = 0.264: coefficient,
- t: total travel time (s),
- γ = 2.13: normalized fuel consumption rate at the most efficient speed,
- ε = 18.3: most fuel efficient speed (m/s), and
- θ: a surrogate variable of vehicle specific power. Specifically,

$$\theta = \begin{cases} VSP & VSP > 0 \\ 0 & VSP \leqslant 0 \end{cases},$$

where,

$$VSP = v \cdot (1.1 \cdot a + 0.132) + 0.000302 \cdot v^3$$

Evaluation:

A database that contains vehicle speed and fuel consumption around a freeway toll station is utilized to evaluate the model proposed by Song et al.. The experiments are conducted using a Volkswagen Jetta at electronic toll collection (ETC) lanes and manual toll collection (MTC) lanes. The estimated fuel consumption indicator I for ETC and MTC lanes are 1.60 and 3.50, respectively, which means that the fuel efficiency is improved by 54.3% by using ETC lanes. The absolute fuel

consumption recorded in the database for ETC and MTC is 37.1 g/km and 87.6 g/km, respectively, and thus the improvement shown is approximate 57.7%. Song et al. conclude that the indicator I is a valid relative estimation for fuel consumption.

The fuel consumption indicator I is also applied to estimate fuel efficiency of different drive cycles, each of which is a series of vehicle speed that simulates various driving conditions. Song et al. observe that high I values are associated with either lower speed but severer acceleration/deceleration or higher idling proportion. In addition, indicator I is also used to observe the fuel consumption variations of traffic from 6:00 to 20:00 on a single road in Beijing. The experiment shows that the most fuel inefficient traffic happens between the morning peak hour 7:30-8:30 and between evening peak hours 16:30-19:30.

Discussion:

Indicator I developed by Song et al. estimates relative fuel consumption based on the SP variable, and shows advantages in its high estimation accuracy and the fact that it is relatively easy to use. The high accuracy shown by I could be attributed to the well-design of I developed by Song et al. On the other hand, it is also possible that indicator I estimates accurate fuel consumption because of the solid design of the variable SP that indicator I is based on. However, Song et al. did not conduct experiments to compare indicator I with the variable SP, and thus the improvement of I to SP is unknown. Furthermore, Song et al. removed road grade from the variable SP in the design of indicator I, thus potentially influencing the accuracy of indicator I when estimating fuel consumption on steep roads.

Tavares

With the aim of optimizing the investment in terms of environment, technology and economics during the collection and transportation of municipal solid waste, Tavares et al. proposed a route optimization method for minimizing vehicle fuel consumption using 3D geographical information systems [19]. Tavares et al. employ a routing cost function for estimating the fuel consumption of a particular road segment rather than the road distance. The cost function takes into account the average vehicle speed on a road, the percentage of vehicle load and the effects of road grade, which can be obtained from the 3D terrain model. An optimal route that minimizes fuel consumption is suggested by the method to achieve the overall aim of the paper.

The model for estimating fuel consumption developed by Tavares et al. is derived from COPERT, a computer program that computes vehicle emissions [54]. In the study of the estimation model, Tavares et al. use data of a particular category of vehicles, i.e., the diesel heavy duty vehicles, available in COPERT.

Input: the average driving speed V (km/h), the percentage of vehicle load LP, and the percentage of road slope x.

Assumptions: the model is designed for estimating fuel consumption of diesel heavy duty vehicles.

Model for estimating fuel consumption of a road segment fc (g/km):

$$fc = FCS \times LCF \times GrCF$$

where,

- FCS : the basic fuel consumption as a function of speed, FCS = 1068.4V −0.4905 ;
- LCF : the additional influence of vehicle weight, LCF = 1 + 0.36 (LP −50)/100 ;
- GrCF: the additional influence of road grade, GrCF = $0.41e^{0.18x}$.

Tavares et al. claim that although the variable LCF can be varied during the waste collection, it remains constant for each collection route.

Applications of the model:

The accuracy of the fuel consumption estimation model is not evaluated by Tavares et al. However, the use of the model is intensively demonstrated in the paper. In particular,

1. 3D road network modeling: The 3D network is modeled as an array of arcs, each of which represents a geographical polyline object and is connected with other arcs. Each arc is further modeled as an array of vertices and segments. Specifically, vertices represent geographical points where roads change directions or gradient; segments are straight lines connecting two adjacent vertices. Fuel consumption FC_k at arc k:

$$FC_k = \sum_{i=1}^{n} (Lseg_i \times fc_i)$$

   Where,

   - $Lseg_i$: the 3D length of the i^{th} segment of arc k (km);
   - fc_i: the fuel consumption of the i^{th} segment of arc k (g/km).

2. Optimization of vehicle routing: An optimal route for collecting municipal solid waste is defined as a route that has the lowest fuel consumption among all possible routes and covers every waste collection point. Thus, the overall cost function for route planning is defined as $\sum_{k=1}^{m} FC_k$, where m is the number of arcs included in a route. Tavares et al. employ ArcGIS, ArcInfo and Network Analyst extension software provided by ESRI[5] to identify a fuel consumption optimal route.

3. Experimental study: Tavares et al. compare two types of routes for collecting and transporting municipal solid waste, i.e., a fuel consumption optimal route and a shortest 3D

[5] http://www.esri.com/software/arcgis/index.html

distance route. Tavares et al. conclude that the shortest 3D distance route requires more fuel than the fuel consumption optimal route, because using the shortest 3D distance as the cost function does not consider road grade. In general, the downhill routes minimize vehicle elevation gain, and thus require less fuel consumption than the steep ascent routes. Hence, a route that avoids steep ascents but travels slightly longer may still cost less fuel consumption than the shortest 3D distance route.

Discussion: Tavares et al. proposed a fuel consumption model based on a 3D network model, and use the model as a cost function for identifying a 3D route to minimize the overall fuel consumption along the waste collection points. The model shows advantages of I) taking into account road grade and also demonstrating its key influential factor to fuel consumption, and II) aggregating fuel consumption of a single segment into fuel consumption of a 3D arc rather than simply computing the instantaneous fuel consumption, which is easier for readers to utilize the model as a routing cost function.

However, Tavares et al. only demonstrated the use of the fuel consumption model, but has not evaluated its accuracy, and thus its overall effectiveness in predicting fuel consumption is unknown yet. Furthermore, some important features that may influence the fuel consumption are not considered in the model. Specifically, vehicle acceleration is not included in the model at all, even though many other fuel consumption models have demonstrated its importance to estimating the fuel consumption; average vehicle speed cannot reflect speed variations of vehicle driving cycles, and thus cannot accurately capture the actual variations of fuel consumption on a road segment; Tavares et al. have not considered scenarios where a truck stops at a collection point, gathers waste and starts over again, where fuel would be significantly consumed. Although Tavares et al. combine the variable of vehicle load LCF in the model, it is unreasonable to assume the load remains constant for each collection route. This is because as waste is collected along the collection points, the load would gradually increase, and thus the truck would consume more fuel than the starting point. It would be useful to measure the vehicle load along the collection points and combine the data into the fuel consumption computation.

SIDRA-4Mode

SIDRA-4Mode, by Bowyer et al. [38], is a four-mode elemental model that estimates fuel consumption of individual vehicles in a common application scenario, where a vehicle drives on a short segment, stops and starts over again. Thus, the driving cycle can be modeled as a cruise-deceleration-idle-acceleration-cruise (CDIAC) cycle. In particular, the overall fuel consumption F_s (mL) can be calculated as:

$$F_s = F_{c1} + F_d + F_i + F_a + F_{c2}$$

where F_{c1} and F_{c2} (mL) are the fuel consumption when a vehicle cruises at speed v_1 with distance x_1 and at v_2 with distance x_2, respectively; F_d, F_i and F_a (mL) are the fuel consumption when the vehicle

status is deceleration, idle and acceleration, respectively. F_c, F_d, F_i and F_a are calculated as follows.

Cruising mode: The cruising mode is defined as traveling from the end of an acceleration to the start of the next deceleration. The cruising fuel consumption Fc (mL) is calculated as follows.

Input: the average cruise speed v_c (km/h) allowing fluctuations, the cruise distance x_c (km), and percentage of grade G. Model for estimating total cruise fuel consumption F_c (mL):

$$F_c = f_c x_c$$

where the cruise fuel consumption per unit distance f_c (mL/km) is:

$$f_c = 1600/v_c + 30 + 0.0075v_c^2 + 108k_{E1}E_{k+} + 171.2E_{k+}^2 + 10.6k_G G$$

where,

- k_{E1}: the calibration parameter:

$$k_{E1} = \min \begin{cases} 12.5/v_c + 0.0000133v_c^2 \\ 0.63 \end{cases}$$

- E_{k+}: the marginal fuel consumption due to speed fluctuations:

$$E_{k+} = \max \begin{cases} 0.258 - 0.0018v_c \\ 0.1 \end{cases}$$

- k_G: the calibration parameter:

$$k_G = \begin{cases} 1 - 2.1E_{k+} & G < 0 \\ 1 - 0.3E_{k+} & G > 0 \end{cases}$$

Deceleration mode: The deceleration mode estimates the fuel consumption during a deceleration from an initial speed v_i to a final speed v_f, with $v_f < v_i$. The deceleration fuel consumption F_d (mL) is calculated as follows.

Input: the initial and final speed v_i and v_f (km/h), the deceleration time t_d (s) and distance x_d (km), and the percentage of grade G. In particular, if t_d and x_d are unknown, they can be estimated as:

$$t_d = (v_i - v_f)/(1.71 + 0.238\sqrt{v_i - v_f} - 0.009v_f)$$

$$x_d = [(0.473 + 0.00155v_i - 0.00137v_f) \times (v_i + v_f) \times t_d]/3600$$

Model for estimating total deceleration fuel consumption F_d (mL):

$$F_d = \max \begin{cases} 0.444t_d + [30k_x + 0.0075k_y k_1(v_i^2 + v_f^2) + 108k_a E_k + 10.6k_x G]x_d \\ 0.444t_d \end{cases}$$

where,

- k_x: the energy-related parameter:

$$k_x = \begin{cases} k'_x & \text{if } 0 \leqslant k'_x \leqslant 1.0 \\ 0 & \text{if } k'_x < 0 \\ 1.0 & \text{if } k'_x > 1.0 \end{cases}$$, where

$$k'_x = 0.129 + 0.00421v_i + 0.0026v_f + 0.0544G$$

- k_y: the energy-related

$$k_y = k_x^{0.75}$$

- k_1: the integration coefficient

$$k_1 = 0.621 + 0.000777v_i - 0.0189\sqrt{v_f}$$

- k_a: the energy-related parameter

$$k_a = k_x^{3.81}(2 - k_x^{3.81})$$

- E_k: the change (decrease) in kinetic energy per unit mass per unit distance during deceleration from v_i to v_f (J/kg.m)

$$E_k = 0.3858 \times 10^{-4}(v_f^2 - v_i^2)/x_d$$

Acceleration mode: The acceleration mode is used to estimate fuel consumed during an acceleration from an initial speed v_i to a final speed v_f, with vi < vf. The acceleration fuel consumption Fa (mL) is calculated as follows.

Input: the initial and final speed v_i and v_f (km/h), the acceleration time ta(s) and distance x_a (km), and the percentage of grade G. When ta and x_a are unknown, they can be estimated using the following functions:

$$t_a = (v_f - v_i)/(2.08 + 0.127\sqrt{v_f - v_i} - 0.0182v_i)$$

$$x_a = [(0.467 + 0.002v_f - 0.0021v_i) \times (v_i + v_f) \times t_a]/3600$$

Model for estimating total fuel consumption F_a (mL) during acceleration:

$$F_a = \max \begin{cases} 0.444t_a + [30 + 0.0075k_1(v_i^2 + v_f^2) + 108E_k + 54k_2E_k^2 + 10.6G]x_a \\ 0.444t_a \end{cases}$$

where,

k_1: the integration coefficient

$$k_1 = 0.616 + 0.000544v_f - 0.0171\sqrt{v_i}$$

k_2: the integration coefficient

$$k_2 = 1.376 + 0.00205v_f - 0.00538v_i$$

E_k: the change (increase) in kinetic energy per unit mass per unit distance during acceleration from vi to v_f (J/kg.m)

$$E_k = 0.3858 \times 10^{-4}(v_f^2 - v_i^2)/x_a$$

Idle mode: The idle mode estimates fuel consumption when a vehicle stops. In practice, a vehicle may be considered as being stopped if the traveling speed is below 5km/h. The idle fuel consumption F_i (mL) is calculated as follows.

Input: the stopped time t_i.

Model for estimating total fuel consumption F_i (mL) during acceleration:

$$F_i = 0.444t_i$$

SIDRA-Running

The running speed model, by Bowyer et al. [38], is a more aggregated model derived from SIDRA-4Mode. The overall fuel consumption of a traveling vehicle includes the fuel consumed during the period of running and when the vehicle is shortly stopped. The running status of the vehicle consists of the cruise, acceleration and deceleration modes defined in the four-mode elemental model. The major difference is that SIDRA-Running comprises speed when the vehicle cruises, accelerates and decelerates in the model, whereas the speed is differentiated in SIDRA-4Mode. Coefficient in the SIDRA-Running is re-calibrated. In particular, the fuel consumption of the running mode F_s can be estimated as follows.

Input:

- x_s: the total travel distance (km);
- t_s: the total travel time (s);
- G: the percentage of road grade;

- v_i and v_f: the initial and final speed during each positive acceleration (km/h), with $v_i < v_f$;
- t_i: the idle (stopped) time (s);

In cases where v_i, v_f and t_i are unknown in advance, they can be estimated using functions, or replaced by other parameters when used in some functions, as shown below.

- v_r: the average running speed (km/h),

$$v_r = 3600 x_s / (t_s - t_i)$$

If t_i is unknown in advance, v_r can be estimated as:

$$v_r = \min \begin{cases} 8.1 + 1.14 v_s - 0.00274 v_s^2 \\ v_s \end{cases}$$, where

$$v_s = 3600 x_s / t_s$$

- ti: thus, can be estimated from vr as:

$$t_i = t_s - 3600 v_s / v_r$$

- E_{K+}: sum of positive kinetic energy changes per unit mass per unit distance (j/kg.m = m/s^2),

$$E_{K+} = 0.3858 \times 10^{-4} [\sum (v_f^2 - v_i^2)] / x_s$$

If v_i and v_f are unknown,

$$E_{K+} = \max \begin{cases} 0.35 - 0.0025 v_r \\ 0.15 \end{cases}$$

Model description:

$$F_s = F_i + f_r x_s$$, where

- F_i: the fuel consumption during idle periods

$$F_i = 0.444 t_i$$

- f_r: the average fuel consumption per unit distance excluding idle periods (mL/km)

$$f_r = 1600 / v_r + 30 + 0.0075 v_r^2 + 108 k_{E1} E_{K+} + 54 k_{E2} E_{K+}^2 + 10.6 k_G G$$, where

  - k_{E1}: the calibration parameter

$$k_{E1} = \max \begin{cases} 0.675 - 1.22/v_r \\ 0.5 \end{cases}$$

- k_{E2}: the calibration parameter

$$k_{E2} = 2.78 + 0.0178 v_r$$

- k_G: the calibration parameter

$$k_G = \begin{cases} 1 - 1.33 E_{K+} & \text{if } G < 0 \\ 0.9 & \text{if } G > 0 \end{cases}$$

SIDRA-Avg

This is the simplest, also the most aggregated, model for estimating fuel consumption, where only the average travel speed is employed [38]. Thus, the model is more suitable for estimating fuel consumption in a traffic network rather than on individual road segments.

Input: the total travel distance x_s (km), and the total travel time t_s (s) including stopped time.

Assumption: As Bowyer et al. stated in the report [38] that "*the average travel speed model does not adequately reflect the increase in aerodynamic drag, and therefore fuel consumption, at high speed*." Thus, the model can only be used in urban road networks, and the average travel speed is below 50km/h. Where the average speed is over 50km/h, the running speed model should be used.

Model description:

The fuel consumption per unit distance f_x (mL/km) can be calculated as:

$$f_x = 1600/v_s + 73.8$$

where $v_s = 3600x_s/t_s$ is the average travel speed (km/h).

If the fuel is consumed by a larger vehicle, f_x is estimated as:

$$f_x = 2400/v_s + 99.1$$

4.3.3 Summary

In the summary of models in Table 7, √ (×) indicates that a model considers (does not consider) a feature. The inputs required by the models are obtainable from GPS trajectories (e.g., velocities and accelerations) or from a 3D spatial network model (e.g., grades), and therefore all models can be applied straightforwardly in most road transportation networks. Table 7 shows that instantaneous models require both velocities and accelerations as input, while most aggregated models take only velocities as input. The road grade is an optional input parameter in the instantaneous as well as the aggregated models. Studies exist that indicate the benefits of considering road grades in the models [63].

Some models, specifically EMIT, SP, Song, and the SIDRA family, explicitly combine physical vehicle features (indicated by the Physical Features column in the table) in the fuel consumption computation, whereas the remaining models, namely VT-Micro, MEF, Joumard, and Tavares, do not consider such features. No correlation exists between the type of model, i.e., instantaneous or aggregated model, and the use of physical features.

	Models	Input			Physical Features	Absolute Value	Vehicle Type	Model Year	Data Source
		Velocity	Acceleration	Grade					
Instantaneous	EMIT [20]	✓	✓	×	✓	✓	Light	2002	U.S.
	VT-Micro [13,40]	✓	✓	×	×	✓	Light	2004	U.S.
	SP [27]	✓	✓	✓	✓	×	Unknown	1999	U.S.
	MEF [32]	✓	✓	×	×	✓	Light	2010	China
	Joumard [29]	✓	✓	×	×	×	Passenger	1995	EU
	SIDRA-Inst [18]	✓	✓	✓	✓	✓	Light, Heavy	1985	Australia
Aggregated	Song [42]	✓	✓	×	✓	×	Light	2009	China
	Tavares [43]	✓	×	✓	×	✓	Heavy	2009	Cape Verde
	SIDRA-4Mode [18]	✓	×	✓	✓	✓	Light, Heavy	1985	Australia
	SIDRA-Running [18]	✓	×	✓	✓	✓	Light, Heavy	1985	Australia
	SIDRA-Avg [18]	✓	×	×	✓	✓	Light, Heavy	1985	Australia

Table 7: A Summary of Models for Estimating Vehicular Environmental Impact

Most models predict an absolute impact value with units (indicated by the Absolute Value column. The units of the absolute impact values reported by instantaneous models are typically volume unit per time unit (e.g., mL/s) or mass unit per time unit (e.g., mg/s). For the aggregated models, volume (or mass) unit per distance unit, e.g., L/km or g/km, are reported. In contrast to the majority of models, SP and Song provide merely a relative indicator of fuel consumption. Although Joumard is able to predict an "absolute" value, its unit is unclear in the description of the model [49]. Therefore, we regard the value as a relative indicator.

All the models have prerequisites on vehicle types, as listed in the Vehicle Type column. The models are also developed during different years and in different regions (as listed in the Model Year and Data Source columns). Therefore, the gathered traffic information and the measured fuel consumption that were used to calibrate the models may also be different. This may to some extent explain the use of different parameters in the models.

4.4 Empirical Studies

We conduct comprehensive empirical studies to compare and analyze the 11 models, in order to obtain insight into the properties of the different categories of models and how to apply the models

to eco-driving and eco-routing.

4.4.1 Setup

The two main input data sources of EcoMark are GPS trajectories and a 3D spatial network, as described in Section 1.3.1. We use GPS data collected from 150 vehicles traveling in North Jutland, Denmark, during January to June 2007. The sampling frequency is 1 Hz, which makes application to the instantaneous models easy.

We apply an existing map matching tool [56] along with a 2D spatial network of North Jutland, Denmark obtained from OpenStreetMap to the GPS data, from which we get a set of 52, 084 trajectories, denoted as $\mathbb{T}$.

The statistics of the trajectories are provided in Table 3 that lists the number of trajectories belonging to different categories. In the table, roadseg indicates the number of road segments traversed by a trajectory; avgvel indicates the average velocity of a trajectory; and duration indicates the total travel time of a trajectory. The units of the average velocity and the duration are km/h and minute(s), respectively. The corresponding 3D spatial network of North Jutland, Denmark is obtained as described in Section 1.3.3.

# total	52,084	# *roadseg* $\leqslant$100	47,333
# *avgvel* $\leqslant$ 50	37,018	# *roadseg* >100	4,751
# 50 < *avgvel* $\leqslant$ 100	14,701	# *duration* < 30	44,544
# *avgvel* > 100	365	# *duration* $\geqslant$ 30	7,461

Table 8: Statistics of Trajectories in $\mathbb{T}$

4.4.2 Evaluating instantaneous models

Experiments are conducted to evaluate whether the instantaneous models show consistent estimation of environmental impact given the same trajectories. To quantify the consistency between two models m and n, vector based cosine similarity is applied, as defined in Equation 1.

$$consistency(m,n) = \frac{1}{|\mathbb{T}|} \cdot \sum_{\mathcal{T} \in \mathbb{T}} \cos(\mathcal{L}_{\mathcal{T}}^{(m)}, \mathcal{L}_{\mathcal{T}}^{(n)}) \tag{1}$$

where L_T is a vector that contains the impact values of trajectory T computed by model m and function cos(•, •) computes the cosine similarity between two vectors.

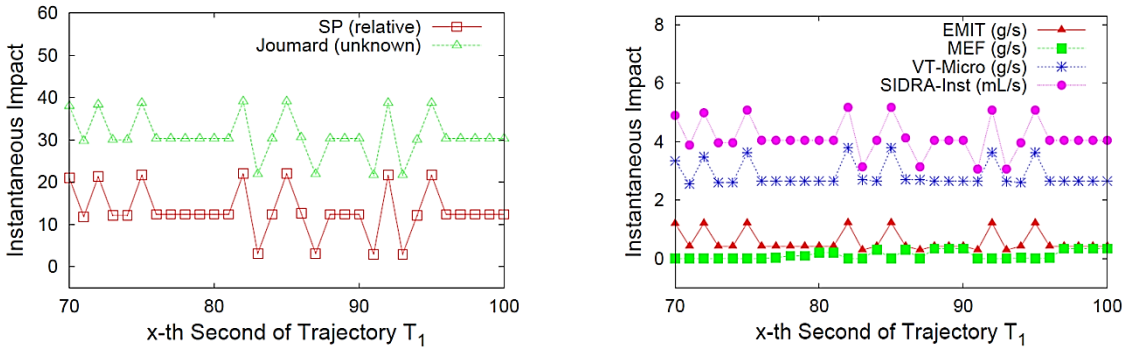
The similarity between each two instantaneous models is shown in Table 9. As can be observed, EMIT, SP, Joumard, and SIDRA- Inst show comparatively similar behavior with similarities above

83%. In contrast, VT-Micro and MEF exhibit dissimilarity with the other models.

	VT-Micro	SP	MEF	Joumard	SIDRA-Inst
EMIT	69.6%	84.5%	19.7%	89.2%	93.5%
VT-Micro	1	50.2%	26.0%	71.5%	62.7%
SP	-	1	0.58%	86.4%	85.1%
MEF	-	-	1	12.2%	13.7%
Joumard	-	-	-	1	83.3%

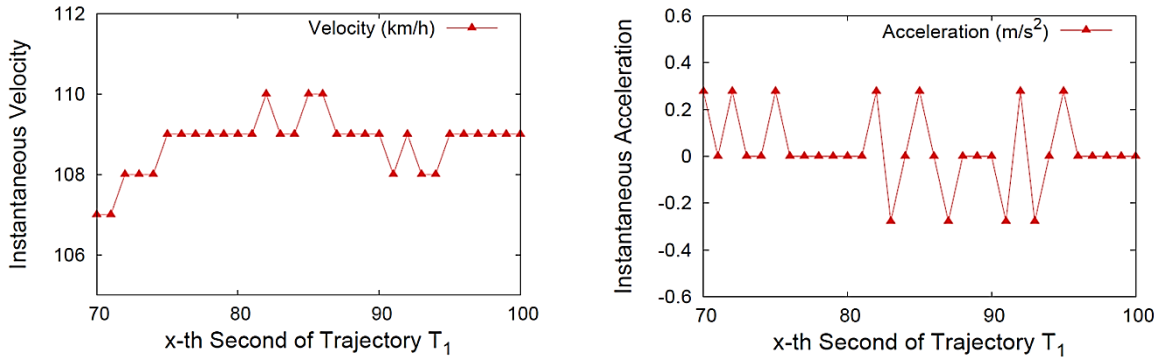
Table 9: Consistencies Between Instantaneous Models

As an example, a trajectory T_1, with 2, 375 GPS observations, is chosen to compare the behaviors of the instantaneous models. Figure 19 shows the instantaneous environmental impact from the 70th to the 100th seconds as computed by the models; Figure 20 shows the instantaneous velocities and accelerations during the same period. also indicates that the instantaneous environmental impact estimated by most models, except MEF, are highly correlated with accelerations.



(a) With relative values (b) With absolute values

Figure 19: Instantaneous Environmental Impact



(a) Instantaneous Velocities (b) Instantaneous Accelerations

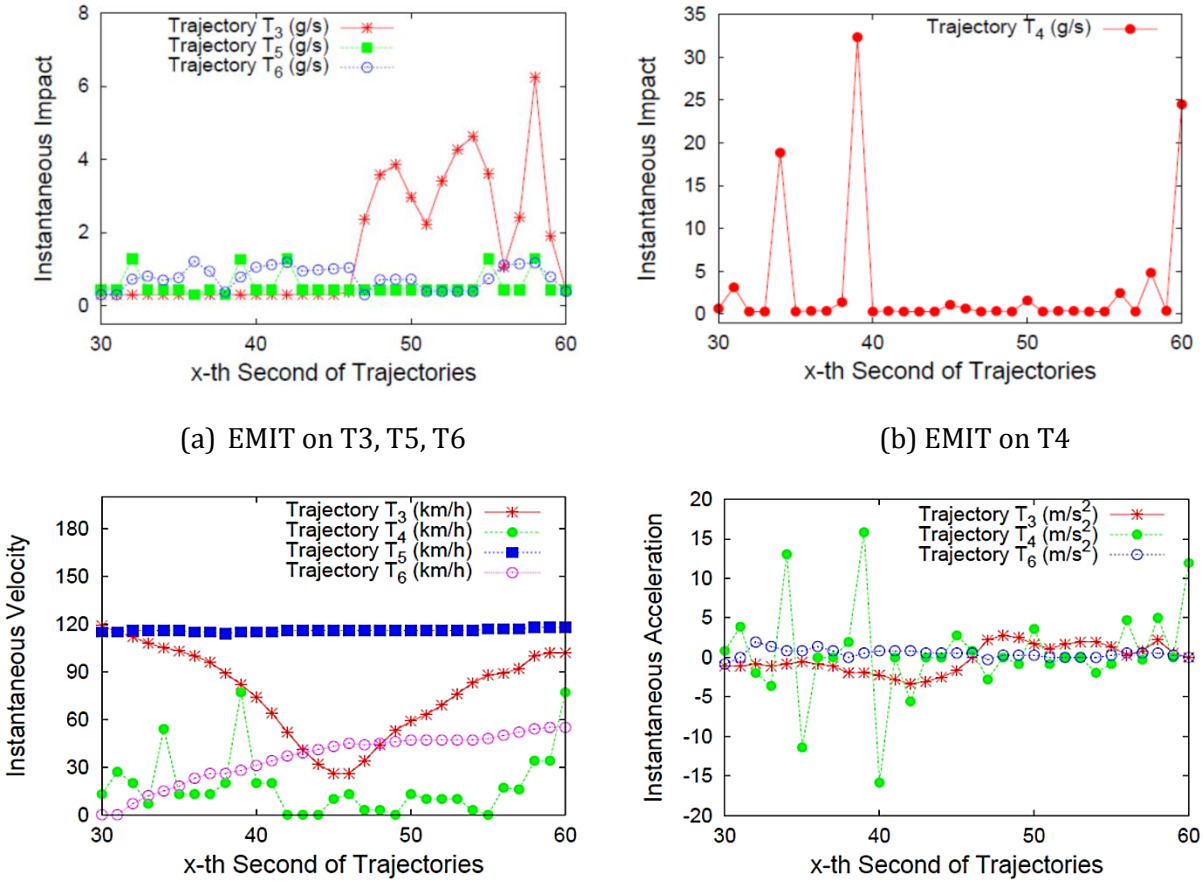
Figure 20: Instantaneous Velocities and Accelerations

Experiments are also conducted on the environmental impact caused by different driving behaviors, which in turn are indicated by different instantaneous velocities and accelerations. Four trajectories with distinct driving behaviors are chosen, as shown in Table 10, whose velocities and accelerations are also plotted in Figure 21(c) and (d). The accelerations of T5, whose values are around zero, are omitted to increase the readability of Figure 21(d).

	High Velocity	Low Velocity
Aggressive Acceleration	$\mathcal{T}_3$	$\mathcal{T}_4$
Moderate Acceleration	$\mathcal{T}_5$	$\mathcal{T}_6$

Table 10: Trajectories with Different Driving Behaviors

EMIT shows high consistency with most instantaneous models (see Table 9) and thus is utilized to demonstrate the environmental impact of the four trajectories. As shown in Figure 21(a) and (b), the results indicate that: (1) environmental impact of moderate accelerations (T5 and T6) is much less than the impact of aggressive accelerations (T3 and T4); (2) driving with high velocities and aggressive accelerations yields higher environmental impact than when driving with low velocities and aggressive accelerations. For example, although the variations in the accelerations of T4 is greater than those of T3, T4 generally has a lower environmental impact than does T3.



(a) EMIT on T3, T5, T6

(b) EMIT on T4

(c) Instantaneous Velocities

(d) Instantaneous Accelerations

Figure 21: Environmental Impact vs Driving Behaviours

4.4.3 Evaluating aggregated models

Aggregated models are used to estimate environmental impact per unit length on road segments traversed by trajectories. A similar consistency comparison is conducted to gain insight into the similarity of each pair of aggregated models by applying Equation 1. As shown in Table 11, all models show similar estimation trends with similarities higher than 80%. Having similarities of at least 96.9%, Tavares, SIDRA-Running, and SIDRA-Avg are highly consistent with each other.

	Tavares	SIDRA-4Mode	SIDRA-Running	SIDRA-Avg
Song	86.4%	81.4%	88.3%	88.8%
Tavares	1	90.8%	96.9%	97.3%
SIDRA-4Mode	-	1	88.6%	88.4%
SIDRA-Running	-	-	1	99.7%

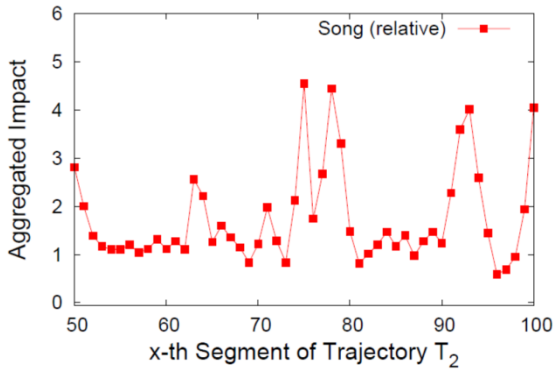
Table 11: Consistencies of Aggregated Models

Trajectory T2, which covers 114 road segments, is utilized as an example to show the behaviors of the aggregated models. The aggregated impact estimated by the aggregated models between the 50th to the 100th road segments is reported in Figure 22. In general, all the aggregated models behave similarly: the average velocity on a segment is inversely proportional to the environmental impact per unit length.

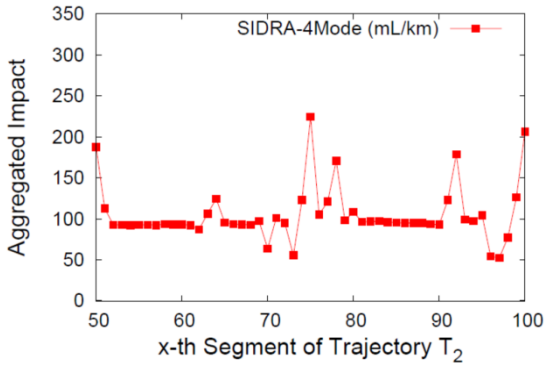
For example, the velocities on the 75th and 78th segments (see the two dips in Figure 22(d)) are lower than on other segments traversed by T2, and thus the corresponding environmental impact per unit length on the two segments is higher. As it takes longer to traverse a unit length with a lower velocity, a higher environmental impact per unit length results.

Existing routing algorithms [65, 66] can be employed to enable eco-routing if the edge weights reflect the environmental impact of traversing the edges (i.e., road segments). Based on the findings reported in Figure 22 and Table 11, we argue that the aggregated models are appropriate for assigning eco-weights (i.e., environmental impact related weights) to road segments in order to enable eco-routing. We suggest the use of SIDRA-Running because the minimum input required, the total length and total travel time (or average velocity) of segments, are fairly easy to obtain.

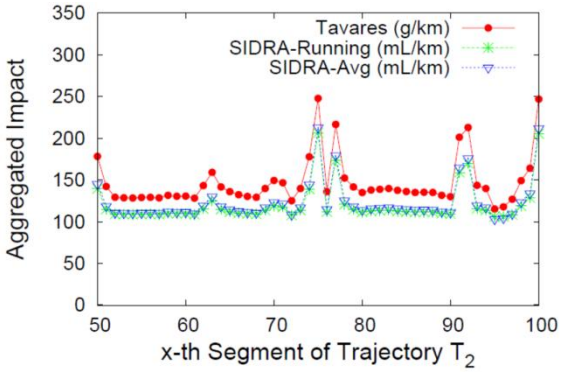
Although the impact estimated by Song and SIDRA-4mode are more sensitive to changes in velocity than for the other aggregated models (see Figure 22), these two models require instantaneous travel information during computation. Therefore, they are not easy to use in scenarios where only average velocities are known. SIDRA-Avg and Tavares are not recommended because the former cannot be used when velocities are above 50km/h [38] and the latter is intended for heavy vehicles.



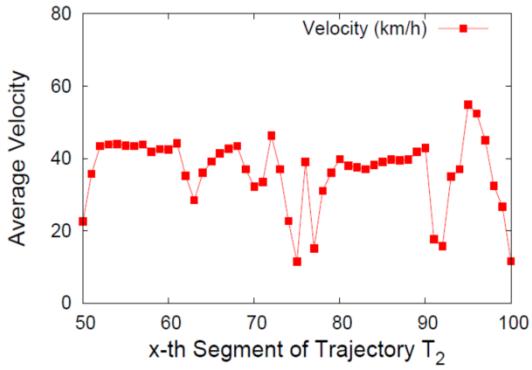
(a) Song



(b) SIDRA-4Mode

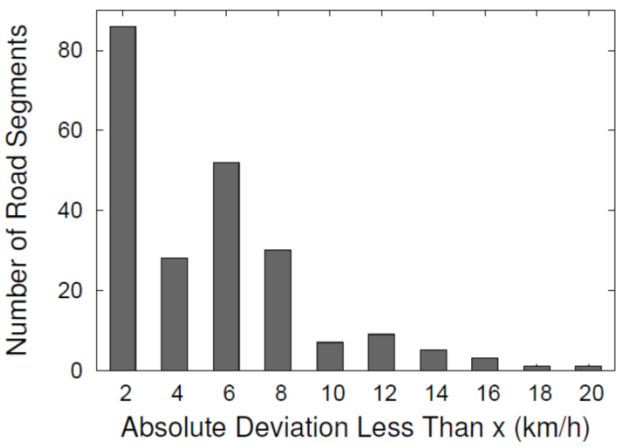


(c) Three Models

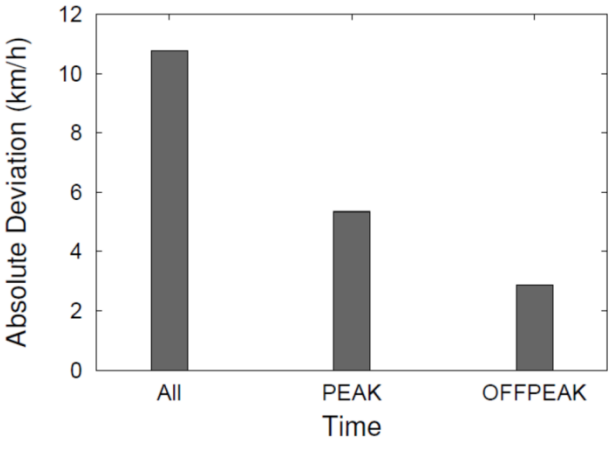


(d) Average Velocities

Figure 22: Comparison of Aggregated Models on Trajectory T2



(a) Absolute Deviation



(b) Temporal Variations

Figure 23: Aggregated Models and Eco-Weights

Given a road segment, the impact estimated by SIDRA-Running is closely related to the average velocity. Thus, it is useful to determine whether different trajectories on the same road segment have similar average velocity. Thus, we study the 222 road segments that are frequently traversed by T. The absolute deviations of the average velocities on these road segments are reported in Figure 23(a). We find that, on 86 segments, the absolute deviations of the average velocities are smaller than 2km/h, as seen in the first bar. Only on 3 segments are the absolute deviations of average velocities high, i.e., between 14km/h and 16km/h, as seen in the 8th bar. Figure 23(a) indicates that the average velocities on most road segments are similar, which renders it is possible to derive a single, meaningful eco-weight for each of such road segment.

The average velocities on some road segments vary significantly over time. In some cases, this is due to rush-hour congestion. We plot the absolute deviation during different periods of a particular road segment in Figure 23(b). Although the average velocities are quite different during the entire day (see the bar for All), the average velocities are similar during peak and off-peak periods (bars PEAK and OFFPEAK). Thus, time dependent eco-weights are suggested for such road segments, e.g., different eco-weights should be used for different periods.

For road segments where drivers behave significantly different, i.e., with large average velocity deviations and without clear peak and off-peak periods, ranges are suggested to be used as eco-weights. Alternatively, driver specific eco-weights should be considered.

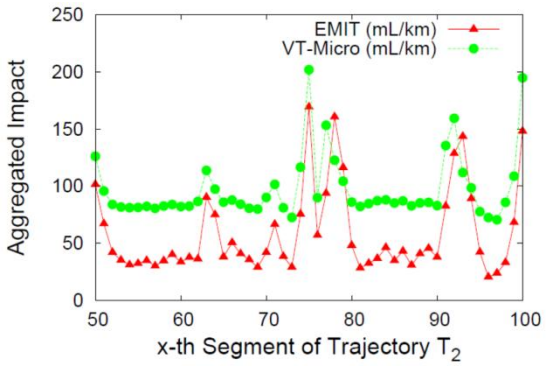
4.4.4 Aggregation of instantaneous models versus aggregated models

The aggregated environmental impact on a road segment can also be obtained by aggregating the instantaneous impact that is estimated by an instantaneous model on the road segment. In the following experiment, the aggregation is computed by summing the instantaneous impacts on the segment and dividing the sum by the length of the segment.

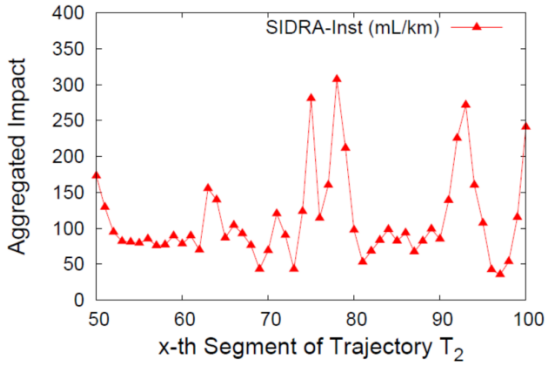
Trajectory T2 is utilized to demonstrate the degree of consistency of the aggregated impact estimated by aggregating instantaneous impact and the aggregated impact estimated directly by the aggregated models.

As shown in Figure 24, the aggregations of VT-Micro, EMIT, and SIDRA-Inst behave similarly to the aggregated models. In particular, VT-Micro shows the most consistent results. In contrast, the aggregations of Joumard, SP, and MEF do not behave consistently with the aggregated models and do not show relationships with average velocities (see Figure 22(d)).

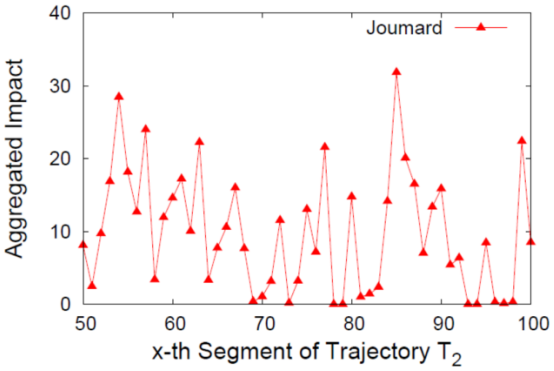
The experiment suggests that some instantaneous models are not suitable for aggregation in order to estimate the aggregated environmental impact and thus should not be used for assigning eco-weights to road segments. However, VT-Micro, EMIT, and SIDRA-Inst demonstrate the ability to provide reasonable aggregated environmental impact.



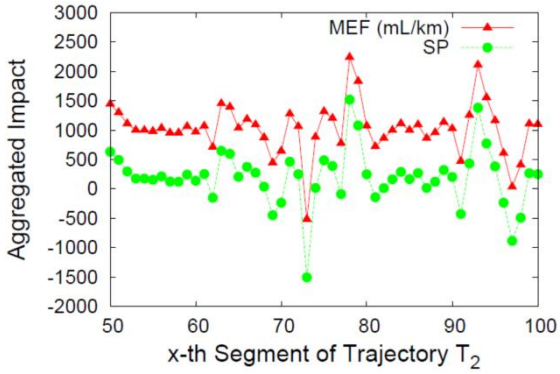
(a) EMIT and VT-Micro



(b) SIDRA-Inst



(c) Joumard



(d) SP and MEF

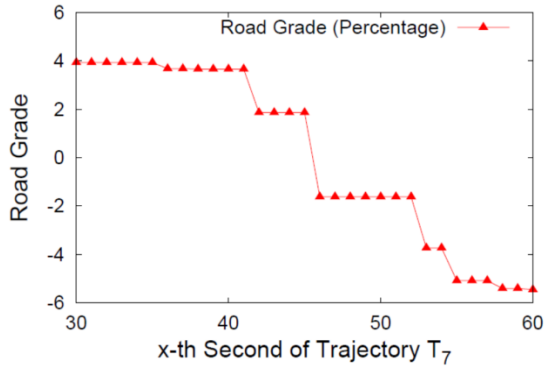
Figure 24: Aggregation of Instantaneous Models on Trajectory T2

4.4.5 Effect of road grades

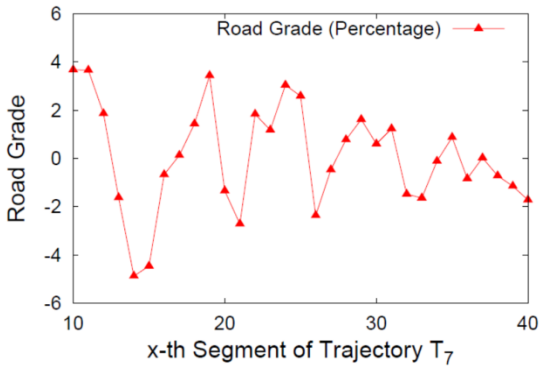
Experiments are carried out to evaluate the influence of varying road grades on environmental impact. A hilly route where the road segments have both positive and negative grades is, therefore, chosen—see Figure 25.

Figure 26 (a) and (b) plots the instantaneous impact estimated by SP and SIDRA-Inst with and without considering grades. The results demonstrate that positive (negative) road grades cause higher (lower) the environmental impact compared to the impact without considering grades. Two aggregated models, SIDRA-4Mode and Tavares, are chosen to demonstrate the effect of road grades on the aggregated impact. As shown in Figure 26(c) and (d), the aggregated impact estimated by both models is higher for segments with positive grade than for segments with negative grade. Tavares is more sensitive to road grades. The difference between the impact with and without

grades as estimated by Tavares is greater than for SIDRA-4Mode.

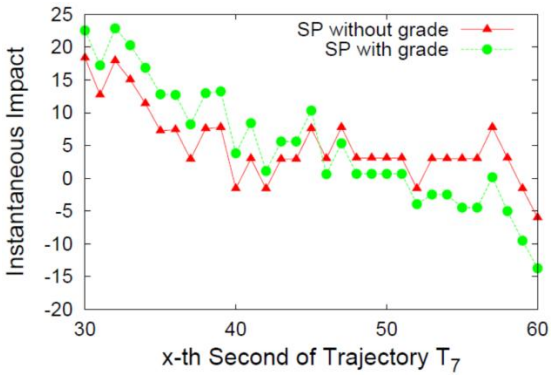


(a) For Instantaneous Models

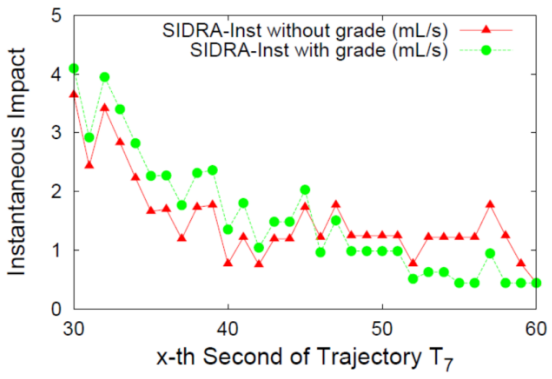


(b) For Aggregated Models

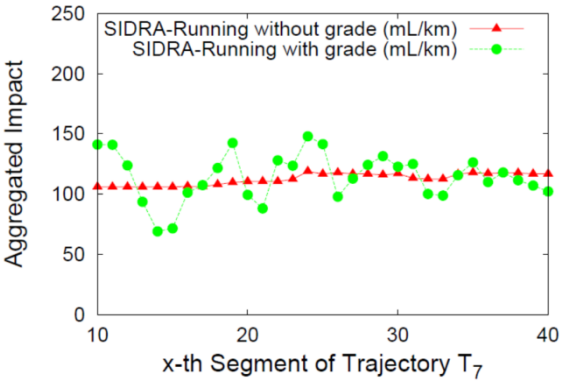
Figure 25: Road Grade



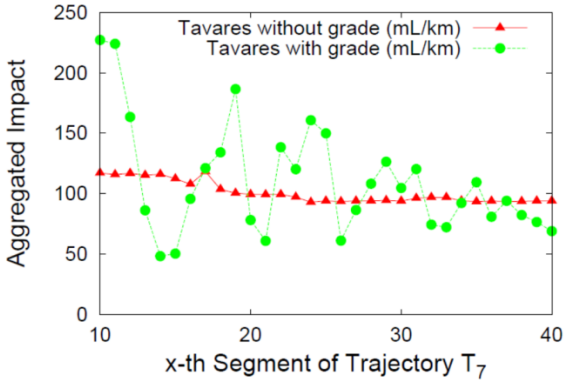
(a) SP



(b) SIDRA-Inst



(c) SIDRA-4Mode



(d) Taverse

Figure 26: Effect of Road Grades on Trajectory T7

4.4.6 Empirical findings

Instantaneous Models and Eco-driving: Instantaneous models estimate instantaneous environmental impact, e.g., second by second fuel usage, from instantaneous velocities and accelerations. Since instantaneous velocities and accelerations reflect driving behaviors, instantaneous models can be used to measure the environmental impact of different such behaviors. Instantaneous models, along with classical data mining methods, can be used to classify good and bad driving behaviors in terms of environmental impact, which in turn may suggest good driving behaviors to drivers. Thus, instantaneous models are appropriate for eco-driving applications.

Aggregated Models and Eco-routing: Aggregated models estimate environmental impact per unit length using average velocities or starting and ending velocities. Such impact can be utilized for assigning eco-weights to road segments, thus enabling eco-routing.

From this point of view, the aggregated models offer a foundation for eco-routing. In addition, instantaneous impact on a road segment predicted by an instantaneous model, e.g., VT-Micro, can be aggregated and then used as the eco-weight of the road segment.

We suggest employing SIDRA-Running to compute eco-weights for road segments because this the least input required by this model can be obtained from GPS trajectories. We also suggest that a single eco-weight per road segment suffices in some cases, while time dependent weights or a range of weights should be considered in other cases, as presented in Section 5.3.

Importance of 3D Spatial Networks: Road grades substantially affect the environmental impact predicted by both instantaneous and aggregated models. Therefore, the use of 3D spatial networks benefits both eco-driving and eco-routing.

4.5 Summary

We develop EcoMark that evaluates models of vehicular environmental impact. Eleven state-of-the-art impact models are categorized into instantaneous models and aggregated models. The models are compared and analyzed based on a substantial collection of 1 Hz GPS trajectories and a 3D spatial network. The empirical study suggests that the instantaneous models are appropriate for eco-driving, while the aggregated models are helpful for eco-routing. The use of a 3D spatial network that records road grades benefits both eco-driving and eco-routing.

5. Basic Eco-Routing Techniques

5.1 Input Specification

5.1.1 Road Map

A road map represents basic information about roads, intersections, and their geographic details in a certain region. The road map provides information necessary to plan transportations, to record traffic flows and to suggest travelling routes, etc.

OpenStreetMap [1] is employed in the techniques addressed in this report, as it offers geographic data and mapping to users and is free of charge. In general, OpenStreetMap provides the following information that is used in the techniques:

- Intersection: a road junction where two or more roads meet.
- Road segment: a particular part of a road that starts from an intersection or an end of the road and ends at a different intersection or a different end of the road. Specifically, there exist no intersections and road ends in the middle of a road segment.
- Coordination of each geographic point on a road segment: the longitude and latitude of each point on a road segment.

5.1.2 Spatial Network

A spatial network captures topological and geometric aspects of a map. For example, a spatial network is presented in Figure 27.

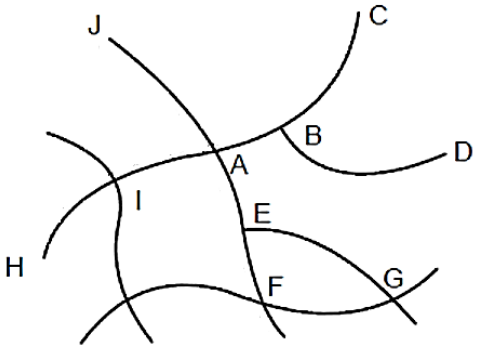


Figure 27: A Spatial Network

A spatial network is modeled as a directed, weighted graph G = (V, E, L, H, F), where,

- V is the vertex set;
- E is the edge set;
- L is a function that records the lengths of edges;
- H is a function that assigns travel time weights to edges; and
- F is a function that assigns fuel weights to edges.

Each component is covered in more detail as follows.

- A vertex $v_i \in V$ represents a road intersection or an end of a road.
- An edge $e_k \in E \subseteq V \times V$ is defined by a pair of vertices and represents a directed road segment that connects the (intersections represented by) two vertices. For example, edge (v_i, v_j) represents a road segments that enables travel from vertex vi to vertex v_j.
- Function L: $E \rightarrow R$ takes as input an edge and outputs the length of the road segment that the edge represents.
- Function H: $E \rightarrow R$ assigns time weights to all edges. In particular, H takes as input an edge, and outputs the average time for traversing the edge.
- Function F: $E \rightarrow R$ assigns fuel consumption weights to all edges. In particular, F takes as input an edge, and outputs the fuel consumed for traversing the edge. The fuel is derived from any of aggregated models, which are presented in Chapter Evaluating Models of Vehicular Environmental Impact.

Figure 28(a) captures the upper right part of the spatial network shown in Figure 27 more detail. Here, where Avenue 1 and Avenue 2 are bidirectional roads and Street 3 is a one-way road that only allows travel from vertex B to vertex D.

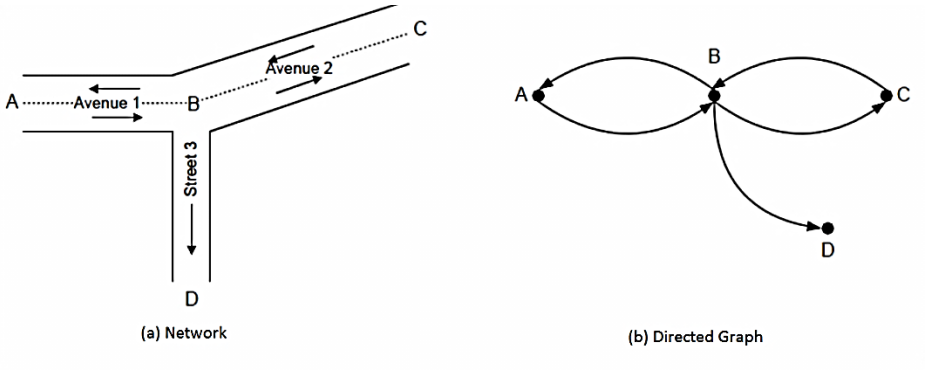


Figure 28: A Subset of Spatial Network

The corresponding directed graph is shown in Figure 28(b). In order to capture the bidirectional Avenue 1, two edges (A, B) and (B, A) are generated. Since Street 3 is a one-way road, only one edge, (B, D), is created in the directed graph. If road segment AB is 135 meters long, we have G.L((A, B)) = G.L((B, A)) = 135.

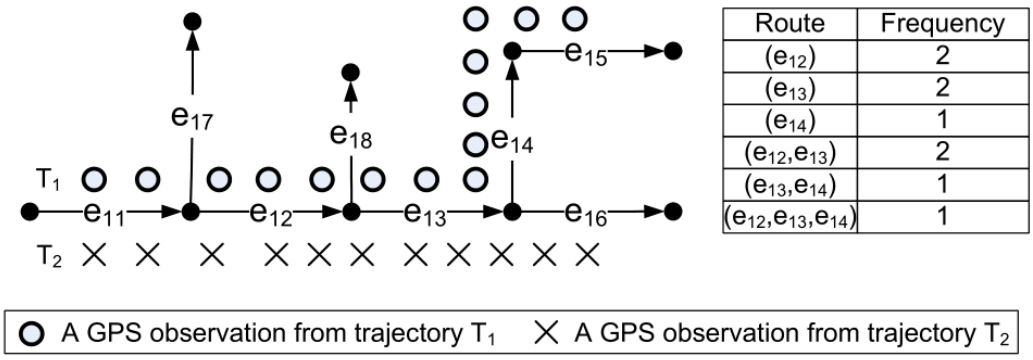
It is essential to model a road network as a directed graph because the cost associated with traveling in the two different directions may differ very substantially. For example, traveling uphill is likely to have a higher fuel cost than traveling downhill. As another example, the congestion may also vary greatly for the two directions of a road.

5.1.3 GPS Trajectories and Routes

A **trajectory**, denoted as T = (p_1, p_2, ..., p_X), is a sequence of GPS observations, where a GPS observation p_i specifies the location (typically in a "longitude-latitude" co-ordinate system) and the velocity of a vehicle at a particular time point. It is clear that instantaneous velocities are available from GPS trajectories, and instantaneous accelerations can be derived based on two consecutive GPS observations. Given a spatial network G, a map matching algorithm [2] is able to associate each observation in a trajectory with a specific edge in G.E.

A **route**, denoted as R = (r_1, r_2, ..., r_Y), is a sequence of edges, e.g., $r_i \in$ G.E. Two consecutive edges in a route shares a vertex, i.e., $r_i.t = r_{i+1}.s$ for $1 \le i \le Y - 1$. The route of a trajectory T, denoted as Route(T), is defined as the sequence of the corresponding map matched edges of trajectory T , excluding the first edge and the last edge.

Figure 29 shows two trajectories T_1 and T_2, both of which start from edge e_{11}, but stop on edges e_{15} and e_{16}, respectively. Thus, Route(T_1)=(e_{12}, e_{13}, e_{14}) and Route(T_2)=(e_{12}, e_{13}).



Route	Frequency
(e_{12})	2
(e_{13})	2
(e_{14})	1
(e_{12},e_{13})	2
(e_{13},e_{14})	1
(e_{12},e_{13},e_{14})	1

Figure 29: Trajectories, Routes, and Frequencies

If route R is a sub-sequence of the route of a trajectory, e.g., Route(T), we regard that trajectory T fully covers route R. The **frequency** of route R, denoted as freq(R), is the number of trajectories whose corresponding routes fully cover route R, which is formally defined as follows.

$$freq(R) = |\{T \mid R \rhd Route(T)\}|,$$

where $A_i \rhd A_j$ indicates sequence A_i is a sub-sequence of A_j; and $|\cdot|$ indicates the cardinality of a set. The table in Figure 3 shows an example of frequencies of routes.

Route queries. Given a graph G = (V, E, L, H, F), start and end vertices v_s, $v_e \in V$, a route query Q(G, v_s, v_e) searches a set of routes $R_{vs,ve}$ on G that starts from vertex v_s and ends at vertex v_e.

Shortest route queries. Given a graph G = (V, E, L, H, F), start and end vertices v_s, $v_e \in V$, a shortest route query Qs(G, v_s, v_e) returns a route Rs $\in R_{vs,ve}$ between v_s and v_e, such that the total distance between v_s and v_e is smaller than any other routes between v_s and v_e. In particular, Function L: E → R defined in G is applied to compute the distance between v_s and v_e.

Fastest route queries. Given a graph G = (V, E, L, H, F), start and end vertices v_s, $v_e \in V$, a fastest route query Qf(G, v_s, v_e) returns a route Rf $\in R_{vs,ve}$ between v_s and v_e, such that the total time required to traverse on Rf is shorter than any other routes between v_s and v_e. In particular, Function H: E → R defined in G is applied to compute the travel time between v_s and v_e.

Eco-route queries. Given a graph G = (V, E, L, H, F), start and end vertices v_s, $v_e \in V$, an eco-route query Qe(G, v_s, v_e) returns a route Re $\in R_{vs,ve}$ between v_s and v_e, such that the environmental impact between v_s and v_e is smaller than any other routes between v_s and v_e. In particular, Function F: E → R defined in G is applied to compute the environmental impact between v_s and v_e.

5.2 Basic eco-routing algorithm

Given a graph G for modeling spatial networks, where various weights are available, e.g., distance weights, travel time weights and environmental impact weights, classical routing algorithms can be applied to conduct basic eco-routing techniques.

Dijkstra's algorithm was proposed by Edsger Dijkstra in 1956. Initially, given two vertices v_s and v_e on a graph G, the algorithm is designed for searching a shortest path between vertices v_s and v_e on the graph, where the weight on each edge is the length of the edge. Later, the weight on each edge can be changed into the travel time needed to traverse the edge. Thus, Dijkstra's algorithm is directly utilized to compute the shortest route and the fastest route between two vertices v_s and v_e.

In this report, Dijkstra's algorithm is adapted to compute an eco-route path between vertices v_s and v_e on the graph G = (V, E, L, H, F). Given the vertex v_s, the algorithm finds a route with least environmental impact between v_s and every possible vertex. Once the least environmental impact route between the start vertex v_s and the end vertex v_e is determined, the search process is stopped.

The pseudocode for basic eco-routing algorithm, which is based on Dijkstra's algorithm, is presented Algorithm 3.

Input :
Graph $G = (V, E, L, H, F)$,
Start vertex v_s, End vertex v_e.
Output:
An eco-route r between v_s and v_e.
/* Initializations */
for *each vertex* $v \in G.V$ **do**
  $fuel[v] \leftarrow \infty$;
  $previous[v] \leftarrow null$;
$fuel[v_s] \leftarrow 0$;
Vertex List $Q \leftarrow G.V$;
/* Search an eco-route between v_s and each vertex in $G.V$ */
while $Q \neq \emptyset$ **do**
  sort $fuel$ by desc;
  vertex $u \leftarrow fuel[0]$'s vertex;
  if $fuel[u]$ *is* ∞ **then**
    break;
  remove u from Q;
  for *each neighbor* u_n *of* u **do**
    $temp = fuel[u] + G.F(e_{u,u_n} \in G.E)$;
    if $temp < fuel[u_n]$ **then**
      $fuel[u_n] \leftarrow temp$;
      $previous[u_n] \leftarrow u$;
/* Get the eco-route between v_s and v_e */
vertex $u_p \leftarrow previous[v_e]$;
while $u_p \neq null$ **do**
  $r.add(u_p)$;
  $u_p \leftarrow previous[u_p]$;
Reverse route r;

Algorithm 3: Basic Eco-Routing Algorithm

5.3 Experiment Description

Spatial Networks: We use a spatial network for our experiments. It contains almost the whole Jutland, Denmark, which results in a graph consisted of 17956 vertices and 39372 edges.

Edge weights: For the environmental impact based graph weight annotation, we use the fuel

consumption on each edge as the weight. Ideally, the exact fuel consumption should be obtained from CAN bus data or sensor data, however, not every vehicle is equipped with pertinent sensors. Thus, we apply the SIDRA-Running model [38] to compute the fuel consumption of each edge. The model takes as input the length and speed limit of each edge, which are derived from OpenStreetMap [1], and estimates the environmental impact of the edge.

Experimental description: 100 pairs of start and end vertices are chosen arbitrarily from the graph. For each pair of vertices, 3 types of routes, i.e., the shortest route, the fastest route and the eco-route, are queried using the classical Dijkstra's algorithm and Algorithm 3, respectively.

The same experiment is repeated for 5 times, whose results are shown in Table 12. The first column of Table 12 lists five possible result categories when given a pair of start and end vertices:

1 shortest = fastest = eco means that the shortest route, the fastest route and the eco-route are the same;

2 shortest = fastest != eco means that the shortest route and the fastest route are the same, but are different from the eco-route.

3 shortest!= fastest = eco means that the fastest route and the eco route are the same, but are different from the shortest route.

4 shortest = eco!= fastest means that the shortest route and the eco-route are the same, but are different from the fastest route.

5 shortest!= fastest != eco means that the shortest route, the fastest route and the eco-route are different.

Given a pair of vertices, the shortest route, the fastest route and the eco-route belong to one of the above categories. Thus, the number of vertex pairs, whose routes belong to a specific category, is presented in Table 12.

As can be observed, over 90 pairs out of 100 in each run, the shortest route, the fastest route and the eco-route are different between the start and the end vertices. This observation indicates that the three types of routes are usually different in most cases. Therefore, specific attention to eco-routing is important and necessary.

Result Category		Exp Run 1	Exp Run 2	Exp Run 3	Exp Run 4	Exp Run 5
1	shortest = fastest = eco	0	0	0	1	1
2	shortest = fastest != eco	2	1	3	3	1

3	shortest!= fastest = eco	0	0	0	0	0
4	shortest = eco!= fastest	3	1	1	2	2
5	shortest!= fastest != eco	95	98	96	94	96

Table 12: Results for Eco-routing Algorithm

Figure 30 shows an example for the shortest, the fastest and eco routes between vertices A and D. As can be observed, the three routes are the same between vertices A and B, and between vertices C and D, but become different between vertices B and C.

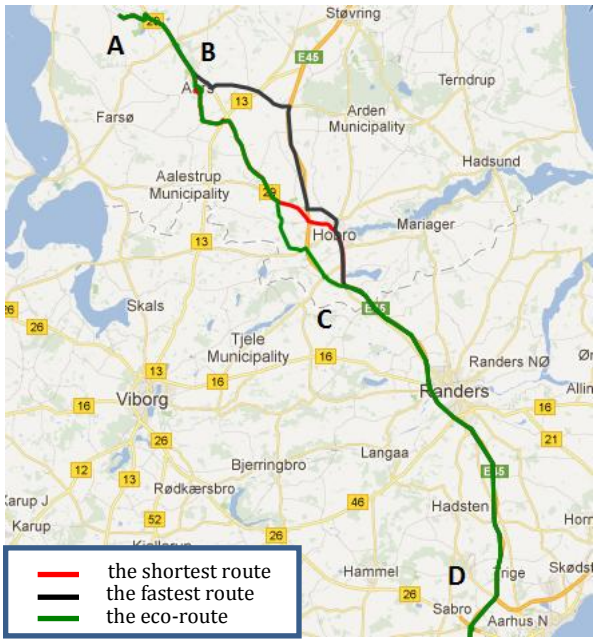


Figure 30: An example for the shortest route, the fastest route and the eco-route

In Category 5, the shortest route, the fastest route and the eco-route are different between a pair of start and end vertices. Thus, between each pair of vertices in Category 5, the three routes are compared among themselves in terms of the total traversed distance, the total travel time and the total environmental impact, respectively.

The results are presented in Table 13. In particular, "S" represents the shortest route, "F" represents the fastest route, and "E" represents the eco-route.

Comparison Category		Exp Run 1	Exp Run 2	Exp Run 3	Exp Run 4	Exp Run 5
Distance comparison	S<E<F	14	16	16	10	15
	S<F<E	81	82	80	84	81
	S<E=F	0	0	0	0	0
Travel time comparison	F<E<S	0	0	1	0	0
	F<S<E	95	98	95	94	96
	F<E=S	0	0	0	0	0
Environmental comparison	E<F<S	0	0	0	0	2
	E<S<F	95	98	96	94	94
	E<S=F	0	0	0	0	0

Table 13: Comparison between the shortest, the fastest and the eco routes

Let us take distance comparison as an example. In the first run of the experiment, there are 14 pairs of vertices whose fastest routes run longer distance than the eco-routes in total 95 pairs. In the rest 81 pairs, the traversed distances of eco-routes are the longest.

Observations are summarized from Table 13 as follows.

- Eco-routes usually traverse longer distances and take longer time than the shortest routes and the fastest routes.
- In terms of total environmental impact, the fastest routes always generate the most environmental impact, the next is the shortest route, and the last is the eco-route.

5.4 Summary

The basic eco routing techniques presented in this chapter are developed based on a map (i.e., OpenStreetMap), a spatial network that models map information in a particular region as a directed graph. Different types of routes are defined, including the shortest route, the fastest route and the eco-route. Classical Dijkstra's algorithm is adapted to conduct the basic eco-routing task. Intensive experiments are carried out to show that the eco-routes are not always the same as the shortest routes and the fastest routes between two vertices v_s and v_e.

6. Risk Assessment

The failure to acquire CAN-BUS data has been identified as low risk, because it was always something to be used for the evaluation of the accuracy of our methods, but not a core component in the methods itself. Our algorithms are purely designed based on raw GPS trajectories and hence can be built independently of CAN-BUS data.

Our EcoMark benchmark allows us to compare various fuel computation models and chose the ones that suit our scenarios the best. We intend to use relative fuel costs to understand how the eco-routes compare to one another.

We also identified that our solution caters only to passengers and does not take into account freight transportation.

7. Conclusion

In this deliverable we study existing fuel computation models and compared them. In order to do so we needed to build more infrastructure which included generation of an accurate 3D spatial network by combining a 2D road network with 3D-LiDAR (laser scan data) data. The availability of slope information affected the fuel computation model's respective accuracies. We also computed a special graph for modeling a spatial network that records information about road intersections and road segments, and measures the length of each road segment, the travel time required to traverse the road segment, and fuel consumption or GHG emission on the road segment. The classical Dijkstra's algorithm, which is originally used for searching the shortest routes and the fastest routes, is adapted to conduct the basic eco-routing tasks.

References

[1] Openstreetmap. http://www.openstreetmap.org/.

[2] F. Pereira, H. Costa, and N. Pereira. An off-line map-matching algorithm for incomplete map databases. European Transport Research Review, 1(3):107–124, 2009.

[3] G. Tavares, Z. Zsigraiova, V. Semiao, and M.G. Carvalho. Optimisation of msw collection routes for minimum fuel consumption using 3D GIS modelling. Waste Management, 29(3):1176–1185, 2009.

[4] L. Sugarbaker, G. Snyder, and D. Maune. Results of the national enhanced elevation assessment. Presentation to International LiDAR Mapping Forum, 2012.

[5] Nevada Legislature. AB 511. http://tinyurl.com/7wtdcp8.

[6] Rushlane. Google self driving technology with laser lidar: Autonomous car accurate to 2 cms. http://tinyurl.com/727owuo.

[7] Official Google Blog. What we're driving at. http://tinyurl.com/73wv9eu.

[8] NASA Jet Propulsion Laboratory. Shuttle radar topography mission. http://www2.jpl.nasa.gov/srtm.

[9] M. Over, A. Schilling, S. Neubauer, and A. Zipf. Generating web-based 3D city models from OpenStreetMap: the current situation in Germany. Computers, Environment and Urban Systems, 34(6):496–507, 2010.

[10] Herman J. Haverkort and Constantinos P. Tsirogiannis. Flow on noisy terrains: an experimental evaluation. In GIS, pages 84–91, 2011.

[11] M. de Berg and C P. Tsirogiannis. Exact and approximate computations of watersheds on triangulated terrains. In GIS, pages 74–83, 2011.

[12] L. Liu and R. C.-W. Wong. Finding shortest path on land surface. In SIGMOD, pages 433–444, 2011.

[13] Y. Liu, Z. Li, R. Hayward, R. Walker, and H. Jin. Classification of airborne lidar intensity data using statistical analysis and hough transform with application to power line corridors. In Digital Image Computing: Techniques and Applications, pages 462–467, 2009.

[14] A. S. Antonarakis, K. S. Richards, and J. Brasington. Object-based land cover classification using airborne lidar. Remote Sensing of Environment, 112(6):2988–2998, 2008.

[15] Thomas Brinkhoff, Hans-Peter Kriegel, and Bernhard Seeger. Efficient processing of spatial joins using R-trees. In SIGMOD, pages 237–246, 1993.

[16] M.-L. Lo and C. V. Ravishankar. Spatial joins using seeded trees. In SIGMOD, pages 209–220, 1994.

[17] Y.-W. Huang, N. Jing, and E. A. Rundensteiner. Spatial joins using R-trees: Breadth-first traversal with global optimizations. In VLDB, pages 396–405, 1997.

[18] M. Isenburg and P. Lindstrom. Streaming meshes. In IEEE Visualization, page 30, 2005.

[19] J. L. Hennessy and D. A. Patterson. Computer Architecture, Fourth Edition: A Quantitative Approach. Morgan Kaufmann, 2006.

[20] M. de Berg. Computational Geometry: Algorithms and Applications. Springer, 2000.

[21] U. Ramer. An Iterative Procedure for the Polygonal Approximation of Plane Curves. Defense Technical Information Center, New York University, 1972.

[22] Bing maps. http://www.bing.com/maps/.

[23] Energy and climate goals of china's 12th five-year plan. http://tinyurl.com/73mntqg.

[24] Fact sheet: Australia's emissions reduction targets. http://tinyurl.com/7vpzufx.

[25] G8 plans 50% reduction in greenhouse gases. http://tinyurl.com/68hn2a.

[26] Google maps. http://maps.google.com/.

[27] Reducing emissions from transport. http://tinyurl.com/7ex6am3.

[28] Road transport: Reducing co2 emissions from light-duty vehicles. http://tinyurl.com/6gtlzzz.

[29] Shuttle radar topography mission. http://www2.jpl.nasa.gov/srtm/.

[30] Usgs center for lidar information coordination and knowledge. http://lidar.cr.usgs.gov/.

[31] What is the eu doing on climate change? **http://tinyurl.com/75qex2g**.

[32] K. Ahn and H. Rakha. The effects of route choice decisions on vehicle energy consumption and emissions. Transportation Research Part D: Transport and Environment, 13(3):151–167, 2008.

[33] K. Ahn, H. Rakha, A. Trani, and M. Van Aerde. Estimating vehicle fuel consumption and emissions based on instantaneous speed and acceleration levels. Journal of Transportation Engineering, 128(2):182–190, 2002.

[34] ARB. EMFAC2007 version 2.30: Calculating emission inventories for vehicles in California. U.S. the California Air Resources Board (ARB), 2007.

[35] J. Barkenbus. Eco-driving: An overlooked climate change initiative. Energy Policy, 38(2):762–769, 2010.

[36] M. Barth and K. Boriboonsomsin. Energy and emissions impacts of a freeway-based dynamic eco-driving system. Transportation Research Part D: Transport and Environment, 14(6):400–410, 2009.

[37] P. Boulter, I. McCrae, and T. Barlow. A review of instantaneous emission models for road vehicles. Transport Research Laboratory, Published Project Report 267, Final, 2007.

[38] D. Bowyer, R. Akçelik, D. Biggs, and A. R. R. Board. Guide to Fuel Consumption Analyses for Urban Traffic Management. Technical report, Australian Road Research Board, 1985.

[39] A. Cappiello. Modeling traffic flow emissions. Master's thesis, Massachusetts Institute of Technology, 2002.

[40] A. Cappiello, I. Chabini, E. K. Nam, A. Lue, and M. A. Zeid. A statistical model of vehicle emissions and fuel consumption. In The IEEE 5th International Conference on Intelligent Transportation Systems, pages 801–809. IEEE, 2002.

[41] E. Dijkstra. A note on two problems in connexion with graphs. Numerische mathematik, 1(1):269–271, 1959.

[42] EEA. Copert 4 User Manual (version 5), Computer Programme to Calculate Emissions from Road Transport. European Environment Agency, 2007.

[43] EPA. User's Guide to MOBILE6, Mobile Source Emission Factor Model. U.S. EPA National Vehicle and Fuel Emissions Laboratory, 2001.

[44] EPA. Technical Guidance on the Use of MOVES2010 for Emission Inventory Preparation in State Implementation Plans and Transportation Conformity. U.S. EPA National Vehicle and Fuel Emissions Laboratory, 2010.

[45] S. Hausberger, M. Rexeis, M. Zallinger, and R. Luz. Emission factors from the model phem for the hbefa version 3, 2009.

[46] J. Hickman, D. Hassel, R. Joumard, Z. Samaras, and S. Sorenson. Methodology for calculating transport emissions and energy consumption. TRL Report SE/491/98: Deliverable for EU project MEET, 1999.

[47] J. L. Jiménez-Palacios. Understanding and Quantifying Motor Vehicle Emissions with Vehicle Specific Power and TILDAS Remote Sensing. PhD thesis, Massachusetts Institute of Technology, 1998.

[48] R. Joumard et al. Methods of estimation of atmospheric emissions from transport: European scientist network and scientific state-of-the-art. INRETS report LTE, 9901, 1999.

[49] R. Joumard, P. Jost, J. Hickman, and D. Hassel. Hot passenger car emissions modelling as a function of instantaneous speed and acceleration. Science of the Total Environment, 169:167–174, 1995.

[50] M. Kaul, B. Yang, and C. S. Jensen. Lifting a spatial network using laser scan data. In Submission.

[51] T. Kono, T. Fushiki, K. Asada, and K. Nakano. Fuel consumption analysis and prediction model for SecoT route search. In 15th World Congress on Intelligent Transport Systems and ITS America's 2008 Annual Meeting, 2008.

[52] W. Lei, H. Chen, and L. Lu. Microscopic Emission and Fuel Consumption Modeling for Light-duty Vehicles Using Portable Emission Measurement System Data. In World Academy of Science, Engineering and Technology, pages 918–925, 2010.

[53] E. K. Nam and R. Giannelli. Fuel consumption modeling of conventional and advanced technology vehicles in the Physical Emission Rate Estimator (PERE). Technical report, U.S. Environmental Protection Agency, 2005.

[54] L. Ntziachristos, Z. Samaras, and E. E. Agency. COPERT III Computer programme to calculate emissions from road transport: methodology and emission factors (version 2.1). European Environment Agency, 2000.

[55] L. Pelkmans, P. Debal, T. Hood, G. Hauser, and M.-R. Delgado. Development of a simulation tool to calculate fuel consumption and emissions of vehicles operating in dynamic conditions. SAE, 2004.

[56] F. Pereira, H. Costa, and N. Pereira. An off-line map-matching algorithm for incomplete map databases. European Transport Research Review, 1(3):107–124, 2009.

[57] T. Peucker, R. Fowler, J. Little, and D. Mark. The triangulated irregular network. In Amer. Soc. Photogrammetry Proc. Digital Terrain Models Symposium, volume 516, page 532, 1978.

[58] H. Rakha, K. Ahn, K. Moran, B. Saerens, and E. Van den Bulck. Virginia tech comprehensive power-based fuel consumption model: Model development and testing. Transportation Research Part D: Transport and Environment, pages 492–503, 2011.

[59] H. Rakha, K. Ahn, and A. Trani. Comparison of MOBILE5a, MOBILE6, VT-MICRO, and CMEM models for estimating hot-stabilized light-duty gasoline vehicle emissions. Canadian Journal of Civil Engineering, 30(6):1010–1021, 2003.

[60] H. Rakha, K. Ahn, and A. Trani. Development of VT-Micro model for estimating hot stabilized light duty vehicle and truck emissions. Transportation Research Part D: Transport and Environment, 9(1):49–74, 2004.

[61] G. Scora and M. Barth. Comprehensive Modal Emission Model (CMEM), Version 3.01 UserŠs Guide. University of California Riverside Center for Environmental Research and Technology, 2006.

[62] G. Song, L. Yu, and Z. Wang. Aggregate Fuel Consumption Model of Light-Duty Vehicles for Evaluating Effectiveness of Traffic Management Strategies on Fuels. Journal of Transportation Engineering, 135:611–618, 2009.

[63] G. Tavares, Z. Zsigraiova, V. Semiao, and M. Carvalho. Optimisation of MSW collection routes for minimum fuel consumption using 3D GIS modelling. Waste Management, 29(3):1176–1185, 2009.

[64] H. Teng, L. Yu, and Y. Qi. Statistical Microscale Emission Models Incorporating Acceleration and Deceleration. In 81st Transportation Research Board Annual Meeting, 2002.

[65] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang. T-drive: driving directions based on taxi trajectories. In GIS, pages 99–108. ACM, 2010.

[66] D. Luxen and C. Vetter. Real-time routing with openstreetmap data. In GIS, pages 513–516. ACM, 2011.

[67] D. Pfoser, S. Brakatsoulas, P. Brosch, M. Umlauft, N. Tryfona, andG. Tsironis. Dynamic travel time provision for road networks. In GIS, page 68. ACM, 2008.

[68] EMEP/CORINAIR Emission Inventory Guidebook - 2007. Technical report, EEA (European Environment Agency), 2007.

[69] R. Akcelik. Operating cost, fuel consumption and pollutant emission savings at a roundabout with metering signals. In ARRB CONFERENCE, 22ND, 2006, CANBERRA, ACT, AUSTRALIA, 2006.

[70] F. AN, M. BARTH, J. NORBECK, and M. ROSS. Development of comprehensive modal emissions model operating under hot-stabilized conditions. Transportation research record, (1587):52–62, 1997.

[71] R. K. Ganti, N. Pham, H. Ahmadi, S. Nangia, and T. F. Abdelzaher. Greengps: a participatory sensing fuel-efficient maps application. In MobiSys, pages 151–164, 2010.

[72] P. J. Sturm et al. Instantaneous Emission Data and Their Use in Estimating Passenger Car Emissions. Technische Universit¨t Graz, MEET Deliverable 6, COST 319 Final Report A2 edition, 1998.

[73] M. J. Barth, F. An, T. Younglove, G. Scora, C. Levine, M. Ross, and T. Wenzel. Development of a comprehensive modal emissions model, Final Report Prepared for: National Cooperative Highway Research Program Transportation Research Board National Research Council edition, April 2000.

[74] M. Hall and L. WILLUMSEN. SATURN-a simulation-assignment model for the evaluation of traffic management schemes. Traffic Engineering and Control, 21(4), 1980.

[75] J. Kent, J. Tomlin, and K. Post. Fuel Consumption Modelling in Traffic Links. Technical report, 1982.

[76] H. Rakha and K. Ahn. Integration Modeling Framework for Estimating Mobile Source Emissions. Journal of Transportation Engineering,´130(2):183–193, 2004.

Glossary

Term	Description
GHG	Green House Gas.
GPS	Global Positioning System.
CAN-BUS	Detailed information about a single vehicle, e.g., actual fuel consumption, engine RPM, and throttle position.
Trajectory	A sequence of GPS observations.